Faculty of Engineering and Information Technology
University of Technology Sydney

High-density Visual Crowd Counting with Perspective Understanding in Deep Neural Networks

A thesis submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

by

Muming Zhao

January 2020

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This research is supported by the Australian Government Research Training Program.

This thesis is the result of me conducted jointly with Shanghai Jiao Tong University as part of a collaborative Doctoral degree.

Signature of Candidate: Production Note: Signature removed prior to publication.

Date: 2020/01/23

Acknowledgments

I would like to thank my principal advisor Jian Zhang in UTS, for his continuous support either in my research or in the finance. He had always expressed his patience to me and never gave me up even at the time when I stuck in my progress. His enthusiasm and keen sense to the research helped guide and push me to complete different works in this thesis. I also want to express my gratitude to my co-supervisor Chongyang Zhang in SJTU for his significant support of my research. He used to spend a lot of time to help me revise my paper and give me valuable comments to improve my academic skills. I also would like to thank my principal supervisor Wenjun Zhang in SJTU, for his continuous support of my dual PhD study in UTS and SJTU. Without the three supervisors, this thesis would be impossible.

I want to thank all my colleagues: Xiaoshui Huang, Yazhou Yao, Junjie Zhang, Jiangchao Yao and Yuangang Pan. I appreciate the time they have spent discussing with me, where I have got inspired a lot.

Finally and most essentially, I am grateful for all the support from my parents, my sister and my dear friends. They are the source of my strength.

Contents

List of Figures

List of Tables

List of Publications

Papers published

- **Muming Zhao**, Jian Zhang, Chongyang Zhang, Wenjun Zhang: Leveraging Heterogeneous Auxiliary Tasks to Assist Crowd Counting, *30th IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 2019.

- **Muming Zhao**, Jian Zhang, Chongyang Zhang, Wenjun Zhang: Towards Locally Consistent Object Counting with Constrained Multi-stage Convolutional Neural Networks, *14th Asian Conference on Computer Vision* (ACCV), 2018.

- **Muming Zhao**, Jian Zhang, Fatih Porikli, Chongyang Zhang, Wenjun Zhang: Learning a perspective-embedded deconvolution network for crowd counting, *IEEE International Conference on Multimedia and Expo* (ICME) , 2017.

Papers in submission

- **Muming Zhao**, Jian Zhang, Chongyang Zhang, Fatih Porikli, Bingbing Ni, Wenjun Zhang: Scale-aware Crowd Counting via Depth-embedded Convolutional Neural Networks, *Transactions on Circuits and Systems for Video Technology* (TCSVT), *under review*.

Abstract

With population growth and worldwide urbanization, crowd gathering in public places has become more common. Thus estimating the number of people and measuring their density has become essential for practical applications such as physical security control and public space management. However, the complex environments of crowded scenes have imposed several challenges to general counting algorithms, among which scale variations of pedestrians is one of the most significant problems. With varying-sized objects, it is rather difficult for density-based counting systems to generate appropriate density estimations that conform to scale variations, which usually significantly degrades the counting accuracy. To handle the perspective distortion and the related scale-variation problem, traditional methods mainly perform feature normalization for perspective correction. However, within the deep learning framework, the perspective distortion has not been explicitly considered and addressed. Can we extend the mechanism of perspective handling with the powerful deep learning technique for further improvement? In this dissertation, we focus on measuring crowd density through deep architectures with in-network perspective understanding. Three works are presented. First, we develop a depth-embedded network that augments the original features to be scale-aware for more accurate density estimation. The depth map of a scene is encoded, rectified and finally embedded into the network via a proposed depth embedding module. Thus the objects, although in the same class, will attain distinct representations according to their scales in the feature space, which will directly benefit scale-aware density estima-

tions. We include a comprehensive comparison with various state-of-the-art methods for the task of crowd counting to verify the efficacy of incorporating geometric priors. Second, a multi-stage model with region-based supervisions is constructed to obtain robust features with implicit understandings of the scene geometry. With the internal multi-stage learning mechanism, features could be refined and adjusted repeatedly to perceive the scale variations. Besides, with local-based supervisions, the model is further constrained to generate locally consistent densities that conform to object scale variations. Experiments are presented to validate the effectiveness of the proposed model for crowd counting. Third, we build a multi-task framework that drives the network to embed desired semantic/geometric/numeric attributes to handle various type of challenges for crowd counting. With the multi-fold regularization effects introduced by three auxiliary tasks, the intermediate features are driven to convey desired properties and thus help improve the main task of density estimation. Extensive experiments have been conducted to indicate the effectiveness of the proposed method.

Chapter 1

Introduction

1.1 Background

Along with the explosive growth of the world's population, crowded scenes are easily observed in public areas such as stations, airports and large squares. Figure. 1.1 shows a sample of crowd images crawled from the Internet. In these situations, turbulence of a small group, such as pushing, stampede or crushes, would spread and eventually cause an overall loss of control of the whole crowd. One recently crowd crush example is the 2014 Shanghai stampede, where 36 people died and 49 were injured among around 300,000 people that gathered for the New Year celebration (Zhou, Pei & Wu 2018). To avoid such tragedy, crowd monitoring and analysis is drawing extensive attention owing to the intense demands of social security. Among the many challenging tasks in crowd analysis, crowd counting plays an essential role because it is one of the basic descriptors of crowd status and a significant indicator of possible intervention to gathering can be detected in time and further measures could be correspondingly conducted to avoid potential dangers.

Beyond for security monitoring and control, crowd counting also enjoys favors in other applications such as public space management. For example, as an intelligence indicator crowd counting can provide valuable information for retailers on the interests of customers by profiling the number of individ-

Figure 1.1: Illustration of crowded scenes.

uals browsing a product. Besides, the number of people in the queue will suggest a suitable number of opened counters to balance the human-resources and the checkout speed. Other information on the crowd flows at different times of a day or at the same time across one week can also be gathered to optimize the retail management (Loy, Chen, Gong & Xiang 2013).

Given the importance and popularity of recent research for visual crowd counting, in this thesis, we mainly study the crowd counting problem with the employment of novel computer vision techniques. This will take advantage of both the widely-distributed surveillance cameras in many public places as well as the automatic and real-time response of modern computer-vision based techniques. Given an image or a sequence of video depicting a crowd scenario, the research aims to develop algorithms to estimate the total number of people in the crowd automatically. Visual crowd counting can be seen as a specific category of the general object counting task, including vehicle counting (Onoro-Rubio & López-Sastre 2016), cell counting (Fiaschi, Köthe, Nair & Hamprecht 2012) and plants' leaf counting (Aich & Stavness 2017),

etc. However, it is unique and challenging due to the compound influences of several internal and external factors, e.g., limited sizes of objects in crowded scenes, cluttered background, severe scale variations of pedestrians, illumination changes, etc. For example, the small areas occupied by individuals make it infeasible to directly implement object detection algorithms to identify each pedestrian, and the cluttered background further increases the ambiguities to identify the foreground regions. Besides, the scale variation is also a critical challenge which is mainly caused by the perspective distortion in general surveillance scenes with the mount-view cameras, influencing the accuracy of pixel-wise crowd density estimation and the final counting results.

1.2 Scope and Limitation of Current Research

To improve the accuracy and the practicability of visual crowd counting, several methods have been proposed. These methods can be broadly divided into three categories: counting by detection, counting by trajectory clustering and counting by regression (Loy et al. 2013). The detection-based methods (Subburaman, Descamps & Carincotte 2012, Dollar, Wojek, Schiele & Perona 2012, Lin & Davis 2010, Li, Zhang, Huang & Tan 2008, Topkaya, Erdogan & Porikli 2014) mainly rely on the localization of single pedestrian with global or part detectors scanning over the image space with extracted features. However, due to the presence of crowding scenes, counting by detection is fragile and error-prone in situations where only limited pixels are occupied by each object. An alternative approach is counting by clustering (Brostow & Cipolla 2006, Rabaud & Belongie 2006). Based on the assumption that a crowd can be regarded as a composition of individual entities with unique yet coherent motion patterns, the number of people can be approximated by clustering the exposed set of trajectories in the crowd. Similar to the detection-based methods, counting by clustering also has limited capacity in crowded scenes where general trackers tend to fail. As an opposite innovation to the previous methods that mainly based on the delineation of each

Figure 1.2: Sample of a pair of image and its corresponding ground-truth density map for density-based counting methods.

entity in the crowd, another approach is inspired by the capability of human beings, in determining a rough density at a glance without enumerating the number of pedestrians in it. This approach is known as counting by regression (Chen, Gong, Xiang & Change Loy 2013, Loy et al. 2013, Change Loy, Gong & Xiang 2013), which counts people in the crowd by learning a direct mapping from low-level imagery features to global or local crowd count. To effectively exploit spatial information in an image, a seminal work (Lempitsky & Zisserman 2010) is proposed to learn to predict a pixel-wise density map for each input image, where the summation of density values over any local/global region reports the corresponding local/global object count. This method significantly contributes to the generation of interpretable prediction results for counting, instead of just outputting a count number which reveals little information on what the model is actually learning. Figure. 1.2 briefly illustrates this density-based counting pipeline, which is popular in most existing counting methods. For modern crowd counting applications that are generally focused on dense scenarios, regression-based methods have dominated the counting field. However, the hand-crafted features have limited the capacity of traditional counting methods.

Recently, the prevalence of deep learning (Krizhevsky, Sutskever & Hinton

2012) has significantly boosted the performance of a wide-area of vision tasks, and for crowd counting there is no exception with numerous work armed with the convolutional neural networks (CNNs) have emerged. For the CNN-based counting approaches, early methods (Zhang et al. 2015, Wang, Zhang, Yang, Liu & Cao 2015, Boominathan, Kruthiventi & Babu 2016) either directly estimate the total count or alternatively learn the count and density map through a shallow CNN, which wraps up the process of feature extraction and regression in traditional methods. To handle the drastic scale variation of pedestrians in crowd images, multi-scale-feature based methods have been firstly proposed in (Zhang et al. 2016). In this paper, a multi-column CNN with different receptive field for each column is introduced to generate multi-scale features, which are then fused to cover the scale variations in the crowd and generate the density map. This idea has been followed on by several variants, which mainly focus on two directions. The first class of methods develop algorithms to build multi-scale features, with image pyramid based multi-resolution input (Onoro-Rubio & López-Sastre 2016) or skip connections based on the hierarchy of CNNs (Zhang, Shi & Chen 2018). The other class of methods pay more attention to the algorithms to adaptively fuse the multi-scale features, with additional switch module (Sam, Surya & Babu 2017), teacher sub-network (Kumagai, Hotta & Kurita 2018) or attention modules (Kang & Chan 2018, Hossain, Hosseinzadeh, Chanda & Wang 2019). Although these methods have largely advanced the counting performances compared to those merely based on plain CNNs, the multi-scale feature fusion scheme itself will inevitably bring in disturbances from adjacent scales to features of objects in one certain scale, which will influence the accuracy of pixel-wise density values for objects with different scales. Experimental results (Sindagi & Patel 2018, Zhao, Zhang, Zhang & Zhang 2018) have demonstrated significant estimation errors in the background as well as inaccurate density estimation when measuring crowd count in local regions. We have noticed that in traditional counting approaches (Ryan, Denman, Fookes & Sridharan 2009, Lempitsky & Zisserman 2010, Chen et al. 2012),

a step of perspective normalization is widely adopted to compensate the feature disparities caused by the scale variations before the features are fed in to the regressor for density estimation. However, most CNN-based methods follow multi-scale feature fusion to address the scale variation problem approximately yet barely consider the perspective distortion that directly leads to the scale variations in their models, which may affect the accuracy of the scale-aware density estimation.

1.3 Research Contribution

Motivated by the above observation, we hypothesize that can we extend and leverage the perspective handling mechanism in traditional methods within the deep architectures? This will simultaneously benefit from the powerful feature representations as well as effective handling of problems related to scale variations for the crowd counting task. In this dissertation, we mainly study high-density crowd counting with perspective understanding in deep models. We provide a comprehensive analysis on how to enable perspective handling with the powerful CNN models with three strategies and examine their efficiency with extensive experiments on popular counting benchmark datasets. In detail, we highlight the main contribution of our work as follows:

- Considering that the perspective distortion has not been explicitly handled in most existing work, in the first part of this dissertation, we propose to generate scale-aware feature representations for scale-aware density estimations, with the geometric information of depth maps. We specifically design a depth-embedding module to inject the depths into the base CNN model to recalibrate the original features to be depth-aware (scale-aware), which are propagated into the next stages for scale-aware crowd density estimation. Experimental results demonstrate the effectiveness of explicitly incorporating geometric priors into the deep neural network for crowd counting.

- Having validated the effectiveness of explicitly incorporating side infor-

mation into the network, we are still curious about another question: without the injected geometric priors how can a network improve its understanding towards the scale variations and the underlying geometrics? Starting from the observation of local inconsistency problem of predicted crowd density maps with the ground-truth density maps, in the second part of the dissertation, we explore the multi-stage network for crowd counting for iterative refinement and error correction, in an attempt to address the aforementioned problem that is closely related to drastic scale variations. Besides, a novel region-based loss function is also proposed to drive the consistency of both global and local counts with the ground-truth. Experiments are conducted to validate the benefits of the implicit scheme with the constrained multi-stage model for crowd counting.

- Since it is both beneficial to either explicitly inject geometric information or implicitly exploit informative architectures, we are wondering if a plain CNN model without specific designs can be entitled with the ability to perceive the scale variations. Towards this end, in the third part of this dissertation, we resort to the compound factors existing in the predicted density maps and leverage these auxiliary attributes to regularize the representation learning of the network to generate features with desired attributes. Specifically, each attribute is formulated as an auxiliary task and a multi-task framework is constructed to facilitate the learning of the network, with the flexibility to keep the original model unchanged at inference. Experimental results have indicated the efficacy of the multi-task framework to improve crowd counting performances.

1.4 Thesis Structure

In the following of the thesis, we will first review the related work to visual crowd counting in section 2. In section 3, we introduce the method of a

depth-embedding module to explicitly incorporate geometric information into the networks to benefit crowd counting. Section 4 illustrates the method of a multi-stage architecture with regional supervisions to handle the local inconsistency problem in existing counting methods. Section 5 describes a multi-task framework to drive the intermediate feature to embed desired geometric/semantic/numeric attributes for more accurate density estimation and counting, and in section 6 we concluded the thesis with possible future directions.

Chapter 2

Literature Review

In general, existing crowd counting algorithms can be mainly categorized into three groups based on different schemes: counting by detection, counting by clustering and counting by regression. In this chapter, we will review the representative work belonging to each of the three paradigms. For the regression-based methods which have proved to be effective for crowded scenes, we will particular provide an overview of this paradigm on each of its key component and talk about both the traditional hand-crafted feature based methods and recent CNN-based methods to fully trace the development of regression-based counting approaches.

2.1 Counting by Detection

Getting a final counting number in an image will be a natural thing given all the detections of individuals presented in the image, with obtaining the precision locations of pedestrians as another reward. Thus the detection-based counting methods is the most intuitive and direct approach to enumerate the number of people in a scene (Loy et al. 2013), which mainly try to detect each individual pedestrian in images based on the statistical characteristics of either the monolithic or part of the human body, and then calculate the total numbers.

A typical approach for pedestrian detection is based on the statistical full-body appearance extracted from a set of training images with pedestrians. The HOG (histogram of oriented gradients) descriptor (Dalal & Triggs 2005) is one of the most powerful and effective hand-crafted features to describe the characteristics of upright pedestrians. Other features such as the Haar wavelets (Viola & Jones 2004), edgelet (Wu & Nevatia 2005) and their combinations are also commonly used. With extracted features, a classifier such as Support Vector Machines (SVM)) is then trained and at inference applied in a sliding window fashion across the whole image to detect pedestrian candidates. Non-maximum suppression (NMS) is generally appended as the last step to clean and merge the candidate regions denoting possible pedestrian locations, where the total counting results will be reported based on the final detections. Considering the potential detection errors that might be caused by the traditional detector, an alternative scheme (Topkaya et al. 2014) is proposed to further cluster and refine the detection outputs to calculate the final counts more accurately, instead of directly counting the coarse detections. For each frame, the traditional HOG detector runs and the detection areas can be initialized based on the raw detection outputs. Then the techniques of Dirichlet Process Mixture Models and Gibbs sampling are used to cluster all the detections based on the spatial, color and temporal features extracted from these detection areas. Finally, the total number of people in the scene will be estimated based on the number of people within each cluster. Although this type of methods can generate reliable results in constrained situations, the accuracy of detection drops dramatically when it comes to crowded scenes, where people are partially occluded and their body parts are nearly invisible (Dollar et al. 2012).

It is also found that in some situations when the human body has pretty large deformations, even though the pedestrians are not heavily occluded in these scenes, it is still pretty difficult for the traditional detectors that employ the ridge templates to correctly localize the people, i.e., the whole-body appearance that captures the outlier features of the upright people becomes

no longer reliable. Towards this problem, the part based model is proposed to bridge this gap for deformed people detection (Felzenszwalb, Girshick, McAllester & Ramanan 2010). Based on the assumption that the human body can be modeled by the combination of parts including the head, breast, and legs in a deformable configuration, a coarse global template covering the entire object and higher resolution part templates are both incorporated in the detection models, and then a discriminative learning scheme is proposed to detect the body parts and finally localize each individual people for counting. Inspired by the observation that the shape of head-shoulder part of people is pretty unique in natural images, the head-shoulder part detection is further proposed for better localization of each individual people (Lin, Chen & Chao 2001). In this work, the features of the head-shoulder parts are extracted using the Haar wavelet transform (Viola & Jones 2004). Then the SVM is employed to classify each featured areas as the head-shoulder contour or not. Finally, the perspective transformation technique is used to estimate the crowd size more accurately.

Despite the 2D information presented in the image for detection-based counting, motion is another important cue for human perception. The motion information is usually integrated into the sampling-based methods (Ge & Collins 2009*a*) to detect pedestrians for crowd counting, where a crowd scene is viewed as a realization of a stochastic process that consists of a random set of people in a bounded region. Each person is associated with a random 'mark' that governs their location and size in the image. Different sampling methods are applied to sample a person hypotheses from an underlying stochastic process and evaluate them against the image observation to find the optimal configuration that best explains the image, where the number of people in the scene and their locations will be automatically inferred. For example, Zhao *et al.* (Zhao, Nevatia & Wu 2008) propose a 3D human shape model, which describe humans by the connection of three 3D ellipsoids, to detect and track people in the crowd. The sampling method of data-driven Markov chain Monte Carlo (DDMCMC) is used to relate the

possible human locations to the image observations of appearance of humans, visibility of body and foreground/background separation. Ge *et al.* (Ge & Collins 2009*b*) inherit and extend this idea however with more flexible and practical shape models (Bernoulli shape masks) to detect and count people in crowded scenes. Although part-based and shape-based methods can mitigate the issues of occlusion to some extent, these methods were still error-prone in situations with extremely severe occlusions and complex background clutters.

Unlike above methods that use the hand-crafted features to describe the targets, the deep learning technique is more efficient and effective on fully exploit the low-level and high-level information automatically from the original images, and thus is more powerful at representing the characteristic of the images. A lot of research has been done on pedestrian detection by taking advantage of the deep neural network (Ren, He, Girshick & Sun 2015). For example, Ouyang & Wang (Ouyang & Wang 2013) explore the compensation ability to handle different important factors that influence the performance of pedestrian detection. Four key components including the feature extraction, deformation, occlusion handling and classification are jointly learned using a unified deep learning framework. With the aid of deep learning, the performance of pedestrian detection has been significantly advanced compared to the hand-crafted feature based methods. However, for scenarios of dense crowds, even the CNN-based detector is hard to generate reliable results (Liu, Gao, Meng & Hauptmann 2018), especially under the farther positions where the pedestrians have very limited pixels.

2.2 Counting by Clustering

Similar to the detection-based methods, counting-by-clustering approaches also rely on the delineation of individuals to perform the counting task. It employs the temporal information of the video and analyzes the trajectories in consecutive frames to detect each people, which is based on the fact that each people in the video has unique trajectory due to the personal preserva-

tion of spaces in the crowd. Then enumerating all the trajectories will output the counting of people.

In (Brostow & Cipolla 2006), an unsupervised data-driven Bayesian clustering algorithm is proposed to delineate individual entities. The basic idea is to track the image feature points and probabilistically group these low-level descriptors into clusters which can represent the independently moving objects. However, this system can fail if strong arm movements present with ridge motion scenes (Saleh, Suandi & Ibrahim 2015). Another paper that counts the number of people based on the trajectories employs an alternate tracking scheme (Sidla, Lypetskyy, Brandle & Seer 2006). The author incorporates the head-shoulder detection into each frame and extracts the region of interest (ROI) based on the ω-like shape regions. Then the feature descriptions are calculated to identify individuals at each frame. The Kalman Filter is used to form the complete trajectory and the total count is obtained using a gateway and a simple trajectory-based heuristic to eliminate the multiple counts for an individual who is associated with multiple trajectories.

Deep learning techniques have also been employed in the visual object tracking task. Ma et al. (Ma, Yang, Zhang & Yang 2015) exploit features extracted from the deep convolutional neural networks to improve the tracking accuracy. The characteristics of features extracted from different layers of the network are analyzed and hierarchically combined to handle the deformation, occlusion and finally boost the performance. The clustering-based counting methods mainly depend on the clustering of individual motions. As efficient as in the videos, some static people may be missed and the total count may be under-estimated. Besides, it is also hard to handle situations where multiple individuals share one trajectory.

2.3 Counting by Regression

As an alternative to the idea of individually delineate each pedestrians, the counting-by-regression methods relate the total counts to the representative

features of the crowd, avoiding the hard task to localize every object directly. Usually a mapping relationship is learned based on several training samples, and the total counts of new testing images can then be easily estimated once the crowd features are extracted. For crowded scenes with severe occlusions and large appearance variation of people, the counting by regression approach will be more robust compared to detection based approaches (Saleh et al. 2015). Initial regression-based methods usually estimate the count number directly, whereas later methods shift the main objective to crowd's density and model the count number as a natural by-product of the estimated density. In the following subsections, these two main categories of counting methods: direct regression and density-based regression, as well as the crucial factors that influence their performances will be introduced in detail.

2.3.1 Direct Regression

Background subtraction: The counting approaches based on direct regression usually involve the pre-processing of foreground segmentation. Typically, the video segmentation algorithms that consider the temporal motion information are employed to subtract the background pixels (Koprinska & Carrato 2001). Various probabilistic background models have also been proposed for the background, among which the mixture Gaussian background modeling technique could be the most popular one (Stauffer & Grimson 1999). Each pixel in the image is modeled as a mixture of Gaussians and an on-line approximation is used to update the model. The Gaussian distributions are then determined if they are the result from a background process. Finally, each pixel is classified according to its representation of the Gaussian distribution. This model works pretty well for stable environments and has demonstrated its effectiveness during these years. However, it cannot handle the static object and may also derive unreliable results when the background is complex and cluttered.

After the foreground segmentation, the basic pipeline of the direct regression based counting include the extraction of local or global features from the

foreground pixels, and then the regression from features to the corresponding counts. In terms the granularity of the extracted features, the *global regression* and *local regression* methods emerge.

Global regression: Inspired by the fact that images with different density levels exhibit different texture patterns, Marana *et al.* propose to estimate the crowd density based on the texture features extracted from the images (Marana, Cavenaghi, Ulson & Drumond 2005). For images with different crowded levels, textures are extracted and labeled to denote the five levels from very coarse to very fine. For a test image, the extracted texture will be classified using a self-organization-map (SOM) neural network based on the pre-labeled textures. Gong *et al.* propose a viewpoint-invariant-learning based method to count people in crowds (Kong, Gray & Tao 2006). For each blob, the edge orientation and blob size histograms are employed to describe one segmented blob. To cope with the perspective distortion, features are normalized with different perspective factors. The relationship between the features and the counts is learned from the labeled training data. In the above methods, the global feature is extracted from one image and used to regress the counts or density level. This is a pretty coarse representation of the crowd features because each entire one image is regarded as only one training sample and no detailed spatial configurations are coded during this process. This will lead to a great demand for training images to provide enough training samples, and also cannot fully exploit the spatial information of the images.

Local regression: In contrast to the the global regression algorithms, local regression is proposed to estimate people number in spatially localized regions (Chen et al. 2012). This work makes a trade-off between the one global regressor for each image and multiple local regressors for multiple local regions in one image. Specifically, it learns only one regressor for each image. However, this regressor is built based on multiple locally spatial regions. Thus the local information can be effectively shared and captured. The total counts will be the summation of all the estimated numbers of the

local regions calculated by the regressor. Similar to the counting-by-detection approach that involves the usage of spatial and temporal information, there are also work that employs the temporal information in the counting by regression approach. Chan *et al.* (Chan, Liang & Vasconcelos 2008) propose to obtain local segmentation regions through the employment of the temporal information in the video and learn the corresponding regression relationships based on these regions. First the crowd is segmented into several components according to their homogeneous motion, then a feature set is extracted from each segmented region, and finally a mapping function is learned using the Gaussian Process regression with the ground truth number in the certain component. Compared to the global regression methods, the local regression algorithms, which usually divide one image into sub-regions and then learn the regression model based on these subsamples, are more helpful with the consideration of the locally spatial information.

2.3.2 Density-based Regression

For either the global or local regression based methods, most of the research utilize the features extracted from the foreground blobs to directly learn the regression relationship, which triggers the employment of various background subtraction techniques. The clear detachment of foreground crowd pixels from the background plays a basic role for further modeling process of the foreground features and their counts. However, severe segmentation errors may occur due to the significant variations of foreground appearances and the cluttered background, thus deriving unreliable counting results. To avoid bringing in the fatal errors caused by segmentation, density-based method (Lempitsky & Zisserman 2010) is proposed to handle this problem. In this method, each pixel will be assigned a density value indicating how many people one pixel represent, according to the features extracted from the pixel, and the total count of the image will be the integral (sum) of the density values of all the pixels. Without directly denoting the foreground pixels, the density-based regression approach avoids the hard task of fore-

ground segmentation and thus is more robust against situations with complex background. Besides, pixel-wise densities also enable the counting task for arbitrary ROI regions, not only the whole image.

Following this research work, Fiaschi (Fiaschi et al. 2012) propose to further simplification scheme by estimating the object density map over patch-wise predictions with a regression random forest. A mapping is learned between patches in the input feature space and the target object density space, which leads to a simpler method with on-par performances and fewer parameters. More recently, an interactive object counting system is proposed based on the innovation of density estimation, to count numbers of objects in images based on multiple user inputs (Arteta, Lempitsky, Noble & Zisserman 2014). This makes it possible for users to specifically count the desired category of an object in the images, and removes the limitation that all the objects in the images should be of the same type. A feature vocabulary is learned and updated progressively along with the user provides more annotations. The ridge regression is then used in one interaction process to learn the mapping relationship based on the current feature vocabulary and the labeled data. Interestingly, the algorithm also provides two visualization methods to present the counting results for the user to decide if further annotations are needed. However, these density-based methods are all based on the hand-crafted feature, which limits their capacity to dense crowds.

2.4 Counting with Deep Neural Network

In recent years, the prevalence of deep learning techniques (Krizhevsky et al. 2012) has triggered a flurry of work exploring deep architectures for various computer vision tasks. As a result, the deep neural network (DNN) has also been introduced into the area of crowd counting and several DNN-based methods have been proposed to boost the counting accuracy. A brief introduction to the deep neural network will be presented, followed by a review of the DNN-based counting methods.

2.4.1 Convolutional Neural Network

As a specialized kind of neural network, Convolutional Neural Networks (ConvNets or CNNs) (LeCun, Boser, Denker, Henderson, Howard, Hubbard & Jackel 1989, LeCun, Boser, Denker, Henderson, Howard, Hubbard & Jackel 1990, LeCun, Bottou, Bengio, Haffner et al. 1998) have achieved tremendous success in the computer vision society for several real-world applications including image classification (Krizhevsky et al. 2012, Sermanet, Eigen, Zhang, Mathieu, Fergus & LeCun 2014), object detection and localization (Girshick, Donahue, Darrell & Malik 2014, Girshick 2015, Ren et al. 2015), pose estimation (Tompson, Jain, LeCun & Bregler 2014, Schwarz, Schulz & Behnke 2015, Carreira, Agrawal, Fragkiadaki & Malik 2016) and semantic segmentation (Long, Shelhamer & Darrell 2015, Noh, Hong & Han 2015, Chen, Papandreou, Kokkinos, Murphy & Yuille 2017). Beyond the computer vision, CNN has also significantly benefited other research areas dealing with data processing including natural language processing (Kalchbrenner & Blunsom 2013, Bahdanau, Cho & Bengio 2015), speech recognition (Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Kingsbury et al. 2012, Graves, Mohamed & Hinton 2013) and recommendation system (Huang, He, Gao, Deng, Acero & Heck 2013, Elkahky, Song & He 2015). The remarkable success of the convolutional networks lies in the usage of the neuroscientific principles in their design policy (Goodfellow, Bengio & Courville 2016), rendering the network extremely suitable to extract hierarchical and representational features from images.

A basic convolution network is usually composed of a series of convolution layers with learnable parameters, followed by non-linearities between the parameterized layers such as using the rectified linear activation function. Pooling layers can also be added to aggregate information from nearby regions to abstract the low-level and high-level representation at different phrases. With a two-dimensional image I as input, each convolution layer usually applies a weighted average operation with a two-dimensional *kernel* w at each spatial position of the input and generates an output which is

often referred to as the *feature map* S. Mathematically, the operation can be written as: $S(i,j) = w * I(i,j) = I(i-m, j-n)w(m,n)$. In contrast to the traditional neural network layers that define a separate interaction between each input unit and each output unit, the convolution layer with a small kernel enables sparse interactions of the convolution network, rendering it extremely efficient for image processing. An input image may contain millions of pixels, however with the small kernel important local features like edges and textures can be detected by sliding the kernel across the image, which dramatically saves the parameters (Goodfellow et al. 2016). A pooling function statistical summarize the output of the network at a certain location based on its nearby outputs. Take the *max-pooling* (Zhou & Chellappa 1988) as an example, it selects the maximum output within a rectangular neighborhood. Other popular pooling operations include the *average-pooling* and *weighted-pooling*. The pooling operation helps to preserve the translation invariance of the representations.

A typical convolution network for classification is usually ended up with a fully connected layer followed by a softmax scoring layer to output the probability of the input belonging to each pre-defined class (Krizhevsky et al. 2012). This is the general configuration for most classification-based tasks; however, for tasks that require structured output with a class label for every pixel, such networks will lack efficiency because the output plane may be smaller than the input plane with the pooling operations. For pixel-wise labeling, one strategy is to produce an initial guess of the image labels, which is further refined using interactions between neighborhood pixels. For example, Pinheiro *et al.* (Pinheiro & Collobert 2014) propose to use a recurrent convolutional network to mimic the processing on a large input context with shared parameters. The system can repeatedly identify and correct its errors from each stage with low inference cost. Similarly, Zheng *et al.* (Zheng, Jayasumana, Romera-Paredes, Vineet, Su, Du, Huang & Torr 2015) combine the convolutional networks with the graphical model of conditional random fields (CRFs) for scene labeling, the latter of which is formulated as recur-

rent networks. In this way, the post-processing methods are integrated into the whole model and trained end-to-end, avoiding the original off-line processing for object delineation. In contrast to previous methods, Long *et al.* (Long et al. 2015) later propose a fully convolutional network (FCN) for semantic segmentation, which does not rely on any pre- or post-processing complications. With in-network upsampling and multi-layer combinations, the proposed FCN is able to accept arbitrary-sized inputs and output a correspondingly-sized output. The effectiveness of the FCN has motivated its applications in several other areas with structured outputs. It has also been introduced into the crowd counting to help the generation of density map, which is the basic architecture for most existing CNN-based counting methods.

2.4.2 Recurrent Neural Networks and Long Short-term Memory Networks

To model the sequential relationships, recurrent neural networks (RNNs) have been proposed and successfully applied to various sequential modeling problems (Goodfellow et al. 2016). RNN is a class of network which computes the output at each time stamp based on previous outputs. Unlike traditional networks which have separate parameters for each input feature and do not specifically link future outputs with current predictions, recurrent networks benefit from the idea of parameter sharing across different parts of the model to enable the modeling of examples in different forms and the generalization across them. Formally, given an input sequence $\mathbf{x} = (x_1, \cdots, x_T)$, a standard recurrent neural network (Graves et al. 2013) computes the hidden vector sequence $\mathbf{h} = (h_1, \cdots, h_T)$ and output vector sequence $\mathbf{y} = (y_1, \cdots, y_T)$ by iterating the following equations from $t = 1$ to T:

$$h_t = \mathcal{H}(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \tag{2.1}$$

$$y_t = W_{hy}h_t + b_y \tag{2.2}$$

where W represents weight matrix (e.g., W_{xh} is the hidden weight matrix for input x_t at time stamp t) , b denotes bias vector for the hidden layer and $\mathcal{H}$ denotes the operation used by the hidden layer which usually is nonlinearity such as tanh or rectified linear units (ReLU).

In practice, the vanilla RNN has some difficulties in modeling long sequences (Bengio, Simard & Frasconi 1994). To mitigate this problem, the long Short-term memory (LSTM) architecture is introduced. Instead of a simple elementwise application of a sigmoid function in the original processing of $\mathcal{H}$ 2.1, LSTM networks exploit unique cells with more parameters and a system of gating units to implement a composition processing (self-loop) of $\mathcal{H}$, which relies on the self-loops to produce paths that allow the gradient flow for long durations (Goodfellow et al. 2016). With its ability to learn on data with long-range temporal dependencies, currently the LSMT networks have been mostly exploited in practical used to replace the vanilla RNN and have been found extremely successful for many sequential modeling tasks such as handwriting recognition (Graves, Liwicki, Fernández, Bertolami, Bunke & Schmidhuber 2009), machine translation (Bahdanau et al. 2015) and image caption (Xu, Ba, Kiros, Cho, Courville, Salakhudinov, Zemel & Bengio 2015).

2.4.3 DNN-based Crowd Counting

Early works mostly rely on fully-connected networks to either directly estimate the count for input patches. For example, Wang *et al.* (Wang et al. 2015) first propose to use a CNN-based deep model to automatically learn effective features for counting. The data augmentation with negative sample expansion has also been proposed to enrich the training data and thus enhance the robustness of the trained model. In contrast to the fully-connected networks that distort the spatial distribution of features in the original image space, the "fully convolutional" architecture (Long et al. 2015) has largely promoted the pixel-labeling tasks, which enables the model to accept arbitrary-sized input and generate correspondingly-sized output. To embed the density-based counting approach into the deep architectures, Boom-

inathan *et al.* (Boominathan et al. 2016) introduce the fully convolutional network (FCN) to predict a density map given an image. More specifically, they design a combination of deep and shallow FCN to capture both the high-level and low-level features that necessitate the counting under large scale variations. This can be viewed as the first work that attempts to handle the drastic scale variations under the deep learning framework.

Among the different factors that influence the counting accuracy, handling of the intra-image scale variations caused by the perspective distortion has been drawing extensive attention of researchers due to its extremely challenging situations (Sindagi & Patel 2018). Considering that the scale variation is mainly caused by the perspective distortion in the surveillance scenes, Zhang *et al.* (Zhang et al. 2015) at the very initial propose to extract candidate training patches from original images at a size proportional to the corresponding perspective values, which are then resized into a fixed size to train the CNN for patch-wise density estimation. In this way, scale variations of pedestrians are alleviated outside the deep model. Later, Zhang *et al.* (Zhang et al. 2016) introduce a multi-column neural network (MCNN) with a different receptive field sizes in each column towards different object scales. The resulted multi-scale features are aggregated with a 1×1 convolution kernel for density map estimation. Compared to the usage with a plain CNN in previous methods (Wang et al. 2015, Zhang et al. 2015), this scale-ensemble mechanism has significantly improved the results and has been widely adopted in several following work. For example, Daniel *et al.* (Onoro-Rubio & López-Sastre 2016) propose to input a pyramid of image patches to the CNN to generate multi-scale features and fuse them for density map prediction. Similarly, Zhang *et al.* (Zhang et al. 2018) employ features from layers at different depth as multi-scale representations, which take advantage of the hierarchy of CNN architectures. Other strategies that focus on constructing the multi-scale features within the CNN have also explored. Aka but not identical to (Zhang et al. 2016), Cao *et al.* (Cao, Wang, Zhao & Su 2018) propose a scale-aggregation module to aggregate features

from the previous layer with several sub-paths, each composed of operation with different receptive field, to generate multi-scale representations. This ensures the preservation of multi-scale property at each layer along with the propagation of the features through the whole model. Skip connections are also considered as an effective way to combine the hierarchical and multi receptive filed information across different layers. In (Oñoro-Rubio, Niepert & López-Sastre 2018) a gated network is proposed which learns the skip connection with a soft weight as the gate to control the message-passing between two layers. Instead of a hard connection in the skip-layer, the information flow can be optimized towards the learning of objective. Besides, Shen *et al.* (Shen, Xu, Ni, Wang, Hu & Yang 2018) propose to address the scale variation via a scale-consistency regularizer which enforces the summation of crowd counts from local patches equal to the total counts in their region union. Two networks are separately built for patch-level and image-level density estimation, respectively. Then these two models are interlaced with a cross-scale Consistency Pursuit Loss added as a regularization except the $L2$-norm loss of their own. Liu *et al.* (Liu, Salzmann & Fua 2019) propose to exploit the context information related to scales to handle the perspective distortion problem. In their model, features are extracted from paths with multiple receptive field sizes and re-weighted according to their importance at each image location. In this way, the contextual information of scale that is best suited in each location is adaptively encoded for more accurate density estimation.

Except for those methods that focus on the construction of multi-scale features to handle the scale variations of pedestrians, other methods gradually drifted to another direction: how to effectively fuse the multi-scale information to further benefit the counting results? The adaptive fusion mechanism emerged considering that at each location, the features corresponding to different scales should have different contribution towards more accurate density estimation, other than being treated uniformly across both locations and scales. Hossain *et al.* (Hossain et al. 2019) propose a scale-

aware attention network which learns to automatically generate the weights for features corresponding to the global and local scales for fusion. They employ two additional networks which complement the main network with their extracted global and local information, respectively. Their adaptive fusion can be seen as across multiple feature scales, however keeps uniform across spatial locations on the features. Kang *et al.* (Kang & Chan 2018) further propose to adaptively fuse the predictions from an image pyramid with adaption across both scales and locations. An attention model is appended to generate an attention map for predictions at each scale, which is then multiplied to the corresponding prediction for rectification. A 1×1 convolution fuses the rectified density map from all the scales for the final crowd density map. Aka but not identical to (Kang & Chan 2018), in (Varior, Shuai, Tighe & Modolo 2019) the author propose an attention model to generate attention maps to adaptively fuse the features extracted from shallow and deep layers of a CNN, which is treated as multi-scale features. Deepak *et al.* (Sam et al. 2017) further push the adaptive fusion mechanism into an extreme, arguing that for corresponding locations only the features at one scale is most suitable and interference from multi-scale features will inevitably result to the inaccuracy of density estimation. Towards this problem, they propose a model which decides the usage of feature form only one best scale for density estimation in different locations. In detail, they crop 3×3 image patches and relay them to different regression models assigned by a "switch" network to enable scale-specific processing and density estimation. Although these methods have demonstrated effectiveness, their ability to handle scale variation is limited either by the number of columns used in the network (Sindagi & Patel 2018) or by the levels of the pyramid of input images that generate multi-scale features.

Considering the fact that scale variation is closely related to the scene's perspective information, the side information has been incorporated in an attempt to generate perspective-aware (i.e., scale-aware) density estimations. Usually a perspective map for a scene is manually labeled based on the mea-

surements of the image height of each pedestrian at different depth of the scene. In traditional hand-crafted methods, the perspective normalization is a necessary step before the features are fed into the regression model, where the perspective map (weights) will be applied to the features to correct the huge feature disparities extracted at different locations. When it comes to the deep learning era, Kang *et al.* (Kang, Dhar & Chan 2017) propose an adaptive convolutional network whose filter weights are adaptively derived from another sub-network with the side information of perspective map as the input. In this way, the scale variations related to the perspective map can be disentangled and cooperated via the learned filter weights. Recently, Shi *et al.* (Shi, Yang, Xu & Chen n.d.) inherit the traditional methods of perspective normalization and extend it into the CNN model. Their model learns to predict the perspective map for each image. The estimated perspective map is treated as a confidence map composing the information on scale variations, which is further encoded and applied to the predictions on the top of convolution layers at different depths.

Beyond addressing the scale variation problem, researchers also investigate other possible directions that could be exploited to improve the crowd counting accuracy. One direction is the combination of regression- with detection-based methods. Liu *et al.* (Liu, Gao, Meng & Hauptmann 2018) propose to take advantage of both the regression-based and detection-based counting approaches for a mutual complement. Their model adaptively decides the appropriate counting mode for different locations on the image with an attention mechanism assigning the reliability of the estimations separately from these two modes. The final crowd count is a weighted combination of the two results. There are also researchers exploring the data augmentation problem for crowd counting. Considering that manually labeling is time-consuming and labor-intensive, Liu *et a.l* (Liu, van de Weijer & Bagdanov 2018) resort to the vast unlabeled data on the Internet and exploit these images to improve the training of a counting network by ranking. With the observation that image patches sampled from an image con-

tain less (or equal) people than the original image, they propose a ranking loss to regularize the learning of the network with unlabeled data as input. Promising results are obtained with the proposed method. Similarly, Wang *et al.* (Wang, Gao, Lin & Yuan 2019) propose to augment the model training with synthetic data for crowd images. They adopt an external software to synthesize a number of crowd images, whose ground-truth annotations are easily obtained without additional labeling cost. They have proved the effectiveness of pre-training counting models with synthetic data with extensive experiments.

Other researchers have also explore the video-based crowd counting with temporal information based on the recurrent neural networks. Most of existing work focus on the exploitation of the temporal information preserved in the video data to better localize the crowd regions from the cluttered background. Zhang *et al.* (Zhang, Wu, Costeira & Moura 2017*b*) propose a deep spatio-temporal network to count vehicles from low quality videos. The FCN part of the network benefits the pixel-wise predictions and the LSTM part captures the temporal dynamics. To strengthen the perception of temporal correlations in the videos, Xiong *et al.* (Xiong, Shi & Yeung 2017) exploit a convolutional LSTM (ConvLSTM) network to fully capture the spatial and temporal dependencies. Recently in (Miao, Han, Gao & Zhang 2019), the author propose to employ a 3D convolution network in combination with the 2D convolution model to learn the spatial-temporal features.

Chapter 3

Scale-aware Crowd Counting via Depth-embedded Convolutional Neural Networks

As illustrated in section 1, although the perspective correction is a necessary step in traditionally hand-crafted feature based counting methods (Loy et al. 2013), it has not been explicitly explored in most existing CNN-based methods. Would it be beneficial to also employ the perspective-related information within the deep neural networks to further improve the performance of crowd counting? Armed with this hypothesis, in this chapter, we first explore the mechanism of incorporating scene geometric information into deep architectures and study its effects on the counting accuracy. We illustrate the effects of intra-image scale-variations on density values and propose to exploit the depth cues to generate scale-aware feature representations to improve the accuracy of density estimation. To this end, A depth embedding module is developed as an add-on to baseline networks, which processes the depth information and spatially re-calibrates the magnitude of individual features. With the proposed module, Depth Embedded Networks (*DeemNet*) is constructed based on the backbones of a deep and a shallow network, respectively. We compare the proposed model with several state-of-the-art

approaches to show the benefits of explicitly injecting perspective-related information for more accurate counting results.

3.1 Introduction

Due to magnifications and perspective related distortions (Loy et al. 2013), images depicting crowed scenes often contain people with drastic scale variations, posing great challenges for general counting systems that operate on uncalibrated camera systems. In Figure 3.1, we show a sample crowd image where objects closer to the camera appear significantly larger compared to the objects at farther distances. Roughly speaking, the scales of objects are inversely proportional to their distances to the camera imaging plane (Chan et al. 2008).

The popular density-based approaches (Lempitsky & Zisserman 2010, Fiaschi et al. 2012) generally determine crowd count by summation of the density values over specific regions in an estimated 2D density map. Following this paradigm, we postulate that a counting model should compensate for the object scale variations and work on *scale-aware* density values to achieve accurate estimates. As illustrated in Figure 3.1, three different regions in the crowd image (marked with black, orange, and red circles) contain the same number of pixels. However, due to perspective-related distortions, each region contains a different number of people, i.e., there are six, three, and one pedestrian in the farthest (black), medium (orange), and the nearest (red) circle, respectively. Since three regions have the same area, the density values at these three positions should vary accordingly to generate the correct estimates when we sum the density values over each region. More specifically, the density values in the farthest circle should be larger than those in the nearer regions. This suggests that a counting system should infer *scale-aware* density values and compensate for the scale variations caused by magnification or perspective related distortions.

To compensate feature disparities between varying-sized objects, classic

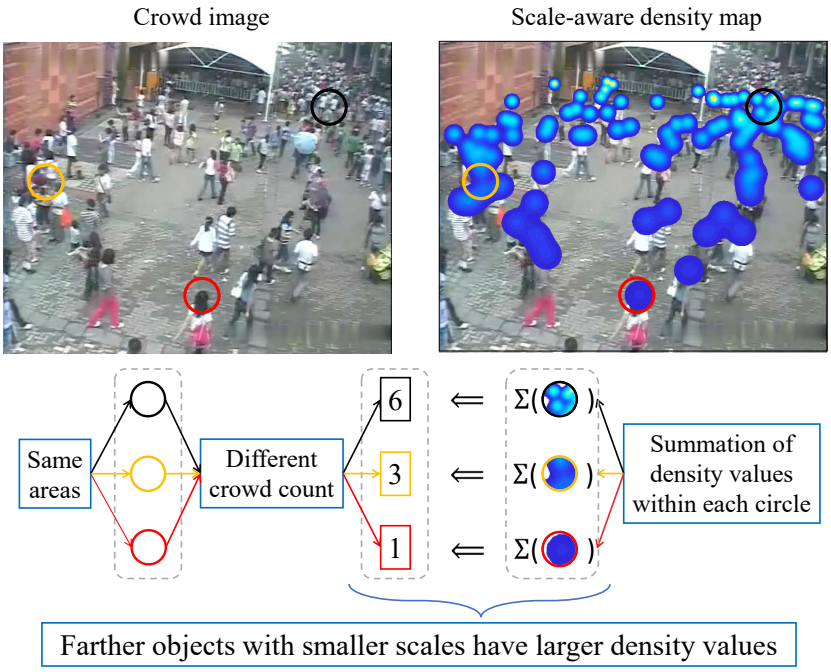

Figure 3.1: Our motivation (best viewed in color): Due to scale changes of pedestrians, the three regions (black, orange and red circles) that occupy the same number of pixels have different crowd counts; 6 in the far field (black), 3 in the midway (orange), and 1 in the near field (red) respectively. Since these three regions have the same area, the density values within the farthest circle should be larger than the ones in the nearer circles. In other words, objects with smaller scales should have larger density values and vice versa. This can be interpreted as *scale-aware* density values.

counting methods (Lempitsky & Zisserman 2010, Chan & Vasconcelos 2012, Sheng, Shen, Lin, Li, Yang & Sun n.d.) usually perform normalization to the feature based on perspective values before they are utilized for density regression, which is called perspective normalization (Chan et al. 2008). While the hand-crafted features have been surpassed by the CNN based methods, the mechanism of feature normalization with helpful side-information to handle scale variations was still useful. Can we also leverage and extend this processing with the power of deep learning? In this chapter, we propose to distill the underlying information from depth cues to generate scale-aware feature representations as well as the scale-aware density estimations. Due to the reason that generally the scale is inversely proportional to the object depth, we believe that with more understanding to the depth of a scene, the network will be better armed with the ability to perceive the scale variations across the image. Furthermore, considering that most existing counting datasets contain only single images, we infer depth results from a pre-trained single-image depth prediction model (Liu et al. 2016), which makes our method more applicable.

Specifically, we propose a depth embedding module which integrates the depth information and spatially re-calibrates the magnitude at individual feature map location to generate scale-aware representations. An encoding layer in the depth embedding module first encodes the depth image into the feature space. Although the encoded depth can provide geometric information it is blind to the whole scene, regardless of the semantic attribute at each location. To specifically emphasize on the attentive foreground objects and avoid the distraction from the background areas, a rectify layer follows to further refine the encoded depth map to generate scale-aware weights, which relies on the spatial attention mechanism to provide guidance information on attentive areas the targets are located in. Finally, an embedding layer applies the inferred weights to re-calibrate the magnitude of the original features at the individual location. Being regulated by the scale-aware weights, features will exhibit geometric diversities in scales among foreground objects

at different positions, which will directly benefit the estimation of scale-aware density values.

3.2 Approach

3.2.1 Overview

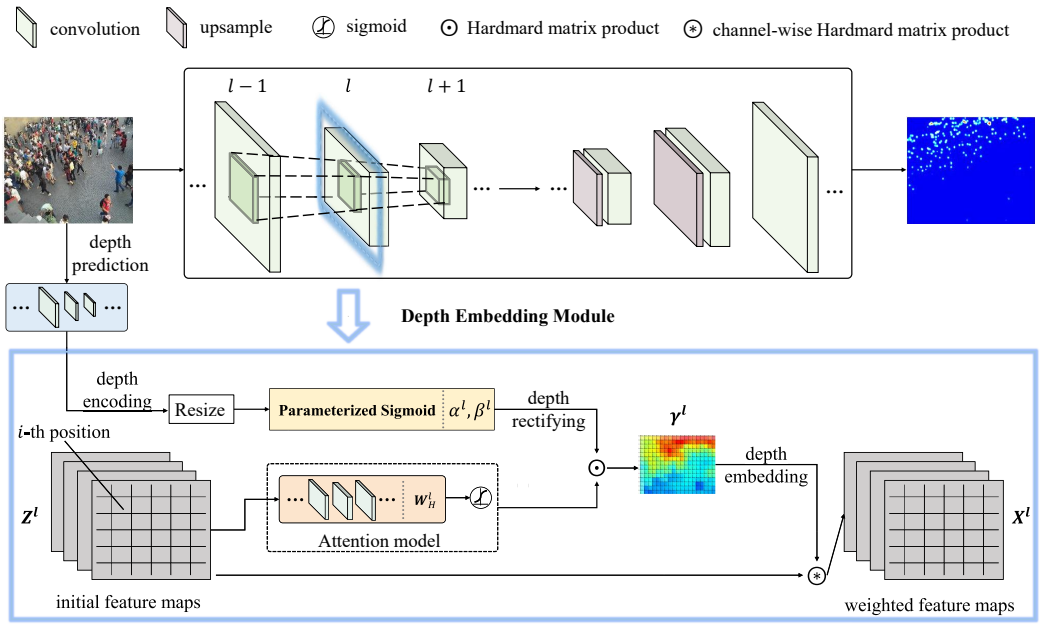

Figure 3.2: Overview of the proposed Deem-CNN. For the l-th layer in the CNN encoder, initial feature maps $\mathbf{Z}^l$ is the output of the previous $(l-1)$-th layer. We build a Depth Embedding Module on top, including a depth encoding layer, a depth rectifying layer and a depth embedding layer to capture essential geometric depth cues to predict attentive scale-aware scaling weights γ^l that are conditional on the feature maps and the predicted depth result. The learned weights re-calibrate the magnitude of features at individual location, results a weighted scale-aware feature map $\mathbf{X}^l$.

We adopt the popular encoder-decoder framework (Shen et al. 2018, Noh et al. 2015) for crowd density estimation, where a CNN encoder transforms an input image to high-level multi-layer feature maps and then a CNN de-

coder decodes the feature maps into a spatial density map. As illustrated in Figure 3.2, our depth embedded network (DeemNet) aims to modulate the original features to embed essential geometric attribute through a depth embedding module which produces scale-aware scaling factors for individual locations on the feature maps.

Formally, suppose for the input image $\mathbf{I}$ we have its depth image $\mathbf{D}$ at hand. At the l-th layer of the encoder, the scaling factors, dubbed as scale-aware weights γ^l, is a function of $\mathbf{D}$ and the current CNN features $\mathbf{X}^l$ at layer l. Thus, DeemNet re-calibrates current features $\mathbf{Z}^l$ using the scale-aware weights γ^l in a recurrent fashion as:

$$\begin{aligned} \mathbf{Z}^l &= \mathrm{CNN}(\mathbf{X}^{l-1}) \\ \gamma^l &= \mathcal{T}^l(\mathbf{Z}^l, \mathbf{D}) \\ \mathbf{X}^l &= f(\mathbf{Z}^l, \gamma^l), \end{aligned} \tag{3.1}$$

where $\mathbf{Z}^l$ is the output from previous conv layers in the CNN model, $\mathbf{D}$ is the predicted depth image using pretrained models (Section 3.2.2), $\mathcal{T}^l$ denotes the transformation function that generates the scale-aware weights in the depth embedding module (Section 3.2.3), $f(\cdot)$ denotes the weighting function that modulates CNN features with the generated weights (Section 3.2.3), and $\mathbf{X}^l$ is the weighted feature after re-calibration. The output features will be taken as input of the next layer and proceed until the decoder which maps the scale-aware representations into scale-aware density values.

3.2.2 Depth Prediction

As an object's scale is close to its distance from the camera, we exploit the depth cues of an image to help model the scale variations between objects at different locations. However, currently most existing counting benchmarks contain only single images. Inspired by the recent success of CNN-based depth prediction approaches, we resort to the work of Liu *et al.* (Liu et al. 2016) which learns a deep convolutional neural fields (DCNF) model for depth prediction. This depth predictor provides an indoor ver-

sion trained using NYU2 (Silberman, Hoiem, Kohli & Fergus 2012) dataset and an outdoor version trained using Make3D (Saxena, Sun & Ng 2009) dataset. In the experiments, we exploit the indoor version for the Mall (Chen et al. 2012) dataset of an indoor scene while the outdoor version for another three datasets (Zhang et al. 2015, Zhang et al. 2016, Idrees et al. 2013) with outdoor scenes. We apply this pre-trained DCNF model without any changes or finetuning on the counting scenes and achieve surprisingly reasonable results. Figure 3.3 visualizes the predicted depth maps for sampled crowd images. As observed, the predicted depth images can adapt to various scene layouts and confidentially depict the distance variations at different positions to the camera imaging plane.

Figure 3.3: Visualization of depth maps from the pre-trained DCNF model for depth prediction (Liu et al. 2016). The first row shows sample images from four crowd counting datasets (Zhang et al. 2015, Zhang et al. 2016, Idrees et al. 2013, Chen et al. 2012), respectively. The first three images all depict outdoor scenes while the last one is from an indoor scene. The second row visualizes the predicted depth map of each sample image.

3.2.3 Depth Embedding Module

As depicted in Figure 3.2, the depth embedding module mainly consists of three parts: depth encoding, rectifying and embedding. Each of these sub-modules will be described in detail in the following article.

Depth encoding Suppose for an input image $\mathbf{I}, \mathbf{I} \in \mathbb{R}^{H \times W \times 3}$, its depth result inferred from the depth prediction model (Liu et al. 2016) is $\mathbf{D}, \mathbf{D} \in \mathbb{R}^{H \times W}$. For the depth embedding module at layer l, the depth image is first resized to match the size of feature maps $\mathbf{Z}^l$ at the corresponding layer. To generate scale-aware features, larger weights need to be assigned to farther, smaller-scaled objects. Considering that the desired distance information has been readily available in the depth map, we then employ a non-linear encoding with the parameterized sigmoid function (Zhang & Woodland 2015) to normalize the depth values into $(0, 1)$:

$$u^l = g(\mathbf{D}) = \frac{1}{1 + e^{-\alpha^l \mathbf{D} + \beta^l}}, \tag{3.2}$$

where α^l and β^l are learnable parameters to tune the encoding function. This function is differentiable and hence it can be trained with the standard SGD algorithms. The partial derivatives of the objective function L with respect to the parameters α^l can be written according to the chain rule as:

$$\begin{aligned} \frac{\partial L}{\partial \alpha^l} &= \frac{\partial L}{\partial u^l} \frac{\partial u^l}{\partial \alpha^l} \\ &= \sum_j \frac{\partial L}{\partial u_j^l} \frac{\partial u_j^l}{\partial \alpha^l} \\ &= \sum_j \frac{\partial L}{\partial u_j^l} \mathbf{D}_j g(\mathbf{D}_j)(1 - g(\mathbf{D}_j)) \end{aligned} \tag{3.3}$$

where u_j^l and $\mathbf{D}_j$ are the j-th element of u^l and $\mathbf{D}$, and the objective function L will be described in Section 5.2.3. Similarly, the partial derivatives of the objective function L with respect to β^l can be written as:

$$\frac{\partial L}{\partial \beta^l} = -\sum_j \frac{\partial L}{\partial u_j^l} g(\mathbf{D}_j)(1 - g(\mathbf{D}_j)) \tag{3.4}$$

Depth rectification While the depth provides information on scale variation, it is blind to the whole scene and does not specifically differentiate between foreground objects and background. With this raw depth map, features at background areas will also be inevitably re-calibrated, which is undesirable and may disrupt the originally learned feature representations. For intuition, the features towards to the background sky at remote places (with larger depth values) will be assigned with very large weights upon the direct application of the initially encoded depth, which is irrelevant in the measurement of scale variations among target objects and also may introduce additional background noises. Towards more effective utilization of the predicted depth, we propose a rectification layer for depth refinement.

Intuitively, a prior information on the potential crowd regions would be beneficial. However, at hand we only have the label of dotted annotations of pedestrians, and it is expensive to label additional crowd segments. In contrast, we introduce the spatial attention mechanism (Xu et al. 2015) to tell where the foreground objects are located in with a soft attention mask v^l for the depth embedding module at layer l. This attention mask will act as guidance to selectively focus on the depth distinction among those targeted objects and de-emphasize the depth cues presented on the background areas. $v^l \in \mathbb{R}^{M \times N}$ can be written as a function of the feature maps $\mathbf{Z}^l \in \mathbb{R}^{M \times N \times C}$:

$$v^l = \text{sigmoid}(\Phi_s(\mathbf{Z}^l)) \tag{3.5}$$

where Φ_s represent a CNN based attention model which is composed of two convolution layers with kernel size of 3×3 (the first layer has 512 filters and the second layer has 1 filter). The attentive weights are further computed by element-wise sigmoid function on the output score map from Φ_s to highlight the most relevant regions across the whole spatial areas. In our case for crowd counting, it will learn to attend to the foreground pedestrian regions. Figure 3.4 visualizes some examples of learned attention masks. The second and the third columns respectively shows the results when the input feature maps are from different layers at increasing depths of the backbone model. It can be observed that the attention masks can effectively highlight the

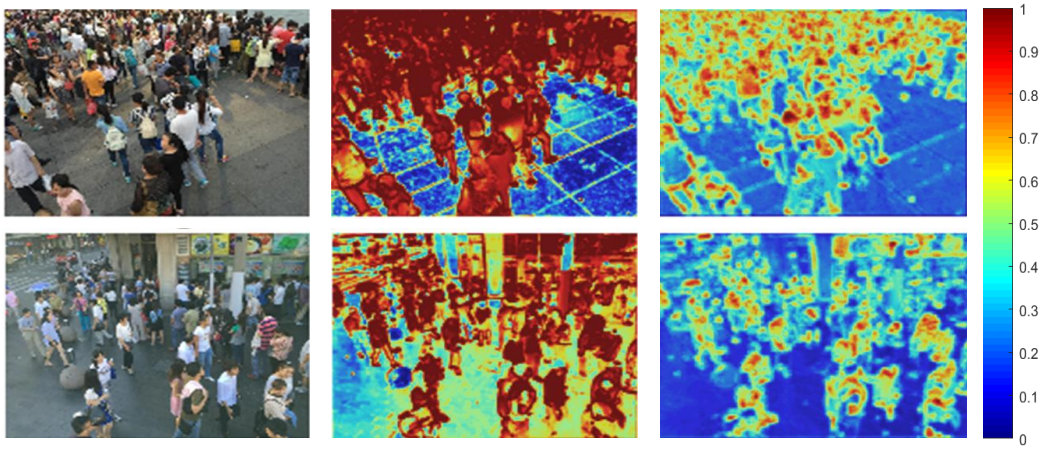

Figure 3.4: Visualization of attention masks. The first column shows two sample images. The second and the third column respectively visualizes the learned attention masks when the attention module is set at increasing depths of the backbone model. In all the heat maps from blue to red, the underlying value becomes larger.

foreground crowd areas from the background. It is also notable that with hierarchical feature representations enabled by the CNNs it is possible to generate attention masks at different semantic levels. As observed, attention masks at increasing depths concentrate on more abstract representations, i.e., from global crowd regions to isolated head locations.

Further, the encoded depth is rectified using the attention mask to obtain the attentively scale-aware weights γ^l:

$$\gamma^l = \mathcal{T}^l(\mathbf{Z}^l, \mathbf{D}) = v^l \odot u^l \tag{3.6}$$

where $\odot$ denotes the Hadamard matrix product operation ($(A \odot B)_{i,j} = (A)_{i,j}(B)_{i,j}$). With multiplicative combination, attention masks v^l will help rectify and suppress irrelevant signals in the background areas of the encoded depth u^l. Figure 3.5 shows the effects of the depth rectification layer. As observed, after rectification the background areas are de-emphasized however the depth disparities among the foreground objects are still preserved and highlighted, implying the effectiveness of the rectification towards attentive scale-aware weights.

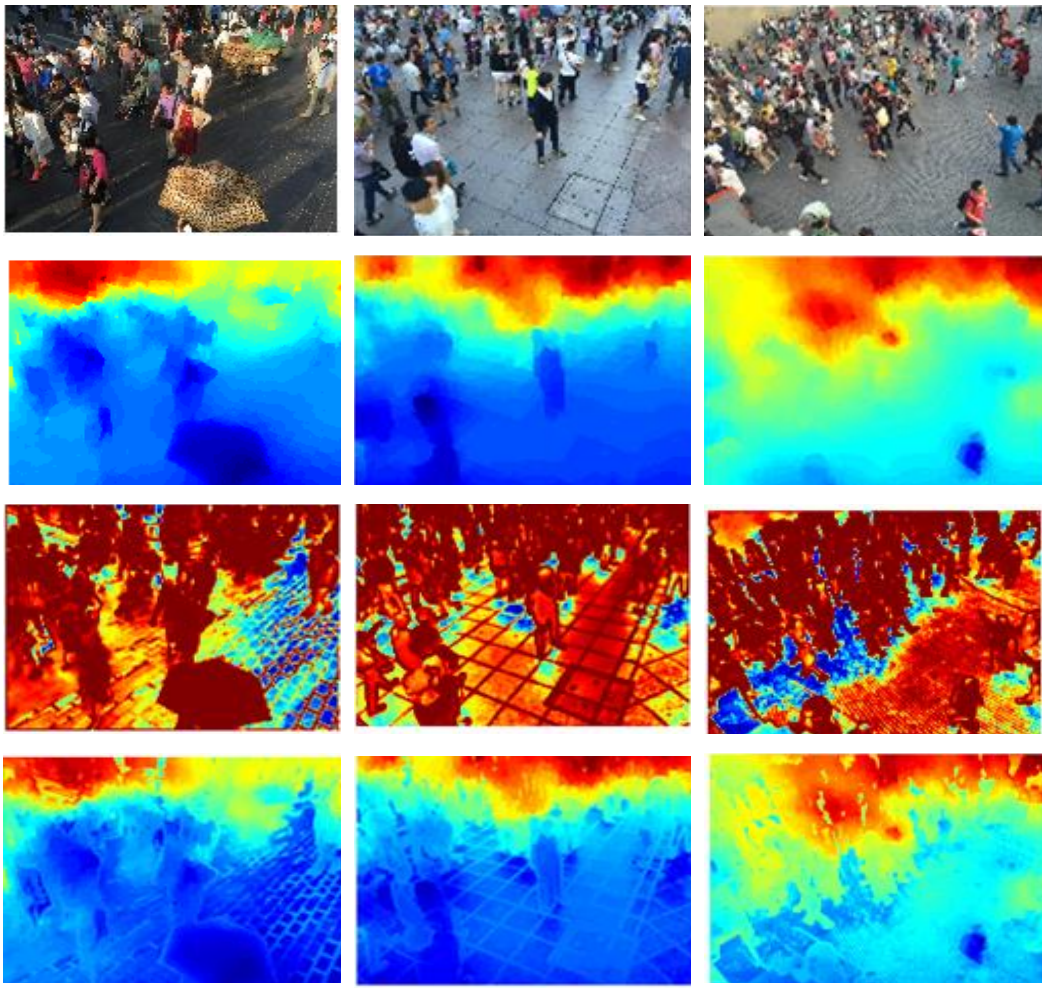

Figure 3.5: Visualization of the image (first row), attention mask (second row), the depth map shown in color (third row) and the generated attentive scale-aware weight maps after depth rectification (last row).

Depth embedding With the scale-aware weights, the original feature $\mathbf{Z}^l$ is tuned using a linear weighting function $f(\cdot)$ as a feedback loop. Different from the existing popular modulating strategy that aggregates features across spatial locations based on the generated weights, function $f(\cdot)$ applies element-wise multiplication. As a consequence, feature activations at different positions are re-calibrated considering both the geometry information and the semantic information at one specific position. The newly derived scale-aware feature $\mathbf{X}^l$ with highlighted scale variations among the foreground objects can be written as:

$$\mathbf{X}^l = f(\mathbf{Z}^l, \gamma^l) = \mathbf{Z}^l \circledast \gamma^l \tag{3.7}$$

where $\circledast$ denotes channel-wise Hardmard matrix product operation.

3.2.4 Depth Embedded Network (DeemNet)

The depth embedding module is self-contained with the same input and output dimension, and hence can be freely dropped in a standard CNN architecture to augment the representation ability, without any additional supervision or modification to the original architecture. To examine its effectiveness on backbone models with various complexity, we develop the depth embedded network (DeemNet) by integrating the proposed module into the encoder part of a shallow and a deep baseline CNN model, respectively. We first devise a lightweight model that has three convolution layers both in the encoder and decoder parts. This counting model is in the fully convolutional fashion and is able to accept arbitrary-sized inputs at inference, dubbed as *CFCN*. For the deeper counterpart we exploit the most recent *CSRNet* (Li et al. 2018) which adapts the VGG network (Simonyan & Zisserman 2015) for crowd counting with dilation processing. Detailed architectures of two backbone models are shown in Table 5.1. Besides, each convolutional layer is followed by a rectified linear unit (RELU) (omitted in the table) and is accordingly padded to keep the spatial resolution. With the two baseline CNNs, we construct two variants of DeemNet: Deem-CFCN and Deem-CSRNet. Notably

it is possible to have multiple depth embedding modules added at different stages of the baseline model. We have implemented various configurations and details will be presented in Section 5.4.

3.3 Model Training

The DeemNet can be trained with the pixel-wise Euclidean loss: $L = \|\mathbf{Y} - \mathbf{Y}_{gt}\|^2$, where $\mathbf{Y}$ and $\mathbf{Y}_{gt}$ are the predicted and the ground-truth density map, respectively. For an image $\mathbf{I}$ with its dotted annotation set A_I, the ground-truth density map is defined as a summation of a set of 2D Gaussian functions centered at each dot, i.e., $\forall p \in \mathbf{I}, \mathbf{Y}_{gt}(p) = \sum_{\mu \in A_\mathbf{I}} \mathbb{N}(p; \mu, \Sigma)$, where $\mathbb{N}(p; \mu, \Sigma)$ denotes a normalized 2D Gaussian kernel evaluated at p, with mean μ and isotropic covariance matrix Σ. Training proceeds in three phases: first the baseline model is optimized using objective L; then the attention model is firstly added and trained to provide better initialization; finally the complete depth embedding module is built and the whole model is trained end-to-end using L.

3.4 Experiments

3.4.1 Implementation

Our system is implemented with the publicly available Matconvnet toolbox (Vedaldi & Lenc 2015) with an Nvidia GTX Titan X GPU. We set the momentum to 0.9 and the weight decay to 0.0005. The initial learning rate is set to 10^{-5} and is divided by 10 when the validation loss plateaus. For each evaluation dataset, image patches are randomly cropped from the training images to augment the training data, and random flipping of patches is also applied for data augmentation. At inference summation of density values across the whole image reports the final counting numbers. Following the convention of most existing work (Zhang et al. 2015, Zhang et al. 2016), We

Table 3.1: Different encoder-decoder architectures evaluated in the experiment.

Architecture	CFCN	CSRNet
Encoder	7×7×32 conv, stride 2 7×7×64 conv, stride 2 5×5×128 conv	(3×3×64 conv)×2, stride 2 (3×3×128 conv)×2, stride 2 (3×3×256 conv)×2, stride 2 (3×3×512 conv)×2, stride 2
Decoder	5×5×64 conv 7×7×32 deconv, upsample 2 7×7×1 deconv, upsample 2	(3×3×512 conv, dilate 2)×3 3×3×256 conv, dilate 2 3×3×128 conv, dilate 2 3×3×64 conv, dilate 2 1×1×1 conv

use the mean absolute error (MAE) and the mean squared error (MSE) to evaluate and compare the counting performances. For a dataset with M test images the MAE is defined as $MAE = \frac{1}{M}\sum_{i=1}^{M}\left|C_{es}^{i} - C_{gt}^{i}\right|$, where C_{es}^{i} and C_{gt}^{i} are the predicted and the ground truth object counts for the i-th image. MSE measures the robustness of the predicted count, which is defined as $MSE = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(C_{es}^{i} - C_{gt}^{i})^2}$.

3.4.2 Datasets

We mainly exploit four benchmark datasets: ShanghaiTech (Zhang et al. 2016), WorldExpo'2010 (Zhang et al. 2015), UCF_CC_50 (Idrees et al. 2013) and the Mall (Chen et al. 2012) to evaluate the counting algorithms, which have also been widely adopted in state-of-the-art methods (Sam et al. 2017, Liu, Gao, Meng & Hauptmann 2018, Liu et al. 2019, Shi et al. n.d.) due to their unique characteristics to help validate various counting approaches. A sample image from each of the datasets are shown in Figure 3.6, and details on the four datasets are described in below.

Figure 3.6: Sample images from the four evaluation datasets: ShanghaiTech (Zhang et al. 2016), WorldExpo'2010 (Zhang et al. 2015), UCF_CC_50 (Idrees et al. 2013) and the Mall (Chen et al. 2012).

ShanghaiTech ShanghaiTech (Zhang et al. 2016) is a large-scale dataset captured in real outdoor scenes. The dataset is split into two parts with significantly varied crowd density. 482 images in part A are all crawled from the Internet, among which 300 images are used for training and the left are for testing. Images in this part are pretty crowded and with severe occlusions. For part B, it consists of 716 annotated images, which are taken by surveillance cameras from different crowd scenes. The perspective distortion in each image is pretty severe, which leads to drastic pedestrian scale variations. In our experiments, we follow the train/test splits (400 for train, 316 for test) in the original paper (Zhang et al. 2016). 20 patches are randomly cropped from each original image for model training, each with a size of 224×224.

WorldExpo'2010 The WorldExpo'10 dataset was firstly introduced in (Zhang et al. 2015). It consists of 1132 annotated video sequences captured with 108 surveillance cameras. 3980 frames are selected and labeled with dotted annotations at the center of pedestrians' heads for evaluation of the crowd counting algorithms. Among all the labeled images, 3380 frames from 103 scenes are set as training data, and the left 600 frames from another five different scenes are held out for testing. The region of interest (ROI) and a perspective map are provided for each scene. We randomly crop 20 patches with a size of 224×224 from each training image for model learning. The ROI is used to mask the predicted density map, and only the predictions within the ROI will be considered.

UCF_CC_50 UCF_CC_50 (Idrees et al. 2013) contains 50 images collected and annotated from crowd scenes which are crawled from the In-

ternet. The dataset exhibits a significant variance in the counting numbers with counts varying between 94 and 4543. The limited number of training images and the drastic variability between different scenes make this dataset very challenging for the counting task. We follow the approach of other state-of-the-art methods (Zhang et al. 2015, Zhang et al. 2016, Sam et al. 2017) and use 5-fold cross-validation to validate the performance of our method on UCF_CC_50. The cropped training patch size is 224×224 in each image.

Mall The Mall dataset (Chen et al. 2012) contains 2000 frames collected in a shopping mall. As an indoor scene, the pedestrian numbers in the images of this dataset are much smaller compared to the ShanghaiTech dataset (Zhang et al. 2016), with the maximum and the minimum number of people in the ROI regions being 13 and 53, respectively. However, this dataset also experiences apparent perspective distortion and illumination variations, which causes significant changes in the size and appearance of objects at different positions of the scene. Following the original experiment settings in (Chen et al. 2012), the first 800 frames are used for training, and the remaining 1200 frames are kept for testing. 12 patches are randomly cropped from each image for model training, each with a size of 160×160.

3.4.3 Diagnostics Experiments

In this section, we conduct extensive experiments to analyze the effects of the proposed depth embedding module on the ShanghaiTech part_B dataset (Zhang et al. 2016).

Component analysis To investigate the effects of each component in the depth embedding module, we conduct experiments with two variants of the proposed module. The first one preserves the depth encoding and embedding layers however remove the depth rectifying layer, dubbed as D-CNN. The other one only contains the attention model in the depth rectification layers and abandon the depth information, dubbed as A-CNN. To further understand the effects of the feature modulation at different positions, the depth embedding module and its variants are also applied at different stages

Table 3.2: Component analysis on ShanghaiTech-B dataset. In each stage the best MAE/MSE is indicated as **bold** and the second best as *Italic*.

Model	Stage		
	1	2	3
CFCN	13.05/21.88 (MAE/MSE)		
A-CFCN	12.67/22.13	12.91/22.44	12.77/22.41
D-CFCN	*12.25*/*21.09*	*11.95*/*21.09*	**12.09**/*20.14*
Deem-CFCN	**11.82**/**19.77**	**11.86**/**20.48**	*12.25*/**20.05**

of the base model. In particular, we denote the stage with original feature map from the n-th conv-relu-pool (or conv-relu) group as stage n. As for CFCN the 1-st, 2-nd and 3-rd stage represent the *pool1*, *pool2* and *conv3* layer, respectively. With baseline model CFCN, experimental results on the effects of each component in the depth embedding module is shown in Table 3.2.

It can be observed that with merely the depth information, the D-CFCN already improves the MAE/MSE over the baseline CFCN no matter whichever stage the feature is augmented, which validates the efficacy of explicitly exploiting the predicted depth to assistant the crowd counting task. When adding the depth rectification layer, the Deem-CFCN further improves the performances compared to D-CFCN, implying the effectiveness to selectively highlight the attentive depth to avoid possible disruption from the background. Beside, with only the attention model, the A-CFCN only slightly outperforms the baseline CFCN, which is inferior compared to the improvements of D-CFCN and Deem-CFCN. This implies that the benefits of the depth embedding module are not mainly relied on the increased parameters brought by the attention block however it lies in the designed mechanism to encode, rectify and embed the depth information to effectively handle the scale variations.

Multi-layer depth embedding To investigate the effects of applying depth embedding modules at multiple layers on the counting accuracy, we

Table 3.3: Diagnostic experiments on ShanghaiTech-B dataset on the number of depth embedding modules. Number of n denotes n proposed modules which are respectively added in the first n stage of the base CNN.

Model	Number		
	1	2	3
CFCN	13.05/21.88 (MAE/MSE)		
Deem-CFCN	11.82/19.77	11.34/18.60	11.65/18.39
CSRNet	10.6/16.0 (MAE/MSE)		
Deem-CSRNet	8.09/12.98	8.05/13.48	8.24/14.40

further conduct experiments with consecutive feature augmentation at multiple stages of the base models of CFCN and CSRNet, as shown in Table 3.3. First it can be observed that for CFCN, despite the number of the integrated depth embedding modules, the Deem-CFCN keeps improves over the baseline model, which again validates the effectiveness of the proposed method. Similar conclusion applies when using CSRNet as baseline. Besides, when using CFCN as the baseline, using two modules respectively at the first and the second stage is better than only using one. However, continuing to add the module to three the performance seems to plateau given the MSE is slightly improved while the MAE is degraded. While for CSRNet, the performance is significantly improved when using one depth embedding module while it reaches the plateau earlier with two modules integrated. Then it starts to become even worse when it comes to three models. Comparing the results both on CFCN and CSRNet, we found that for shallow networks with lower representation ability, consecutive augmentation of features to embed essential depth cues is beneficial and help fully exploit the potential of the baseline model itself. While for deeper networks which is originally stronger to generate robust representations, depth embedding at earlier stage should be enough, and we conjecture that with more modules embedded the whole model with increased capacity is prone to result in overfitting, which may degrade the improvements.

Table 3.4: Comparison results of different methods on the ShanghaiTech-B.

Method	MAE	MSE
LBP + RR (Saunders, Gammerman & Vovk 1998)	59.1	81.7
Crowd-CNN (Zhang et al. 2015)	32.0	49.8
MCNN (Zhang et al. 2016)	26.4	41.3
Switch-CNN (Sam et al. 2017)	21.6	33.4
CP-CNN (Sindagi & Patel 2017*b*)	20.1	30.1
DecideNet (Liu, Gao, Meng & Hauptmann 2018)	20.7	29.4
ACSCP (Shen et al. 2018)	17.2	27.4
IG-CNN (Sam, Sajjan, Babu & Srinivasan 2018)	13.6	21.1
ASD (Wu, Zheng, Ye, Hu, Yang & He 2019)	8.5	13.7
CSRNet (Li et al. 2018)	10.6	16.0
Deem-CSRNet	**8.1**	**13.0**

3.4.4 Comparison with State-of-the-art

The proposed method is compared with several state-of-the-art methods on four challenging benchmarks for crowd counting, as shown in Table 3.4, 3.5, 3.6 and 3.7. Since the Mall (Chen et al. 2012) dataset contains only one scenes and also contains a few images, we use the Deem-CFCN with two depth embedding modules added on stages 1 and 2 to benchmark the performance on this dataset. For other datasets, Deem-CSRNet with one depth embedding module integrated on stage 1 is applied for results comparison.

ShanghaiTech-B As observed in Table 3.4, our method outperforms the recent state-of-the-art approaches on this dataset. Especially, compared to those methods which handle scale variations mainly by employing multi-scale features (Zhang et al. 2015, Zhang et al. 2016, Sam et al. 2017, Wu et al. 2019), our models with the proposed depth embedding module achieve better performance, which demonstrates the efficacy to exploit the depth to explicitly model scale variations for crowd counting.

WorldExpo'2010 Table 3.5 compares the MAE with other methods

Table 3.5: Comparison results of MAE on WorldExpo'2010 dataset.

Method	S1	S2	S3	S4	S5	Avg
LBP + RR (Saunders et al. 1998)	13.6	59.8	37.1	21.8	23.4	31.0
Crowd-CNN (Zhang et al. 2015)	9.8	14.1	14.3	22.2	3.7	12.9
MCNN (Zhang et al. 2016)	3.4	20.6	12.9	13.0	8.1	11.6
Switch-CNN (Sam et al. 2017)	4.4	15.7	**10.0**	11.0	5.9	9.4
CP-CNN (Sindagi & Patel 2017*b*)	2.9	14.7	10.5	10.4	5.8	8.86
CSRNet†	2.3	**13.0**	14.2	10.5	3.5	8.7
Deem-CSRNet	**2.1**	14.1	12.7	**9.4**	**3.4**	**8.3**

† This is our re-implementation of the CSRNet. Close average MAE (8.7) has been achieved compared to the reported MAE (8.6) in the original paper (Li et al. 2018), however with different result on each separate scene. For comparison, we base the Deem-CSRNet on our own-implemented CSRNet.

on each test scenes as well as the average MAE across all the scenes. As observed, our approach outperforms previous methods with an average MAE of 8.34, demonstrating the effectiveness of the proposed method on cross-scene counting. We have noticed that with the depth embedding module, the counting errors of scene 2 increases. Based on our analysis, in this scene, the ROI regions are almost directly under the surveillance camera. In this situation, the perspective distortion and the scale variation is not the primary factor influencing the counting accuracy, which thus limits the effectiveness of the proposed method to mainly handle scale variations presented in general surveillance scenes.

UCF_CC_50 As observed in Table 3.6, our method improves over the baseline model and achieves the best MAE compared to other state-of-the-art methods, which implies the effectiveness of the proposed approach on extreme dense scenes.

Mall As observed in Table 3.7, with two depth embedding modules injected in the baseline model, the performance improves and is comparable compared with other methods, which demonstrates the effectiveness and

Table 3.6: Comparison results of MAE and MSE on UCF_CC_50 dataset.

Method	MAE	MSE
Lempitsky *et al.* (Lempitsky & Zisserman 2010)	493.4	487.1
Idrees *et al.* (Idrees et al. 2013)	419.5	541.6
Crowd-CNN (Zhang et al. 2015)	467.0	498.5
MCNN (Zhang et al. 2016)	377.6	509.1
MoCNN (Kumagai et al. 2018)	361.7	493.3
Hydra2s (Onoro-Rubio & López-Sastre 2016)	333.7	425.3
Switch-CNN (Sam et al. 2017)	318.1	439.2
CP-CNN (Sindagi & Patel 2017*b*)	295.8	**320.9**
Mohammad *et al.* (Hossain et al. 2019)	271.6	391.0
CSRNet (Li et al. 2018)	266.1	397.5
Deem-CSRNet	**253.4**	364.4

Table 3.7: Comparison results of MAE and MSE on Mall dataset.

Method	MAE	MSE
Ridge Regression (Saunders et al. 1998)	3.59	19.0
MORR (Chen et al. 2012)	3.15	15.7
Count Forest (Pham, Kozakaya, Yamaguchi & Okada 2015)	4.40	2.40
Weighted VLAD (Sheng et al. n.d.)	2.41	9.12
Exemplary Density (Wang & Zou 2016)	1.82	2.74
Boosting CNN (Walach & Wolf 2016)	2.01	N/A
MoCNN (Kumagai et al. 2018)	2.75	13.4
DecideNet (Liu, Gao, Meng & Hauptmann 2018)	**1.52**	**1.90**
CFCN	3.14	3.90
Deem-CFCN	2.10	2.66

robustness of the proposed approach on small datasets with fewer people. The DecideNet (Liu, Gao, Meng & Hauptmann 2018) additionally fuses the detection-based results to adjust the regression-based density estimation, which specifically benefit sparse scenes like the Mall dataset and thus achieves the best results against pure regression-based methods.

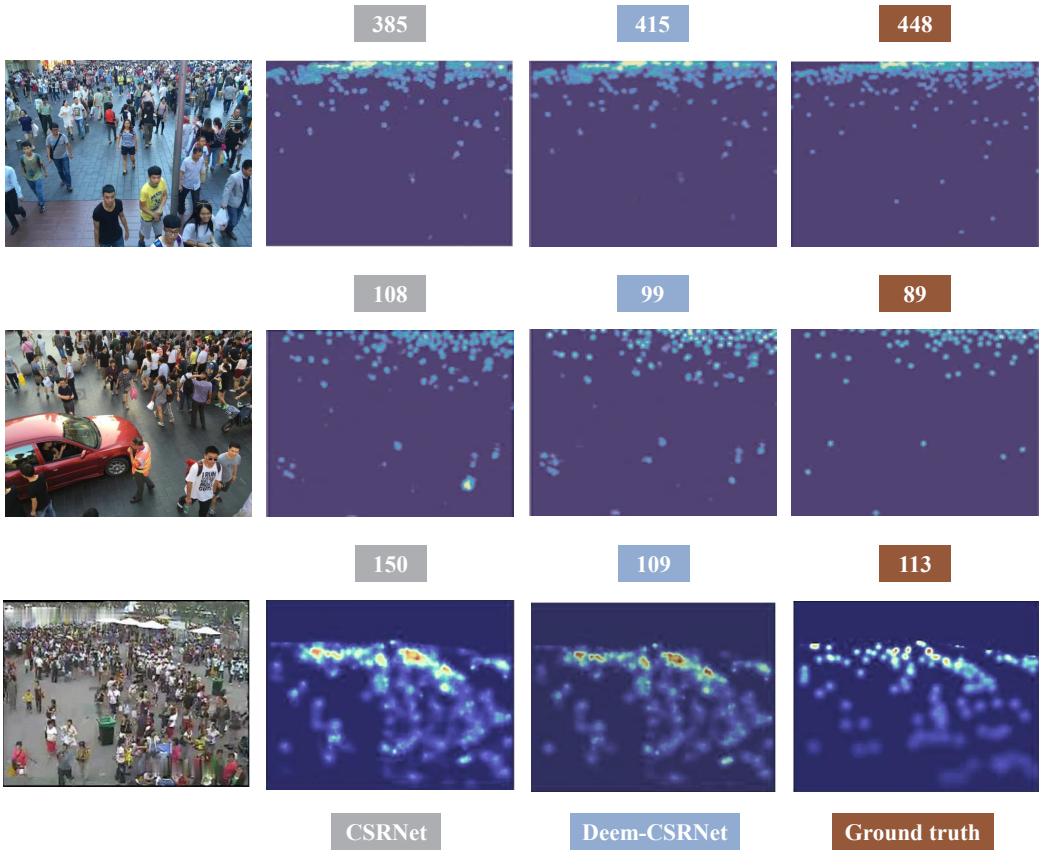

Figure 3.7: Qualitative visualization. From the first to the last column are: the images, estimated density maps without using the depth embedding module (CSRNet), estimated density maps with the depth embedding module (Deem-CSRNet) and the ground truth density maps. Crowd counts are labeled on the top, and local counts for each one-quarter-sized sub-regions of the image are also labeled for comparison.

Figure 3.7 qualitatively visualizes and compares the density maps and estimated counts with (Deem-CSRNet) and without (CSRNet) the depth

embedding module. As observed, with the proposed module the estimated density map become more close to the ground truth, and also the estimated counts become more accurate. For example, in the second sample after depth embedding the response in the nearer, lower-right regions are decreased and become more close to the ground truth, implying the ability of the Deem-CSRNet to generate scale-aware density values for pedestrians at different positions.

3.5 Conclusion

Given the fact that perspective handling is effective and necessary in traditional hand-crafted feature based methods and the circumstances that it has not explored with modern deep architectures, this chapter explores the mechanism of injecting perspective-related information into deep neural networks to improve crowd counting performances. To drive the backbone model successfully absorb the injected information and respond correspondingly, a novel depth embedding module is proposed to improve the representation capacity on scale variations of a network by dynamic spatial-wise feature recalibration with rectified depth cues. The proposed depth embedding module is fully differentiable and compatible with existing CNN-based approaches. Experimental results demonstrate the effectiveness of the depth embedded networks (Deem-CNN) which achieve state-of-the-art performance on multiple datasets.

From this chapter, a new conclusion can be drawn. The success of Deem-CNN which additionally incorporates geometric information into the network inversely indicates the limitations of plain CNN architectures in perceiving and modeling the scale variations to generate scale-aware features. For tasks requiring awareness to the scene geometrics, it will be beneficial to consider additionally geometric priors to inform the network on the geometric variations.

Chapter 4

Towards Locally Consistent Object Counting with Constrained Multi-stage Convolutional Neural Networks

In Chapter 3, we have validated the effectiveness of explicitly incorporating side information into the network. The conclusion could be drawn that it is not quite effective for a plain CNN model to capture the underlying geometric relationships between objects and thus is powerless to depict their scale variations. Companied with such a conclusion, we are curious about another question: without the injected geometric priors, how can a network improve its understanding of the scale variations and the underlying geometrics? This chapter explores this question on the design mechanism to improve the model for crowd counting. Starting with an observation of the local inconsistency problem of the density map prediction with plain CNN architectures, we propose a constrained multi-stage Convolutional Neural Networks (CMS-CNN) for jointly handling from two aspects. The multi-stage formulation help pursue locally consistent density map through repeatedly evaluation and refinement, with an additional grid loss function to further constrain the model

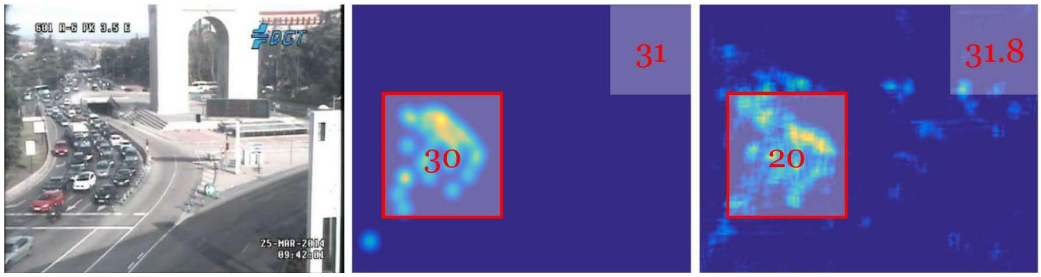

Figure 4.1: Illustration of a locally inconsistent density map prediction. (a) to (c): the original image, the ground truth and the estimated density map. We observe that although the estimated total count (shown in the upper right box) is very close to the ground truth, the quality of prediction is not satisfactory with observation of obvious background noise and count errors of local regions (shown in the red-line-framed boxes).

to satisfy the demanding of locally consistent density values. Comparisons have been conducted with several recent state-of-the-art methods to reveal the effectiveness of the multi-stage design and the constraint function.

4.1 Introduction

Based on the density-based counting paradigm, recent CNN-based counting approaches (Zhang et al. 2015, Onoro-Rubio & López-Sastre 2016, Zhang et al. 2016, Sam et al. 2017, Sindagi & Patel 2017*b*, Xie, Noble & Zisserman 2018) have significantly advanced the performance of crowd counting in several benchmark datasets. However, despite the reported low errors, does the global counts really count? Our observation is that despite the improved global counting accuracy, significant local counting errors exist when we dive into the predicted density map. This phenomenon has also been mentioned in (Guerrero-Gómez-Olmedo, Torre-Jiménez, López-Sastre, Maldonado-Bascón & Oñoro-Rubio 2015, Sindagi & Patel 2018). However, this problem has not been sufficiently investigated and addressed.

Here we term this problem as ***local inconsistency***. This is to denote

the fact that, although a predicted density map can report accurate global count for an input image, the quality of prediction is not good from local perspectives: errors arise when counting objects in subregions of the image. This can be mainly attributed to the various object scales for most images taken in surveillance scenes with perspective distortion. With this property, the model is usually difficult to generate density values which adapt to the drastic changing scales. An example of a locally inconsistent prediction of density map is shown in Fig. 4.1. It can be observed that the estimated global count (*31.8*) is very close to the ground-truth(*30*). However, errors are exposed to the selected ROI and background regions. For the ROI area with objects, the predicted local count is only *20*, which is far more satisfactory compared to its real value (30). At the same time, the predicted count (*11.8*) for the background region takes a nearly 30% proportion of the estimated global count (*31.8*), whose influence to the counting accuracy should not be neglected. The existence of local inconsistency of the predicted density map not only degrades the reliability of the finally reported object count, and also limits the quality of predicted object density distribution for related higher-level tasks (Sindagi & Patel 2018). In Section 4.2 we mathematically demonstrate that for an image the local object counting errors decide the upper bound of the global counting errors. In this way, pursing a locally consistent density map which aims to decrease local counting errors as much as possible is a reliable way to help improve the global counting accuracy.

In this chapter, we start from this observation of locally inconsistent problem and propose a joint solution from two aspects. Current existing CNN-based methods handle object scale variations mainly by engineering multi-scale features either with multi-column architectures (Zhang et al. 2016, Sam et al. 2017, Kumagai et al. 2018) or with multi-resolution inputs (Onoro-Rubio & López-Sastre 2016). We differently exploit a simple yet effective stacking formulation of plain CNNs. Benefited from the internal multi-stage learning process, the feature map is repeatedly refined, and the density map is allowed to correct its errors to approach the ground-truth density dis-

tribution. The multi-stage network is fully convolutional and can generate corresponding-sized density map for an arbitrary-sized input image. We also propose a grid loss function to further refine the density map. With finer local-region-based supervisions, the model is constrained to generate locally consistent density values to help minimize the global training errors. The grid loss is differentiable and can be easily optimized with the Stochastic Gradient Descent (SGD) algorithm.

4.2 Relationship Between Global Counting Errors and Local Counting Errors

Given a pair of ground truth and predicted density map $\{D_{gt}, D_{es}\}$ of an image I, we manually divide both of the two maps into T non-overlap grids denoted as $B = \{b_1, b_2, \cdots, b_T\}$. Mean Absolute Error (MAE) is used to measure the global counting accuracy, i.e., $E_I = \left|\sum_{i=1}^{N} D_{gt}(p_i) - \sum_i D_{es}(p_i)\right| = \left|C_I^{dif}\right|$, where N is the pixel number in image I and C_I^{dif} is the count difference between the ground truth and the predicted ones of image I. Reformulate above equation in terms of subregions will obtain:

$$\begin{aligned} E_I &= \left|\sum_{j=1}^{T}\sum_{i\in b_j}(D_{gt}(p_i) - D_{es}(p_i))\right| \\ &= \left|\sum_{j=1}^{T} C_{b_j}^{dif}\right| \leq \sum_{j=1}^{T}\left|C_{b_j}^{dif}\right| = \sum_{j=1}^{T} E_{b_j}, \end{aligned} \tag{4.1}$$

where $C_{b_j}^{dif}$ denotes the count difference of ground truth and the predicted ones within local region b_j and thus E_{b_j} is the corresponding MAE error. From Eqn. (4.1) it can be concluded that summation of MAE of object counts in each non-overlap subregions is an *upper bound* of the MAE of the global object counts in the whole image. From this perspective, pursuing a locally consistent density map which aims to decrease local counting errors will help improve the reliability as well as drive the accuracy of the global object counts.

4.3 Constrained Multi-stage Convolutional Neural Networks

Our overall model consists of two components, *multi-stage convolutional neural network* and the *grid loss*. Since the grid loss provides additional supervisions, it can be viewed as constraints to the proposed multi-stage network. Before presenting the details, we first give the formulation of density-map-prediction based object counting paradigm.

4.3.1 Density Map Based Object Counting

In this work, we formulate the object counting as a density map prediction problem (Lempitsky & Zisserman 2010). Given an image I with the dotted annotation set A_I for target objects, the ground truth density map D_{gt} is defined as the summation of a set of 2D Gaussian functions centered at each dot annotation, i.e., $\forall p \in I, D_{gt}(p) = \sum_{\mu \in A_I} \mathbb{N}(p; \mu, \Sigma)$, where $\mathbb{N}(p; \mu, \Sigma)$ denotes a normalized 2D Gaussian kernel evaluated at p, with mean μ on each object location and isotropic covariance matrix Σ. Total object count C_I for image I can be obtained by summation of pixels' values over the density map. Note that all the Gaussian are summed to preserve the total object count even when there are overlaps between objects (Onoro-Rubio & López-Sastre 2016).

Given this counting framework, the goal of our work is to learn a mapping function from an input image I to its estimated object density map D_{es}, i.e., $\forall p \in I, D_{es}(p) = F(p|\Theta)$, where the underlying model is parameterized by Θ.

4.3.2 Multi-stage Convolutional Neural Network

To generate locally consistent density values, we resort to the stacking formulation of plain CNNs. We exploit the internal multi-stage inference mechanism to repeatedly evaluate the feature map and allow the generated density

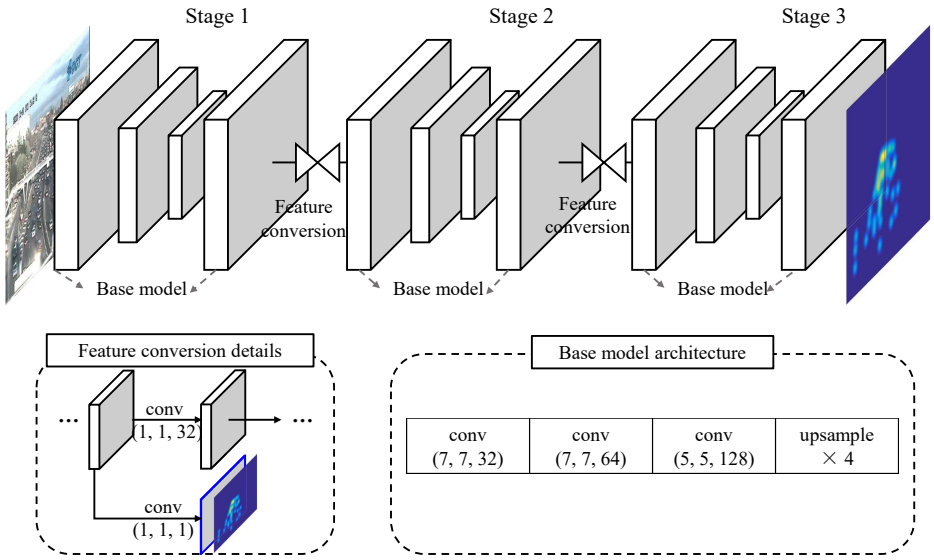

Figure 4.2: Architecture of the multi-stage convolutional neural network. We stack several base models sequentially with feature conversion blocks which i). perform feature dimension alignment of feature maps between two adjacent base models, and ii). generate a prediction for each base model to enable intermediate supervision. The first base model accepts the input image, and the rest base models in the following stages accept feature maps which comes from the previous feature conversion block.

map to be refined to figure out the best-suited density values. Mathematically, For each pixel p in the training image I, we learn the mapping function $F(p|\Theta)$ in a compounded way with a series of functions from different stages:

$$F(p|\Theta) = f_K(\cdot|W^K) \circ \cdots \circ f_s(\cdot|W^s) \circ \cdots \circ f_2(\cdot|W^2) \circ f_1(\cdot|W^1)(p), \quad (4.2)$$

where $\{f_s, s = K, K-1, \cdots 2, 1\}$ represents the base model parameterized by W^s in the s stage, and $\circ$ denotes the function compounding operation. With this decomposition, we can add intermediate supervisions (Lee, Xie, Gallagher, Zhang & Tu 2015) to each base model f_s to facilitate the training

process. A pixel-wise $L2$-norm loss function can be applied for training:

$$\begin{aligned} L(W, D_{es}) &= \frac{1}{N} \sum_{p} \sum_{s} \alpha_s \left\| D_{es}^s(p) - D_{gt}(p) \right\|_2^2, \\ &= \frac{1}{N} \sum_{p} \sum_{s} \alpha_s L_p^s \end{aligned} \tag{4.3}$$

where $D_{es}^s = f_s(X^{s-1}|\widehat{W}^s)$ is a side output density map of base model f_s, X^{s-1} are feature maps produced by the model f_{s-1} in the previous stage, $W = \{W^s, \widehat{W}^s\}_{s=1,\cdots,K}$ are parameters of the whole model, N is the number of pixels in image I and α_s is the weight for the side output loss of base model f_s.

Fig. 4.2 illustrates the proposed multi-stage model, where the base model is formulated as a fully convolutional neural network (Long et al. 2015). For a convolution (conv) layer, we use the notation (h, w, d) to denote the filter size $h \times w$ and the number of filters d. Inspired by (Zhang et al. 2015) the convolution part of our base model contains three convolution layers with sizes of (*7, 7, 32*), (*7, 7, 64*) and (*5, 5, 128*) respectively, each followed by a ReLu layer. Max Pooling layer with 2×2 kernel size is appended after the first two convolution layers. Considering the input image is downsampled by a stride of 4, we add a deconvolution layer at the end of each base model to perform in-network upsampling to recover the original resolution. The resulted feature maps of each base model are fed into the subsequent stage after dimension alignment with a 1×1 convolution layer of the feature conversion block. Inspired by the success of training CNN models with deep supervisions (Lee et al. 2015, Newell, Yang & Deng 2016), another 1×1 convolution layer is appended on the feature maps to predict a side output of density map, where the intermediate supervision will be then applied. Applying supervisions on each base model help facilitate the learning process of the whole network. The feature conversion and intermediate supervision block are illustrated in Fig. 4.2. Except for the first base model that accepts the input image, the first convolution layers of the following base models are modified to be consistent with the dimensions of previously generated feature

maps.

4.3.3 Grid Loss

To further refine the density map to generate accurate global counts as well as the local counts, we also propose a grid loss function as the supervision signal. With the consideration of training error in local regions, the model is constrained by the grid loss to correct those density values which result to severely conflicts of estimated local counts with the ground truth.

Divide an image into several non-overlapping grids, and the grid loss can be depicted with local counting errors in each sub-region. The traditional pixel-wise loss (Eqn. (4.3)) measures pixel-level density divergence while the grid loss reflects region-level counting difference. Considering the numerical gap between the numerical value between the global and local counting errors, we depict the grid counting loss with the average density loss for pixels within each specific area. This is based on the assumption that within a relatively small area, it has a great chance that pixels' density values are very similar. Then it can be regarded that every single pixel within this area has a density loss which contributes to the total count loss. By distributing the total count loss to each pixel, the grid loss help drive the correction of most violated density values and improve regression accuracy. Following previous notation in Eqn. (4.3), for a group of non-overlap grid set $B = \{b_1, b_2, \cdots, b_T\}$ in the predicted density map D_{es}, the grid loss is defined as

$$L_{grid} = \sum_{j=1}^{T} \left\| \frac{1}{|b_j|} \left(\sum_{p \in b_j} D_{es}(p) - \sum_{p \in b_j} D_{gt}(p) \right) \right\|_2^2, \tag{4.4}$$

where $|b_j|$ denotes the pixel number in this grid. Reformulation of the grid loss for the multi-stage model will be

$$\widehat{L}_p^s = (1 - \lambda^s) L_p^s + \lambda^s L_{grid}^s, \tag{4.5}$$

where λ^s is a weight scaler applied to trade off between the estimator, i.e., the traditional pixel-wise loss and the modulator, i.e., the proposed grid loss.

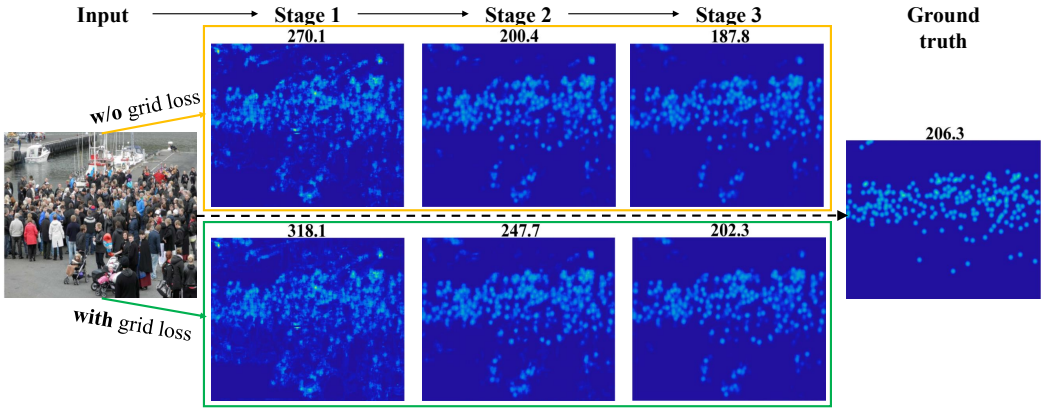

Figure 4.3: Effects of the grid loss on a three-stage model. It can be observed that training with grid loss drives the model to learn to correct the regression errors and produce more accurate object counting results.

Substitute L_p^s in Eqn. (4.3) with Eqn. (4.5) will derive the final grid loss used to supervise the whole network. With this formulation, it can be observed that each pixel is not only supervised by the original density loss, and is also additionally regularized by the average density loss of the block it belongs to. This will drive the model to correct those density values that are not consistent with local object counts and improve final counting accuracy. In Fig. 4.3 a sample image is given to show the effects of the grid loss on a three-stage model. It can be seen that training the multi-stage model with grid loss is able to drive the model to correct regression errors and obtain more accurate object counts.

4.4 Experimental Results

4.4.1 Implementation

Our model is implemented using MatConvNet (Vedaldi & Lenc 2015) with the SGD optimization. The hyper-parameters of our network include the mini-batch size (64), the momentum (0.9) and the weight decay (5×10^{-4}). Training starts from an initial learning rate of 1×10^{-6}, which is divided by

10 after the validation loss plateaus. Considering the difficulty to train a deep model from scratch, we take advantage of the widely-used pre-training strategy. The base CNN model is first trained and then is duplicated to construct the multi-stage network. Additional weights, e.g., the feature alignment layers between adjacent base models are randomly initialized. Finally, the whole model is fine-tuned end-to-end. During training, 20 image patches with a size of 224×224 are randomly cropped from each training image for data augmentation. Randomly flipping and color jitter are performed for data augmentation. Note that the ground truth density map is a combination of 2D Gaussian functions, and their numeric values are very small (10^{-3}∼10^{-5}) to enable effective learning. For this reason, we magnify the ground truth density map by a factor of 100 during the training process. With end-to-end training, it takes about 15 hours to train a 3-stage CNNs on a single NVIDIA TITAN X GPU. For testing it takes about $0.15s$ for an image of size 576×720.

Given a test image I, we directly use the output from the last stage of the network as the density map prediction. Three standard metrics are utilized for evaluation: mean absolute error (MAE), mean squared error (MSE), and the grid average mean absolute error (GAME). MSE and MAE evaluate the global object counts while ignoring the local consistency of predicted density maps. We additionally include the Grid Average Mean Absolute Error (GAME) (Guerrero-Gómez-Olmedo et al. 2015) as a complementary evaluation metric. After dividing a density map into 4^L non-overlapping regions, GAME for level L is defined as:

$$GAME(L) = \frac{1}{M} \cdot \sum_{i=1}^{M} \left(\sum_{l=1}^{4^L} \left| C_{es}^{il} - C_{gt}^{il} \right| \right), \tag{4.6}$$

where C_{es}^{il} and C_{gt}^{il} denotes the predicted and ground truth counts within the region l respectively. The higher L, the more restrictive this GAME metric will be on the local consistency of the density map. Note that the MAE metric is a special case of GAME when $L = 0$.

There are three hyper-parameters in the proposed grid loss function: the grid size, the loss weights α for each base model and the weights λ to balance the pixel-wise loss and grid loss. We experimentally fix α to be 1 across different stages and study the effects of another two parameters. We use block dimension to denote the partitioned block size in the image. The weighting scaler λ is in charge of the modulation degree of a block count loss on its inner pixels. We conduct experiments comparing the MAE of applying grid loss to a 2-staged model with different hyper-parameter settings of the block dimension of 1×1, 2×2, 4×4, 8×8, 16×16 for an given image and a variety of λ in 0.9, 0.5, 0.1, 0.01. Experimental results show our method performs best with $\lambda = 0.5$ and partitioned block size of 4. We use this setting across all our experiments unless otherwise specified. $\lambda = 0.9$ degrades the original performance for almost all the grid size settings, which implies that large weighting scaler may disturb the normal density learning process. As λ further decreases, the network converges to the performance training with the pixel-wise loss. When the grid size is too big, each grid area will become too small to effectively include objects, and the performance starts to degrade to the per-pixel density loss.

Table 4.1: Performance of ablation experiments for network structures and supervisions.

index	Design choices	MAE	MSE
a.	MS-CNN-1 (the base model)	107.7	173.2
b.	CMS-CNN-1	101.0	160.4
c.	MS-CNN-2	82.3	140.4
d.	CMS-CNN-2	74.2	127.6
e.	MS-CNN-3	74.4	129.7
f.	CMS-CNN-3	73.0	128.5

4.4.2 Ablation Experiments

We perform extensive ablation experiments on ShanghaiTech Part-A dataset to study the role of the multi-stage convolution network and the grid loss separately play in the whole constrained multi-stage networks. Results of alternative design choices are summarized in Table 4.1. For simplicity, we denote the multi-stage model with n stages as MS-CNN-n, and the corresponding constrained model trained with grid loss as CMS-CNN-n.

From Table 4.1 several observations could be drawn. First, the multi-stage formulation of plain CNNs (compare between a, c, e) and the proposed grid loss (compare a and b, c and d, e and f) both demonstrate effectiveness in improving counting accuracy. Second, the overall MAE performance of the constrained multi-stage CNNs (CMS-CNN) can be improved by adding stage by stage. We observe the MSE performance of MS-CNN-3 degrades the performance of CMS-CNN-2 a little bit. We suspect this may be the reason that with more stages added, the model becomes deeper to be well optimized.

4.4.3 Comparison with the State-Of-The-Arts

ShanghaiTech The ShanghaiTech dataset (Zhang et al. 2016) is a large-scale dataset which contains 1198 annotated images. It is divided into two parts: there are 482 images in part-A and 716 images in part-B. Images in part-A are collected from the Internet and the part-B are surveillance scenes from urban streets. We follow the official train/test split (Zhang et al. 2016) which is 300/182 for part-A and 400/316 for part-B. For validation, about 1/6 images are randomly selected from the original training data to supervise the training process.

Table 4.2 reports the comparison results with five baseline methods: Crowd-CNN (Zhang et al. 2015), MCNN (Zhang et al. 2016), Cascaded-MTL (Sindagi & Patel 2017a), Switch-CNN (Sam et al. 2017), CP-CNN (Sindagi & Patel 2017b). On Part-A our methods achieves best MAE among all the

comparison methods, and the second-best MSE. We observed that most images in Part-A are extremely crowded and also have pretty uniform object scales within the image, where the context information matters much compared to considering object scale variations to derive accurate counting results. In (Sindagi & Patel 2017*b*) the counting method is proposed from the perspective of context information modeling, which better suits the situation on Part-A. On Part-B our method outperforms all other methods and evidences a 40% improvements in MAE over CP-CNN (Sindagi & Patel 2017*b*). Fig. 4.4 illustrates the inference process in each stage with the CMS-CNN-3 model of two sample images from ShanghaiTech dataset. For the first image, it can be observed that the total object counts gradually approaches the ground truth. What's more, errors exist in the upper left background region are gradually refined and the local counting accuracy is also gradually improved. The similar situation can be observed for the second image, where the predicted density map is becoming more consistent with the ground-truth density distributions.

Table 4.2: Comparison results on the ShanghaiTech dataset.

Method	Part-A		Part-B	
	MAE	MSE	MAE	MSE
Crowd-CNN (Zhang et al. 2015)	181.8	277.7	32.0	49.8
MCNN (Zhang et al. 2016)	110.2	173.2	26.4	41.3
Cascaded-MTL (Sindagi & Patel 2017*a*)	101.3	152.4	20.0	31.1
Switch-CNN (Sam et al. 2017)	90.4	135.0	21.6	33.4
CP-CNN (Sindagi & Patel 2017*b*)	73.6	**106.4**	20.1	30.1
CMS-CNN-2 (ours)	74.2	127.6	15.0	25.8
CMS-CNN-3 (ours)	**73.0**	128.5	**12.0**	**22.5**

TRANCOS We also report our results on another dataset for car counting to validate the effectiveness of the proposed method. TRANCOS (Guerrero-Gómez-Olmedo et al. 2015) is a publicly available dataset which contains 1244 images of different traffic scenes obtained by surveillance cameras. An

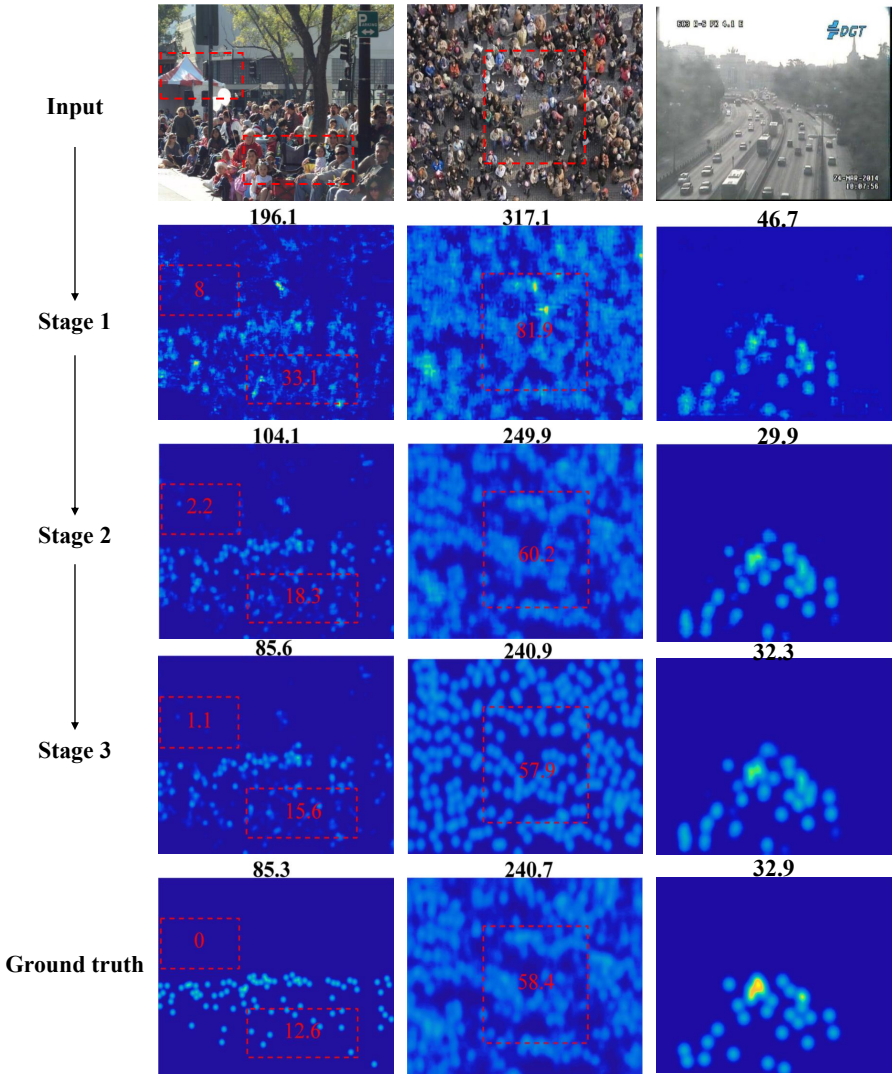

Figure 4.4: Density map prediction results as input images proceed through the multi-stage convolution model. The first row lists images sampled from the ShanghaiTech dataset (first two) and the TranCos dataset (last one). The second to the fourth rows show the intermediate outputs from the first two stages and the final prediction of the last stage, respectively. The ground truth density maps are shown in the last row. Object count derived from the density map are labeled on top of each prediction result. For the first two crowded sample images we also randomly select several subregions to track the local object counts, which are shown in the red boxes.

ROI map is also provided for each image. We strictly follow the experimental setup proposed in (Guerrero-Gómez-Olmedo et al. 2015) for training and testing, where there are 403, 420 and 421 images are settled for train, validation and test, respectively.

Table 4.3 reports the comparison performance on this dataset with four state-of-the-art approaches: density MESA (Lempitsky & Zisserman 2010), regression forest (Fiaschi et al. 2012), Hydra CNN (Onoro-Rubio & López-Sastre 2016) and MCNN (Zhang et al. 2016). The GAME metric with $L = \{0, 1, 2, 3\}$ is utilized for evaluation. Across all the levels of GAME, our method achieves the best results compared to other approaches. There is another work (Zhang, Wu, Costeira & Moura 2017*a*) reporting their GAME∼0 result of 5.31 on this dataset. However, the other three metrics (GAME∼1, 2, 3) are unavailable for direct and effective comparison. A qualitative result for a sample image from the TRANCOS dataset is shown in Fig. 4.4 (the third column). It can be seen that the model is able to generate accurate global counting errors with obvious improvements stage-by-stage to become consistency with ground-truth density map.

Table 4.3: Comparison results of GAME on the TRANCOS dataset.

Method	GAME 0	GAME 1	GAME 2	GAME 3
regression forest (Fiaschi et al. 2012)	17.8	20.1	23.6	26
density MESA (Lempitsky & Zisserman 2010)	13.8	16.7	20.7	24.4
Hydra CNN (Onoro-Rubio & López-Sastre 2016)	11	13.7	16.7	19.3
MCNN (Zhang et al. 2016)	9.9	13	15.1	17.6
CMS-CNN-2 (ours)	7.8	9.8	11.6	13.7
CMS-CNN-3 (ours)	**7.2**	**9.7**	**11.4**	**13.5**

4.5 Conclusions

This chapter explores a specific multi-stage architecture and its effect to address the local inconsistency problem of crowd counting. The internal multi-stage inference provides opportunities for features to be repeatedly

evaluated and refined, with which the final feature will be better optimized towards more accurate density estimation. The effect of the proposed grid loss function is also studied. With local-region-level supervisions, the model also demonstrates the ability to correct density values which violate the local counts. Extensive experiments and comparisons with recent state-of-the-art approaches demonstrate the effectiveness of the proposed method.

This chapter indicates that without additional geometric prior, implicitly enhancing the model capacity with some specific design scheme, e.g., the multi-stage architecture explored here, can also provide benefits. It's also notable that the grid loss function can be seen as an implicit way to embed the geometric information into the model since for the same-area grid generally contains different number of pedestrians due to the scale variations. This also validates the importance to inform the network to be aware of the perspective-related variations, either explicitly with geometric priors or implicitly with additional constraints.

Chapter 5

Leveraging Heterogeneous Auxiliary Tasks to Assist Crowd Counting

In Chapters 3 and 4, it is proved beneficial for crowd counting to either explicitly inject geometric information or implicitly exploit informative architectures and constraints. These two methods improve their ability to handle scale variations of objects, however at the expense of appending moderate modifications to the original base model. This observation has driven us to seek the solution for another problem: whether a plain CNN model without specific designs can be entitled with the ability to perceive the scale variations? In other words, how can we fully excite the capacity of a CNN towards the difficult problems of crowd counting? This chapter exhibits our exploration of this question by resorting to those compound factors in the density prediction. A few auxiliary attributes are leveraged to regularize the representation learning of the network in order to generate features with desired attributes. Each attribute is formulated as an auxiliary task, which together provide joint regularization effects to the backbone CNN for more robust representations and density estimation. Comparisons are conducted with state-of-the-art results on several challenging datasets to validate the

efficacy of the proposed method.

5.1 Introduction

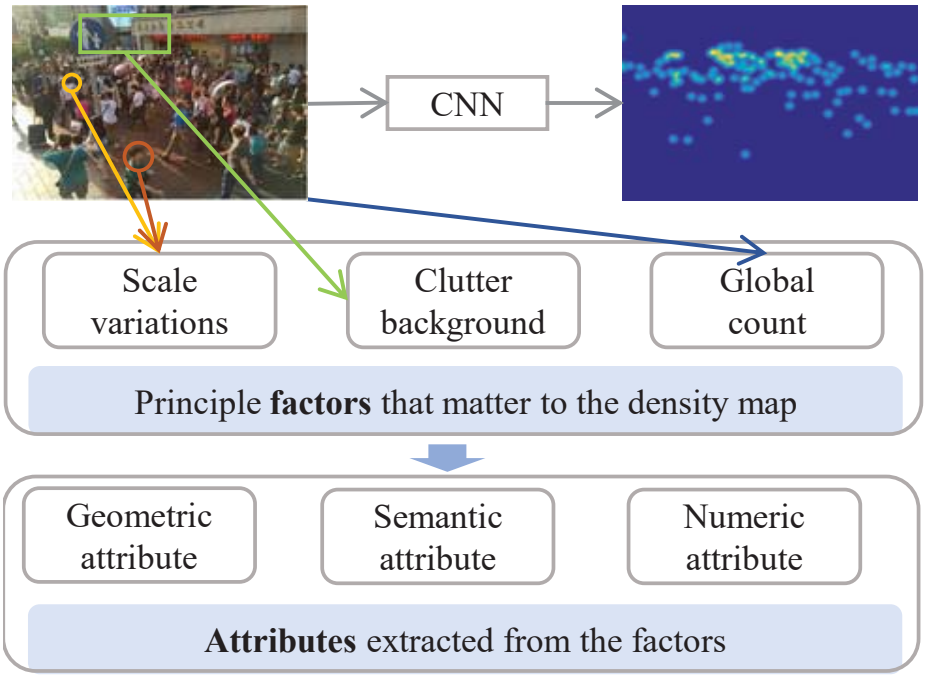

Figure 5.1: Motivation.

The compound presence of drastic scale variations, the cluttered background, and the severe occlusions makes it a challenging task to generate a high-quality density map. Various CNN-based methods (Zhang et al. 2015, Zhang et al. 2016, Sindagi & Patel 2017*b*, Li et al. 2018) have been proposed to handle the challenging situations mainly by fusing multi-scale or multi-context features to improve the feature representations for crowd counting. For example, Zhang *et al.* (Zhang et al. 2016) generate multi-scale features with the multi-column network towards more robust feature representations against drastic scale variations. Sindagi *et al.* (Sindagi & Patel 2017*b*) incorporate local and global contextual information of crowd images and fuse multi-context feature for density estimation. Their successes demonstrate the

effectiveness to incorporate information from various sources (different sub-models). Motivated by these methods, we propose to leverage heterogeneous attributes of the density map, however use these information as guidance to fully exploit the potential of the underlying representation itself, without explicit modifications to the features.

Figure 5.1 illustrates our motivation with the observation of three factors of the density estimation. Considering the formulation of density-estimation-based counting paradigm (Lempitsky & Zisserman 2010) which sums the density values over any region to report the final count, it is desired the estimated densities vary along with object scales (different occupied regions) given the factor of intra-image scale variations of crowd images. Specifically, the nearer, larger objects should have smaller density values compared with farther objects with smaller scales. We term this as the geometric attribute of the density map. Besides, the cluttered background is another factor that should not be neglected. For more accurate density estimation, the density distribution is also desired to conform with crowd spatial distributions to avoid the background clutters, which can be viewed as the semantic attribute of the density estimation. Additionally, the global count is also an important indicator measuring the overall density level of one certain image, which can be termed as the numeric attribute of the density estimation. These attributes are heterogeneous and cater for different aspects of crowd images, which should be beneficial to the quality of the density map predictions.

Inspired by these observations, in this chapter we propose to leverage the heterogeneous attributes compounded in the density map prediction. Specifically, we formulate each attribute as an auxiliary task. For the geometric attribute, we propose the monocular depth prediction to emphasize the relative depth variations of the crowd image, considering that generally the scale varying of one certain object across the scene is inversely proportional to the depth. For the semantic attribute, we introduce the crowd segmentation to highlight the foreground over the background clutters. For the numeric attribute, we introduce the direct count estimation to take care of the overall

count accuracy while optimizing per-pixel density. Learning of the auxiliary tasks will drive the intermediate features of the backbone CNN to embed the desired geometric information, semantic information and the overall density level information, which generate more robust feature against the scale variations and cluttered background. Although more objectives are involved, they are readily available either with external models or can be inferred directly from the original density map, which do not need any additional annotations. Furthermore, our formulation of the essential attributes as auxiliary tasks introduce flexibility to our approach, which can benefit any backbone CNN model for crowd counting without increasing additional computations at inference.

5.2 Methodology

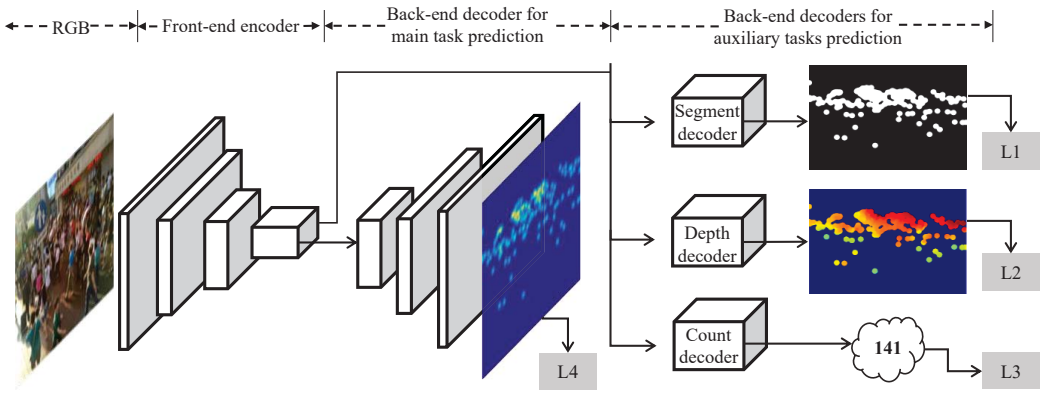

Figure 5.2: Overview of the proposed approach with the learning of three auxiliary tasks in CNNs (AT-CNN). The symbols of L1 to L3 denote the losses to optimize the auxiliary tasks of crowd segmentation, depth prediction and count regression. The symbols of L4 is the loss for the main task of density estimation.

As discussed in Section 5.1, we propose to leverage heterogeneous attributes to assist crowd counting, which mainly aims to improve the feature representations of the backbone CNN with the learning with auxiliary tasks. Generally, the crowd density estimation can be viewed as an encoding-

decoding process with a front-end CNN (encoder) mapping the input image to a high-dimensional feature maps and a back-end CNN (decoder) interpreting the features from the encoder into pixel-wise density values. Denoting the front-end CNN as a function g^e parameterized with $\mathbf{w}^e$, then the features F from the encoder can be represented as $F = g^e(\mathbf{X}; \mathbf{w}^e)$ for an input image $\mathbf{X}$. For any given backbone CNN model, our method constructs the auxiliary tasks prediction (AT) module which uses the deep features $F$from the front-end CNN to optimize the auxiliary predictions and inversely improve the intermediate representations itself. The framework of our method is shown in Figure. 5.2. During training, ground-truth labels for the density estimation and the three auxiliary tasks, i.e., depth prediction, crowd segmentation and count estimation are used. Although four different kinds of supervision signals are used, we do not require any extra annotation effort. Specifically, we exploit modern CNN-based depth prediction models to derive the ground-truth labels for the auxiliary depth prediction, and the crowd segment and count can be directly inferred from the density map labels, respectively.

5.2.1 Auxiliary Tasks Prediction

Based on the deep features from the front-end CNN, we build the three auxiliary tasks, i.e., depth prediction, crowd segmentation and the count estimation. These three tasks, with each in charge of one characteristics of the density map, can provide multi-fold regularization effects to optimize the front-end CNN. We describe the details for each auxiliary task in the following article.

Attentive Crowd Segmentation Due to the complex situations such as the extremely limited pixels of pedestrians occupied in the image as well as the cluttered background, the crowd density map is usually noisy. Towards this problem, we introduce the crowd segmentation as an auxiliary task, which will help the front-end CNN generate more discriminative representations and thus purify the output prediction.

A segmentation decoder network g^{seg} parameterized with $\mathbf{w}^{seg}$ is built as

the back-end CNN for crowd segmentation. Performing a two-way classification task, the inputs to the decoder is the feature F from the front-end encoder and the outputs is a crowd segment $\widehat{\mathbf{S}}$ with values indicating the probability of pixels belonging to the targets: $\widehat{\mathbf{S}} = g^{seg}(F; \mathbf{w}^{seg})$. Ground-truth labels for crowd segmentation can be inferred from the dotted annotations of pedestrians provided in counting dataset (Zhang et al. 2016, Zhang et al. 2015) by simple binarization as shown in Figure 5.3. We dubbed the result as *attentive* crowd segment, since it conveys important information clarifying the attentive areas occupied by the targeted objects. Strictly speaking the derived segment map is not the same as the ones usually seen in semantic segmentation (Kang & Wang 2014) where detailed boundaries of objects are depicted, however we show in experiment this simple strategy can yield effective improvements for density estimation.

Given a pair of input image and the ground-truth attentive crowd segmentation map $\{\mathbf{X}, \mathbf{S}\}$, loss function for the segmentation decoder is the binary cross-entropy between the predicted and the ground-truth probability of each pixel:

$$L_1 = \frac{1}{|\mathbf{X}|} \sum_{(i,j)\in\mathbf{X}} t_{ij} log o_{ij} + (1 - t_{ij}) log(1 - o_{ij}), \tag{5.1}$$

where $t_{ij} \in \{0, 1\}$ is the actual classes of pixels in $\mathbf{S}$ with 1 for the target area and 0 for the background, and o_{ij} denotes the pixel-wise probability in the prediction $\widehat{\mathbf{S}}$.

Distilled Depth Prediction To handle the perspective distortion in surveillance scenes (Sindagi & Patel 2018), we introduce the single-image monocular depth prediction as an auxiliary task. Informally speaking, for a given object category (e.g., pedestrians) the size of an object in the image is inversely proportional to the distance from the camera (Kong & Fowlkes 2018). In the regions with larger depth values, the objects have smaller sizes and should be adversely assigned with larger density values to guarantee their summation gives accurate counts. By inferring the depth maps, the front-end CNN is imposed to take care of the scene geometry and hence gains the

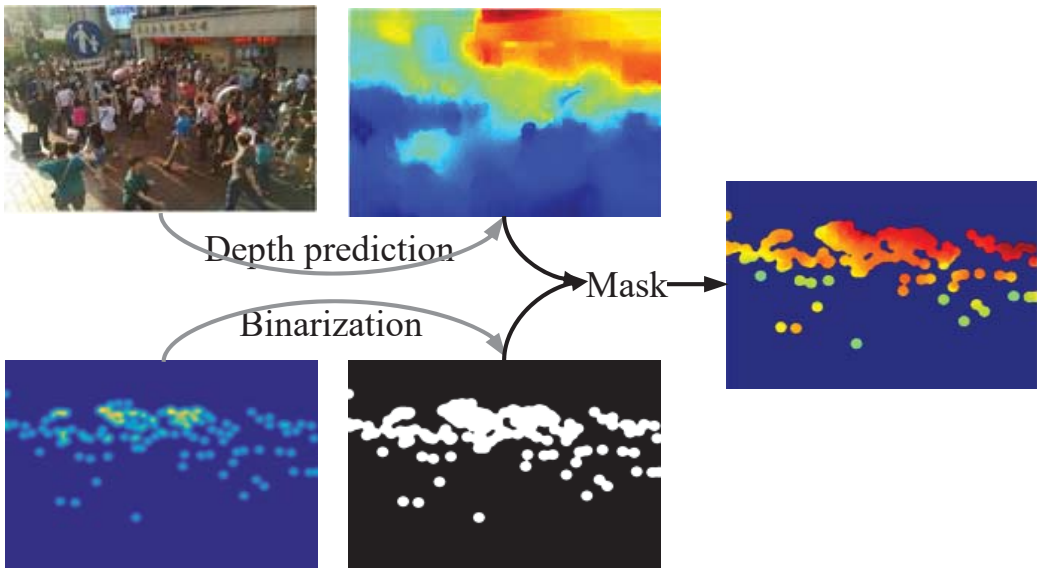

Figure 5.3: Label generation for auxiliary tasks. Given a pair of crowd image and its ground truth density map (the first column), the depth map can be estimated using external depth prediction algorithms (Liu et al. 2016) and the crowd segment is inferred through binarization of the density map (the second column). The distilled depth map (the third column) used to supervise the auxiliary task is obtained by masking the original estimated depth map with the crowd segment map.

awareness to the intra-image scale variations, which will help generate more discriminative features for scale-aware density estimation.

Similar to the task of crowd segmentation, a depth decoder network g^{dep} parameterized with $\mathbf{w}^{dep}$ is built for depth prediction. The input to the decoder is the features F from the front-end CNN and the output is the depth map with values indicating the distances of each pixels to the camera: $\widehat{\mathbf{D}} = g^{dep}(F; \mathbf{w}^{dep})$.

Towards this task, we resort to depth maps derived from the CNN-based DCNF model (Liu et al. 2016) for monocular depth prediction. The DCNF can estimate depths for general scenes with no geometric priors nor any extra information injected, and hence is suitable in our situation to help illustration of geometry in crowded scenes. Given the input crowd image $\mathbf{X}$, we use the

pre-trained DCFN model (Liu et al. 2016) to generate a *raw* measurement of depth $\mathbf{D}_{raw}$. As observed in Figure 5.3, it is capable of depicting depth disparities between pedestrians at different positions. However, due to the DCNF model has not been specifically adapted to the target scenes in the crowd counting tasks and the predictions contain clutters that degrades the efficiency, especially for background areas. Towards this problem, we further calculate a *distilled* depth map $\mathbf{D}$ which only preserve the depth information of the attentive target areas. This is derived using both the raw depth map and the attentive crowd segment: $\mathbf{D} = \mathbf{S} \odot \mathbf{D}_0$, where $\odot$ denotes the Hadamard matrix multiplication. With the distilled depth as the supervision for depth prediction, the front-end CNN is desired to be especially aware to the depth relationships/scale variation between those attentive areas with target objects.

With the training pairs of $\{\mathbf{X}, \mathbf{D}\}$, the depth decoder can be trained using a simple Euclidean loss for the predicted depth map $\widehat{\mathbf{D}}$:

$$L_2 = \frac{1}{|\mathbf{D}|} \sum_{(i,j)\in\mathbf{D}} \left\| \widehat{\mathbf{D}}_{ij} - \mathbf{D}_{ij} \right\|_2^2 \tag{5.2}$$

Crowd Count Regression Most density estimation based counting algorithms optimize their counting model by measuring the per-pixel errors between the predicted and the ground-truth density maps (Zhang et al. 2015, Zhang et al. 2016, Sam et al. 2017, Onoro-Rubio & López-Sastre 2016, Sindagi & Patel 2017*b*). However, one problem is this supervision is not directly related to the evaluation metric of MAE/MSE (Loy et al. 2013) which measures global counting errors of input images. To this end, we introduce another auxiliary task of crowd count regression which directly estimates the crowd count from the encoded features. Empowered with this auxiliary task, the front-end encoder will generate features adapted to the overall density level of the input image, which helps produce more accurate density values.

A count decoder g^{num} parameterized with $\mathbf{w}^{num}$ is built to map the features F from the front-end encoder to the crowd count $\widehat{C}$: $\widehat{C} = g^{cnt}(F; \mathbf{w}^{cnt})$. The ground-truth count C can be directly derived by count the dotted an-

notations in an input image $\mathbf{X}$. The L_2 norm is used to train the count decoder:

$$L_3 = \left\| \widehat{C} - C \right\|_2^2 \tag{5.3}$$

5.2.2 Main Tasks Prediction

The density estimation decoder g is built on the features F emitted from the front-end encoder to perform the main task of density estimation. To generate the ground-truth density maps, we follow (Lempitsky & Zisserman 2010) to apply 2D Gaussian kernels on each dotted annotations, where the same-spread (sigma Σ) Gaussian kernels are simply adopted at different positions. The decoder for the main task is trained using the Euclidean loss for the density map $\widehat{\mathbf{Y}}$:

$$L_4 = \frac{1}{|\mathbf{Y}|} \sum_{(i,j)\in\mathbf{Y}} \left\| \widehat{\mathbf{Y}}_{ij} - \mathbf{Y}_{ij} \right\|_2^2 \tag{5.4}$$

5.2.3 Optimization

The final learning objective function utilizes multiple losses weighted by hyper-parameters:

$$L_{mt} = \sum_{i=0}^{4} \lambda_i L_i \tag{5.5}$$

The four losses $L_i, i \in \{1, 2, 3, 4\}$ corresponds to the task of attentive crowd segmentation, distilled depth prediction, count regression and the density estimation, respectively. We employ a stage-wise procedure to train the network with auxiliary tasks, by varying the hyper-parameters as detailed in Section 5.3.

5.3 Implementation

We implemented the network using the publicly available Matconvnet toolbox (Vedaldi & Lenc 2015) with a Nvidia GTX Titan X GPU. Stochastic gradient descent (SGD) is used to optimize the parameters. We set the

momentum and weight decay to 0.9 and 0.0005, respectively. We used the initial learning rate of 10^{-6} and divided it by 10 when the validation loss plateaus. Parameters of all the deconvolution layers are fixed as the bilinear up-sampling kernels for training and inference. During training, random flipping is applied to augment the input image patches.

Training of the proposed model proceeds in four stages. First, we train the feed-forward baseline model for density estimation. Staring from the baseline, we successively train the segment decoder, the depth decoder and the count decoder. In the third stage, the four decoders are jointly optimized and the model is trained end-to-end using the objective function of Eq. 5.5.

Once the model has been trained, the auxiliary tasks prediction module can be detached,and the original model with more powerful capacity is used at inference.

5.4 Experiments

In this section, we evaluate the proposed crowd counting method on three benchmark datasets of the shanghaiTech-B (Zhang et al. 2016), the world-Expo'2010 (Zhang et al. 2015) and the Mall (Chen et al. 2012) dataset. Following the convention of existing work (Zhang et al. 2015, Zhang et al. 2016), metrics of the mean absolute error (MAE) and the mean squared error (MSE) are computed for evaluation.

5.4.1 Datasets

Extensive experiments have been conducted on three datasets: the Mall (Chen et al. 2012), ShanghaiTech part_B (Zhang et al. 2016) and WorldExpo'2010 (Zhang et al. 2015). For the detailed information of each of the three datasets, please refer to the descriptions in Chapter 3. For the Mall dataset, We use the public splits (800/1200) for training and testing. To augment training data, we crop image patches with a size of 160×160 from the original image. For ShanghaiTech part_B, following the public splits, 400 images are for train-

ing and the left 316 are for testing. We crop image patches with a size of 224×224 for data augmentation. For WorldExpo'2010, official split is also used with 400 images for training and the left 316 for testing. Image patches with a size of 224×224 are cropped to augment training data. For all the datasets, 1/6 of the training images are randomly selected as validation to monitor the training process.

Table 5.1: Different encoder-decoder architectures evaluated in the experiment.

Architecture	AT-CFCN	AT-CSRNet
Encoder	7×7×32 conv, stride 2 7×7×64 conv, stride 2 5×5×128 conv	(3×3×64 conv)×2, stride 2 (3×3×128 conv)×2, stride 2 (3×3×256 conv)×2, stride 2 (3×3×512 conv)×2, stride 2
Decoder (for density, depth and segment prediction)	5×5×64 conv 7×7×32 deconv, upsample 2 7×7×1 deconv, upsample 2	(3×3×512 conv, dilate 2)×3 3×3×256 conv, dilate 2 3×3×128 conv, dilate 2 3×3×64 conv, dilate 2 3×3×1 conv
Decoder (for count regression)	N× N× 64 conv 1× 1×32 conv 1× 1× 1 conv	N× N× 512 conv, dropout 0.5 1× 1× 256 conv 1× 1× 128 conv 1× 1× 64 conv 1× 1× 1 conv

5.4.2 Diagnostics Experiments

To deeply analyze the proposed approach and demonstrate its effectiveness, we conduct diagnostics experiments on two evaluation datasets: the ShanghaiTech-B (Zhang et al. 2016) and the Mall (Chen et al. 2012). For the backbone CNN, we experiment with two models with various capacity to adapt to various dataset sizes and also to study the performance gains grounded on different models. A lightweight counting FCN model (CFCN)

with 3 convolution layers for both the encoder and decoder is chosen for the Mall dataset (Chen et al. 2012), and another much deeper counting model of CSRNet (Li et al. 2018) which adapts VGG network (Simonyan & Zisserman 2015) for crowd counting with dilation processing. Detailed architectures of the AT-CFCN and AT-CSRNet which integrate the auxiliary tasks prediction module are shown in Table 5.1. The convolution kernel N in the decoder for count regression depends on the input image size and the downsample factors in the front-end encoder, which transforms the feature maps into 1×1 vectors for count estimation.

From the base backbone model of CFCN/CSRNet, we compare several different variants, including those with only one auxiliary task (i) base CNN + DE: performing the depth prediction (DE) task with the front-end CNN; (ii) base CNN + SE: performing the crowd segmentation (SE) task with the front-end CNN; (iii) base CNN + CT: performing the count estimation (CT) task with the front-end CNN. The variants with two auxiliary task include (iv) base CNN + DE + SE: performing the depth prediction and crowd segmentation task at the same time; (v) base CNN + DE + CT and (vi) base CNN + SE + CT which are similar to (iv) with learning of two auxiliary tasks. Finally, we compare with the variant where all the three auxiliary tasks are integrated: (vii) of base CNN + DE + SE + CT.

Several conclusions could be drawn from Table 5.2. i). The three auxiliary tasks all take effects on decreasing the counting errors in terms of the MAE and MSE (compare $b \sim d$ vs a). This demonstrates that the auxiliary tasks carry the key information that influences the accuracy of the density estimation and jointly optimizing the main task with one of them benefit the density estimation. ii). Including any two of the three auxiliary tasks will further decrease the counting errors (compare e vs b, e vs c, f vs b), and leveraging all of them achieves the best performance. This result is in alignment with our hypothesis that the auxiliary tasks each focus on heterogeneous attributes of the density map and their collaboration will further improve the representations for more accurate density estimation. iii). The proposed ap-

Table 5.2: Diagnostic experiments of AT-CFCN and AT-CSRNet on the ShanghaiTech-B dataset (Zhang et al. 2016).

Item	Method	AT-CFCN		AT-CSRNet	
		MAE	MSE	MAE	MSE
a	base CNN	12.89	22.30	9.91	15.03
b	base CNN + DE	11.72	19.76	8.73	13.63
c	base CNN + SE	12.31	20.66	9.20	14.14
d	base CNN + CT	12.24	21.49	9.11	14.39
e	base CNN + DE + SE	11.52	19.78	8.28	13.97
f	base CNN + DE + CT	11.58	19.73	8.32	13.57
g	base CNN + SE + CT	11.88	20.42	8.51	13.66
h	base CNN + DE + SE + CT	**11.05**	**19.66**	**8.11**	**13.53**

proach not only improves the simpler model (CFCN), and also significantly improves the deep model (CSRNet) which are naturally armed with stronger representation ability. This further validates the necessity and effectiveness of the proposed approach to explicitly leverage the heterogeneous attributes existing in the density map. Similar situations can be observed from Table 5.5 for the Mall dataset (Chen et al. 2012).

5.4.3 Comparison with State-of-the-art

The proposed method is compared with several state-of-the-art methods on three challenging benchmarks. The comparison results are shown in Table 5.4, 5.6 and 5.5. As demonstrated in Table 5.4 and 5.6, our method outperforms previous methods on both the ShanghaiTech-B dataset (Zhang et al. 2016) and the WorldExpo'2010 dataset (Zhang et al. 2015). The images in both of these two datasets are collected from outdoor scenes with significant perspective variations and complex background clutters, which easily incurs the geometric and the semantic inconsistency problems. The superior performance of the proposed method demonstrates the effectiveness to leverage the auxiliary attributes during the training process to help pursue the

Table 5.3: Diagnostic experiments of AT-CFCN on the Mall dataset (Chen et al. 2012). Dep, Seg and Cot represents the corresponding auxiliary task of depth prediction, crowd segmentation and count regression, respectively.

Item	Method	MAE	MSE
a	base CNN	3.14	3.90
b	base CNN + DE	2.79	3.51
c	base CNN + SE	2.68	3.37
d	base CNN + CT	2.83	3.55
e	base CNN + DE + SE	2.36	3.02
f	base CNN + DE + CT	2.48	3.18
g	base CNN + SE + CT	2.34	2.99
h	base CNN + DE + SE + CT	**2.28**	**2.90**

geometric and semantic consistency of the density estimation. Our method is also validated on the Mall dataset (Chen et al. 2012) for sparse crowd in indoor scenes. Due to the perspective distortion is not very obvious in the indoor scenes, the effectiveness of our approach against the scale variations is limited in this dataset. However in Table 5.5 we still achieve competitive results compared with prior art, showing our approach is not only effective to dense crowd but also generals well to the images with sparse pedestrians.

To gain further understanding of the proposed approach, we conduct detailed comparison experiments with the recent state-of-the-art CSRNet (Li et al. 2018) on ShanghaiTech part-B, where test images are divided into ten groups according to the increasing number of people in each image. It can be observed from Figure. 5.4 (a) that our method outperforms the CSRNet across most data splits, demonstrating the robustness and the effectiveness of the proposed approach. We further visualize a failure case from the last data split in Figure 5.4 (b). We keep the depth decoder at testing and save the depth predictions. As shown in the second column of Figure 5.4 (b), we found that the depth map for the sample image failed to properly depict the depth relationships especially for the farthest crowd in the left upper corner,

Table 5.4: Comparison with other state-of-the-art crowd counting methods on the ShanghaiTech-B dataset (Zhang et al. 2016).

Method	MAE	MSE
LBP + RR (Saunders et al. 1998)	59.1	81.7
Crowd-CNN (Zhang et al. 2015)	32.0	49.8
MCNN (Zhang et al. 2016)	26.4	41.3
Switch-CNN (Sam et al. 2017)	21.6	33.4
CP-CNN (Sindagi & Patel 2017*b*)	20.1	30.1
DecideNet (Liu, Gao, Meng & Hauptmann 2018)	20.75	29.42
ACSCP (Shen et al. 2018)	17.2	27.4
IG-CNN (Sam et al. 2018)	13.6	21.1
CSRNet (Li et al. 2018)	10.6	16.0
AT-CNN	**8.1**	**13.5**

Table 5.5: Comparison with other state-of-the-art crowd counting methods on the Mall dataset (Chen et al. 2012).

Method	MAE	MSE
SquareChn Detector (Benenson, Omran, Hosang & Schiele 2014)	20.55	439.1
R-FCN (Dai & R-fcn 2016)	6.02	5.46
Faster R-CNN (Ren et al. 2015)	5.91	6.60
Ridge Regression (Saunders et al. 1998)	3.59	19.0
MORR (Chen et al. 2012)	3.15	15.7
Count Forest (Pham et al. 2015)	4.40	2.40
Weighted VLAD (Sheng et al. n.d.)	2.41	9.12
Exemplary Density (Wang & Zou 2016)	1.82	2.74
Boosting CNN (Walach & Wolf 2016)	2.01	N/A
MoCNN (Kumagai et al. 2018)	2.75	13.4
DecideNet (Liu, Gao, Meng & Hauptmann 2018)	**1.52**	**1.90**
AT-CNN	2.28	2.90

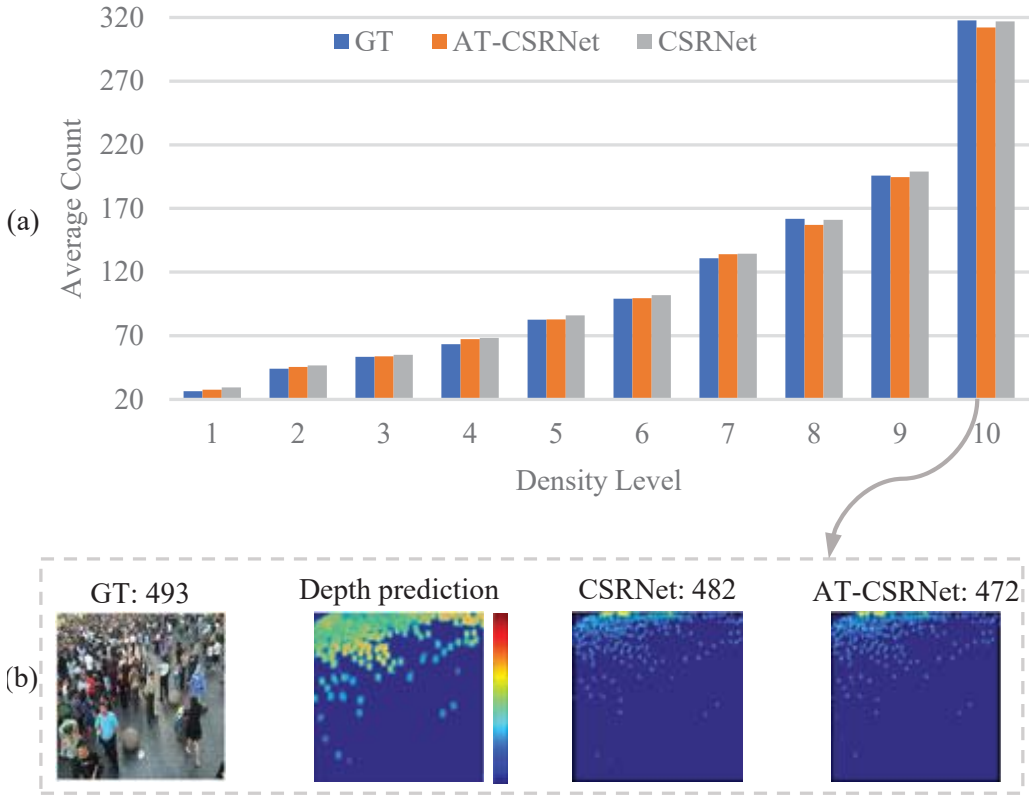

Figure 5.4: (a) Histogram: comparison of average count estimation on 10 splits of ShanghaiTech-B dataset according to the increasing number of people in each image. (b) Visualization of a failure case from the last split.

Table 5.6: Comparison with other state-of-the-art crowd counting methods on the WorldExpo'2010 dataset (Zhang et al. 2015).

Method	S1	S2	S3	S4	S5	Average
LBP + RR (Saunders et al. 1998)	13.6	59.8	37.1	21.8	23.4	31.0
Crowd-CNN (Zhang et al. 2015)	9.8	14.1	14.3	22.2	3.7	12.9
MCNN (Zhang et al. 2016)	3.4	20.6	12.9	13.0	8.1	11.6
Switch-CNN (Sam et al. 2017)	4.4	15.7	10.0	11.0	5.9	9.4
CP-CNN (Sindagi & Patel 2017*b*)	2.9	14.7	10.5	10.4	5.8	8.86
IG-CNN (Sam et al. 2018)	2.6	16.1	10.15	20.2	7.6	11.3
DecideNet (Liu, Gao, Meng & Hauptmann 2018)	2.0	13.14	8.9	17.40	4.75	9.23
CSRNet (Li et al. 2018)	2.9	11.5	8.6	16.6	3.4	8.6
ACSCP (Shen et al. 2018)	2.8	14.05	9.6	8.1	2.9	7.5
AT-CNN	1.8	13.7	9.2	8.6	3.7	**7.40**

which may lead to inaccuracy of the density estimation and hence the count result. This indicates the insufficient ability of the trained depth decoder. Considering the fact that ground truth depth maps currently used to train our model are generated by existing depth algorithms which have not been specifically adapted to crowd scenes, we guess with more accurate depth ground truth provided, the depth decoder could be better optimized and inversely benefit the base model for better results on such kind of examples.

In Figure. 5.6 we visualize the predicted density maps and the estimated counts of our method (AT-CSRNet) and the CSRNet. Overall our method generate more accurate count estimations and reserve more consistency with the crowd distributions. For instance, for the first image, the estimation of CSRNet shows inaccurate estimation in the umbrella area, however however with the learning of auxiliary segmentation task which inversely help refine the intermediate features and avoid such falsely activated density estimations in our prediction. Similar situations can be observed for other sample images.

5.4.4 Parameter Study of the Weights for Auxiliary Tasks

The weights λ_i in Equation 5.5 determines the influence of each auxiliary task on the main task, which is a key parameters in our approach. To optimize the selection of λ_i, we conduct comparative experiments with the AT-CFCN model on the ShanghaiTech-B dataset to study the influences on density estimation when parameter of λ varies. As shown in Figure. 5.5, for the depth prediction task, the MAE error decreases when the weights lies in a certain range of values. Too small weights are hard to contribute to the main tasks while too large weights will drift the feature representations and deteriorate the performances. Similar situations can be be observed for the crowd segmentation loss and the count regression loss. In our experiment, we select the weights for depth prediction loss, crowd segmentation loss and the count regression loss as 0.6, 0.04 and 1, respectively.

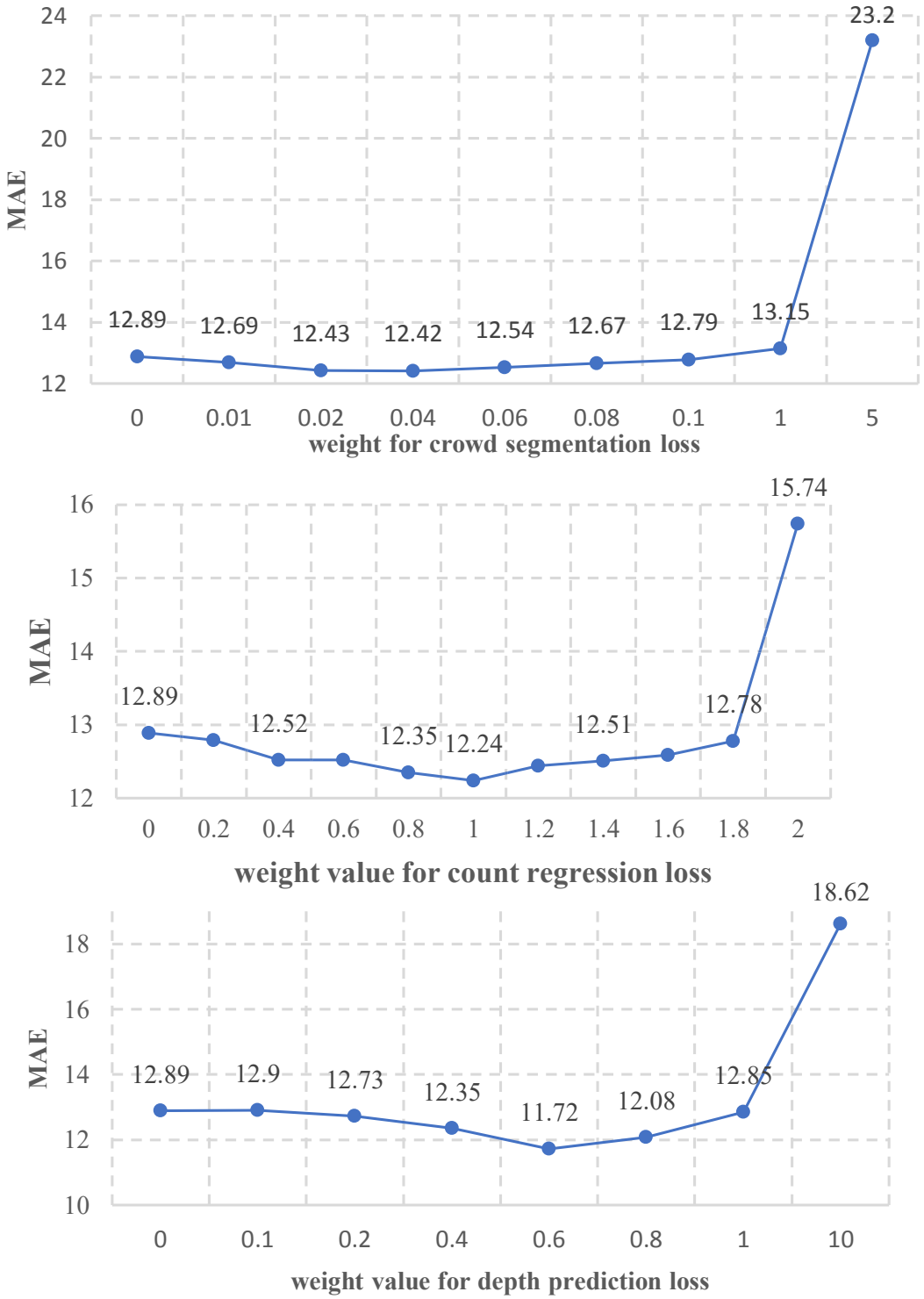

Figure 5.5: Comparison of MAE with different weight of the loss for the three auxiliary tasks on ShanghaiTech-B dataset (Zhang et al. 2016)

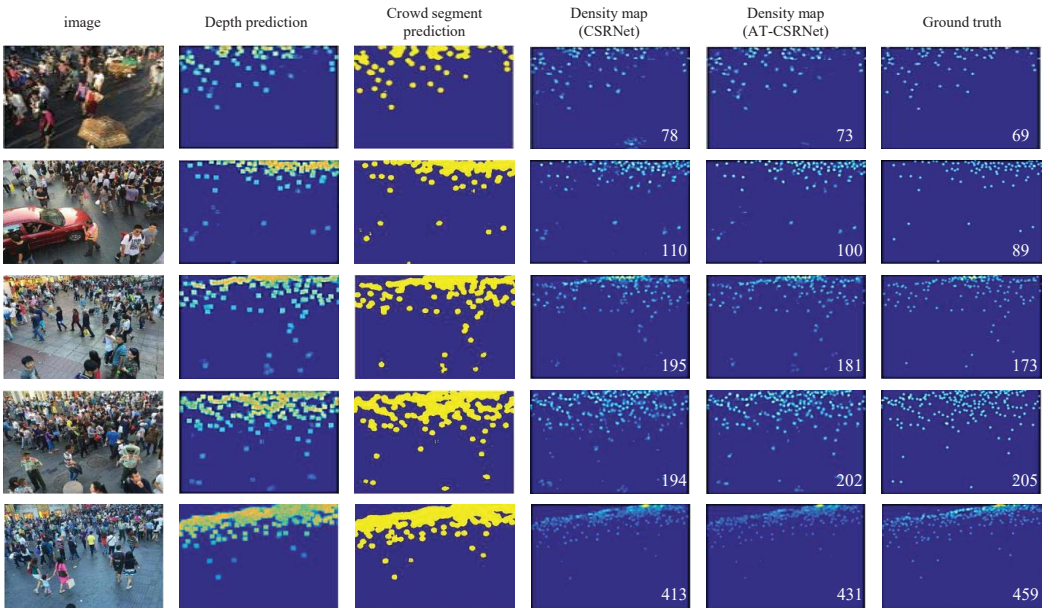

Figure 5.6: Visualization and comparison of density estimation. First column: test image; Second column: depth map predicted by the depth decoder of our method; Third Column: crowd segmentation predicted from the segment decoder of our method; Fourth column: estimated density map by CSRNet (Li et al. 2018); Fifth column: estimated density map by our method (At-CSRNet); Last column: Ground-truth density maps. Count estimation from each density map are labeled at the right corner of the corresponding prediction.

5.5 Conclusion

In this chapter, we propose to leverage the heterogeneous attributes compounded in the density map to assist crowd counting task. Specifically, the observed attributes are formulated as three auxiliary tasks to regularize the learning of the intermediate features for the main task of density estimation. Learning of the auxiliary tasks drives the embedding the geometric information, semantic information and the overall density level information, which helps the feature to be more robust against the scale variations and cluttered background. The proposed method does not incur any additional computations at inference, which gained efficiency over the general feature fusion scheme to augment the representations. Extensive experiments on multiple datasets exhibit state-of-the-art performances of the proposed method on major datasets.

This chapter demonstrates the efficacy to apply informative constraints as auxiliary tasks to improve the model capacity in perceiving desired properties. It will be very efficient for practical usage where the original model can be better optimized with minimum cost.

Chapter 6

Conclusion and Outlook

This chapter provides some conclusions for this dissertation, and discusses some possible research directions in the future.

6.1 Conclusion

This dissertation was split into 3 parts. The first part proposes a depth-embedding module with geometric priors to improve the model capacity for crowd counting. The second part illustrates a multi-stage network with implicit constraints as additional loss functions to enhance the base model. Based on the conclusion from the previous two parts, the third part elaborates the mechanism which formulates informative constraints as auxiliary tasks to fully excite the ability of a CNN model with itself.

In Chapter 3, we propose a depth embedding module as add-ons into existing networks. This module exploits essential depth cues to spatially re-calibrate the magnitude of the original features. In this way, the objects, although in the same class, will attain distinct representations according to their scales, which directly benefits the estimation of scale-aware density values. We conduct a comprehensive analysis of the effects of the depth embedding module and validate that exploiting depth cues to explicitly perceive object scale variations in convolutional neural networks improves performance.

Experimental results demonstrate the superiority of the proposed approach to the current state-of-the-art methods on four popular benchmark datasets. The success of Deem-CNN indicates that for tasks requiring awareness to the scene geometrics, it will be beneficial to consider additionally geometric priors to inform the network on the geometric variations.

The positive results in side-information injection make us curious about a parallel question on how to enhance a model without prior messages. The subsequent part of the dissertation explores this problem. Based on the observation of local inconsistency problem, we propose a constrained multi-stage Convolutional Neural Networks (CNNs) to jointly pursue locally consistent density map from two aspects. Different from most existing methods that mainly rely on the multi-column architectures of plain CNNs, we exploit a stacking formulation of plain CNNs. Benefited from the internal multi-stage learning process, the feature map could be repeatedly refined, allowing the density map to approach the ground-truth density distribution. For further refinement of the density map, we also propose a grid loss function. With finer local-region-based supervisions, the underlying model is constrained to generate locally consistent density values to minimize the training errors considering both the global and local counts accuracy. Experiments with overall significant results compared with state-of-the-art methods demonstrate the effectiveness of our approach. This chapter validates that without additional geometric prior, implicitly enhancing the model capacity with some specific design schemes and additional constraints is also beneficial.

With the observations from both two parts, the third part investigates how to fully excite the possibility of an original network to understand the scene geometry. To this end, we resort to the compound factors existing in the density prediction. Three geometric/semantic/numeric attributes essentially important to the density estimation are identified, each of which is formulated as an auxiliary task. With the multi-fold regularization effects induced by the auxiliary tasks, the backbone CNN model is driven to embed desired properties explicitly and thus gains robust representations

towards more accurate density estimation. Extensive experiments on three challenging crowd counting datasets have demonstrated the effectiveness of the proposed approach.

6.2 Short-term Outlook

The research projects shown in this dissertation are only the beginning of the story in tackling the challenges in visual crowd counting with deep neural networks. There are a few short-term research directions that the author hopes to take in the future as a continuation of the research projects presented in this dissertation.

6.2.1 Semi-supervised and Weakly-supervised Learning

For the next step research, I am interested in developing more effective solutions such as semi-supervised/weakly-supervised learning for counting or other related problems. During the PhD research, I found that for most algorithms, the amount and diversity of training data is the key to the accuracy. However, data annotation is labor-intensive especially for problems with densely pixel-wise labeling, e.g., crowd counting needs to enumerate dot annotations for every pedestrian in the image, which is very time-consuming and error-prone especially in crowded scenes. How to simplify the labeling process (e.g., only use crowd count as labels), exploit unlabeled data (e.g., use the vast amount of unlabeled web crowd images) and even learn without labels will be quite important to advance the efficiency of crowd counting algorithms. There are a few methods that have attempt to leverage the unlabeled data (Liu, van de Weijer & Bagdanov 2018) or synthetic data (Wang et al. 2019) to improve the robustness of a trained model for crowd counting, and it will be promising to further study the semi-supervised or weakly-supervised learning based on these research.

6.2.2 Model Adaption

Model adaption is practically valuable for crowd counting. As a real-world application, it is important to study the model adaption across different datasets and various scenes. This will alleviate the cost to re-train a model in a new scene and also fully exploit the prior work. How to apply a trained model on one dataset to another dataset with a minimum cost of re-training for cross-dataset estimation? Or how to effectively use a trained model with images from one scene to another unseen scene for cross-scene counting? Effective model adaption methods will significantly improve the practicability of the crowd counting research. Recent work (Shi, Zhang, Liu, Cao, Ye, Cheng & Zheng 2018) have evaluated their model performances using cross-dataset crowd counting as a case study, however there are few methods that specifically designed to tackle the cross-scene and cross-dataset counting with transfer learning techniques. How to disentangle the influence of geometry variations across different scenes into the CNN parameters might be a potential direction for the counting research.

6.2.3 Multi-view Crowd Counting

Currently most crowd counting methods are based on single-view images, whose information is limited due to severe occlusions and small sizes of far away pedestrians. How to effectively fuse multi-view information captured by multiple cameras for one scene to improve the counting accuracy is worth studying. Recent work (Zhang & Chan 2019) have shown the superiority when multi-view information is fused. This can also be further combined with the drone-based images which easily provides multi-view information of a scene.

In summary, in this dissertation the problem of how to improve the capacity of a crowd counting model is mainly focused. This is the first step to complete a practical counting system. Based on the results and conclusions, there are many following-on challenges in the generality and scalability of a

counting model that need to mine and solve.

Bibliography

Aich, S. & Stavness, I. (2017), Leaf counting with deep convolutional and deconvolutional networks, *in* '2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22-29, 2017', IEEE Computer Society, pp. 2080–2089.

Arteta, C., Lempitsky, V., Noble, J. A. & Zisserman, A. (2014), Interactive object counting, *in* 'Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part III', Springer, pp. 504–518.

Bahdanau, D., Cho, K. & Bengio, Y. (2015), Neural machine translation by jointly learning to align and translate, *in* '3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings'.

Benenson, R., Omran, M., Hosang, J. & Schiele, B. (2014), Ten years of pedestrian detection, what have we learned?, *in* 'Computer Vision - ECCV 2014 Workshops - Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part II', Springer, pp. 613–627.

Bengio, Y., Simard, P. Y. & Frasconi, P. (1994), 'Learning long-term dependencies with gradient descent is difficult', *IEEE Trans. Neural Networks* **5**(2), 157–166.

Boominathan, L., Kruthiventi, S. S. & Babu, R. V. (2016), Crowdnet: A deep convolutional network for dense crowd counting, *in* 'Proceedings

of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016', ACM, pp. 640–644.

Brostow, G. J. & Cipolla, R. (2006), Unsupervised bayesian detection of independent motion in crowds, *in* '2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2006, New York, NY, USA, 17-22 June 2006', Vol. 1, IEEE Computer Society, pp. 594–601.

Cao, X., Wang, Z., Zhao, Y. & Su, F. (2018), Scale aggregation network for accurate and efficient crowd counting, *in* 'Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part V', Springer, pp. 734–750.

Carreira, J., Agrawal, P., Fragkiadaki, K. & Malik, J. (2016), Human pose estimation with iterative error feedback, *in* '2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016', IEEE Computer Society, pp. 4733–4742.

Chan, A. B., Liang, Z.-S. J. & Vasconcelos, N. (2008), Privacy preserving crowd monitoring: Counting people without people models or tracking, *in* '2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2008), Anchorage, Alaska, USA, 24-26 June 2008', IEEE Computer Society, pp. 1–7.

Chan, A. B. & Vasconcelos, N. (2012), 'Counting people with low-level features and bayesian regression', *IEEE Transactions on Image Processing* **21**(4), 2160–2177.

Change Loy, C., Gong, S. & Xiang, T. (2013), From semi-supervised to transfer counting of crowds, *in* 'IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013', IEEE Computer Society, pp. 2256–2263.

Chen, K., Gong, S., Xiang, T. & Change Loy, C. (2013), Cumulative attribute space for age and crowd density estimation, *in* '2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013', IEEE Computer Society, pp. 2467–2474.

Chen, K., Loy, C. C., Gong, S. & Xiang, T. (2012), Feature mining for localised crowd counting, *in* 'British Machine Vision Conference, BMVC 2012, Surrey, UK, September 3-7, 2012', Vol. 1, BMVA Press, p. 3.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K. & Yuille, A. L. (2017), 'Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs', *IEEE transactions on pattern analysis and machine intelligence* **40**(4), 834–848.

Dai, K. J. & R-fcn, Y. L. (2016), Object detection via region-based fully convolutional networks, *in* 'Annual Conference on Neural Information Processing Systems, NIPS 2016, Barcelona, Spain, December 5-10, 2016', pp. 379–387.

Dalal, N. & Triggs, B. (2005), Histograms of oriented gradients for human detection, *in* '2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, San Diego, CA, USA, 20-26 June 2005', Vol. 1, IEEE Computer Society, pp. 886–893.

Dollar, P., Wojek, C., Schiele, B. & Perona, P. (2012), 'Pedestrian detection: An evaluation of the state of the art', *IEEE transactions on pattern analysis and machine intelligence* **34**(4), 743–761.

Elkahky, A. M., Song, Y. & He, X. (2015), A multi-view deep learning approach for cross domain user modeling in recommendation systems, *in* 'Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy, May 18-22, 2015', ACM, pp. 278–288.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D. & Ramanan, D. (2010), 'Object detection with discriminatively trained part-based mod-

els', *IEEE transactions on pattern analysis and machine intelligence* **32**(9), 1627–1645.

Fiaschi, L., Köthe, U., Nair, R. & Hamprecht, F. A. (2012), Learning to count with regression forest and structured labels, *in* 'Proceedings of the 21st International Conference on Pattern Recognition, ICPR 2012, Tsukuba, Japan, November 11-15, 2012', IEEE Computer Society, pp. 2685–2688.

Ge, W. & Collins, R. T. (2009*a*), Evaluation of sampling-based pedestrian detection for crowd counting, *in* '2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance', IEEE, pp. 1–7.

Ge, W. & Collins, R. T. (2009*b*), Marked point processes for crowd counting, *in* '2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2009, 20-25 June 2009, Miami, Florida, USA', IEEE Computer Society, pp. 2913–2920.

Girshick, R. (2015), Fast r-cnn, *in* '2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015', IEEE Computer Society, pp. 1440–1448.

Girshick, R., Donahue, J., Darrell, T. & Malik, J. (2014), Rich feature hierarchies for accurate object detection and semantic segmentation, *in* '2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014', IEEE Computer Society, pp. 580–587.

Goodfellow, I., Bengio, Y. & Courville, A. (2016), *Deep learning*, MIT press.

Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H. & Schmidhuber, J. (2009), 'A novel connectionist system for unconstrained handwriting recognition', *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(5), 855–868.

Graves, A., Mohamed, A.-r. & Hinton, G. (2013), Speech recognition with deep recurrent neural networks, *in* 'IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013', IEEE, pp. 6645–6649.

Guerrero-Gómez-Olmedo, R., Torre-Jiménez, B., López-Sastre, R., Maldonado-Bascón, S. & Oñoro-Rubio, D. (2015), Extremely overlapping vehicle counting, *in* 'Pattern Recognition and Image Analysis - 7th Iberian Conference, IbPRIA 2015, Santiago de Compostela, Spain, June 17-19, 2015, Proceedings', Springer, pp. 423–431.

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Kingsbury, B. et al. (2012), 'Deep neural networks for acoustic modeling in speech recognition', *IEEE Signal processing magazine* **29**.

Hossain, M., Hosseinzadeh, M., Chanda, O. & Wang, Y. (2019), Crowd counting using scale-aware attention networks, *in* 'IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, January 7-11, 2019', IEEE, pp. 1280–1288.

Huang, P.-S., He, X., Gao, J., Deng, L., Acero, A. & Heck, L. (2013), Learning deep structured semantic models for web search using clickthrough data, *in* '22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013', ACM, pp. 2333–2338.

Idrees, H., Saleemi, I., Seibert, C. & Shah, M. (2013), Multi-source multi-scale counting in extremely dense crowd images, *in* '2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013', IEEE Computer Society, pp. 2547–2554.

Kalchbrenner, N. & Blunsom, P. (2013), Recurrent continuous translation models, *in* 'Proceedings of the 2013 Conference on Empirical Methods

in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL', ACL, pp. 1700–1709.

Kang, D. & Chan, A. (2018), Crowd counting by adaptively fusing predictions from an image pyramid, *in* 'British Machine Vision Conference 2018, BMVC 2018, Northumbria University, Newcastle, UK, September 3-6, 2018', BMVA Press, p. 89.

Kang, D., Dhar, D. & Chan, A. (2017), Incorporating side information by adaptive convolution, *in* 'Annual Conference on Neural Information Processing Systems, NIPS, 2017, 4-9 December 2017, Long Beach, CA, USA', pp. 3868–3878.

Kang, K. & Wang, X. (2014), 'Fully convolutional neural networks for crowd segmentation', *CoRR* **abs/1411.4464**.

Kong, D., Gray, D. & Tao, H. (2006), A viewpoint invariant approach for crowd counting, *in* '18th International Conference on Pattern Recognition, ICPR 2006, Hong Kong, China, 20-24 August 2006', IEEE Computer Society, pp. 1187–1190.

Kong, S. & Fowlkes, C. (2018), 'Recurrent scene parsing with perspective understanding in the loop', pp. 956–965.

Koprinska, I. & Carrato, S. (2001), 'Temporal video segmentation: A survey', *Signal processing: Image communication* **16**(5), 477–500.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012), Imagenet classification with deep convolutional neural networks, *in* 'Annual Conference on Neural Information Processing Systems, NIPS 2012, Lake Tahoe, Nevada, United States.', pp. 1097–1105.

Kumagai, S., Hotta, K. & Kurita, T. (2018), 'Mixture of counting cnns', *Machine Vision and Applications* **29**(7), 1119–1126.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. & Jackel, L. D. (1989), 'Backpropagation applied to handwritten zip code recognition', *Neural computation* **1**(4), 541–551.

LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E. & Jackel, L. D. (1990), Handwritten digit recognition with a back-propagation network, *in* 'Advances in Neural Information Processing Systems 2, NIPS Conference, Denver, Colorado, USA, November 27-30, 1989', Morgan Kaufmann, pp. 396–404.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. et al. (1998), 'Gradient-based learning applied to document recognition', *Proceedings of the IEEE* **86**(11), 2278–2324.

Lee, C.-Y., Xie, S., Gallagher, P., Zhang, Z. & Tu, Z. (2015), Deeply-supervised nets, *in* 'Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015', JMLR.org, pp. 562–570.

Lempitsky, V. & Zisserman, A. (2010), Learning to count objects in images, *in* '24th Annual Conference on Neural Information Processing Systems 2010, 6-9 December 2010, Vancouver, British Columbia, Canada.', Curran Associates, Inc., pp. 1324–1332.

Li, M., Zhang, Z., Huang, K. & Tan, T. (2008), Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection, *in* '19th International Conference on Pattern Recognition ICPR 2008, Tampa, Florida, USA, December 8-11, 2008', IEEE Computer Society, pp. 1–4.

Li, Y., Zhang, X. & Chen, D. (2018), Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes, *in* '2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018', IEEE Computer Society, pp. 1091–1100.

Lin, S.-F., Chen, J.-Y. & Chao, H.-X. (2001), 'Estimation of number of people in crowded scenes using perspective transformation', *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* **31**(6), 645–654.

Lin, Z. & Davis, L. S. (2010), 'Shape-based human detection and segmentation via hierarchical part-template matching', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(4), 604–618.

Liu, F., Shen, C., Lin, G. & Reid, I. (2016), 'Learning depth from single monocular images using deep convolutional neural fields', *IEEE transactions on pattern analysis and machine intelligence* **38**(10), 2024–2039.

Liu, J., Gao, C., Meng, D. & Hauptmann, A. G. (2018), Decidenet: Counting varying density crowds through attention guided detection and density estimation, *in* '2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018', IEEE Computer Society, pp. 5197–5206.

Liu, W., Salzmann, M. & Fua, P. (2019), Context-aware crowd counting, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019', Computer Vision Foundation / IEEE, pp. 5099–5108.

Liu, X., van de Weijer, J. & Bagdanov, A. D. (2018), Leveraging unlabeled data for crowd counting by learning to rank, *in* '2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018', IEEE Computer Society, pp. 7661–7669.

Long, J., Shelhamer, E. & Darrell, T. (2015), Fully convolutional networks for semantic segmentation, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015', IEEE Computer Society, pp. 3431–3440.

Loy, C. C., Chen, K., Gong, S. & Xiang, T. (2013), Crowd counting and profiling: Methodology and evaluation, *in* 'Modeling, Simulation and Visual Analysis of Crowds - A Multidisciplinary Perspective', Vol. 11, Springer, pp. 347–382.

Ma, C., Yang, X., Zhang, C. & Yang, M.-H. (2015), Long-term correlation tracking, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015', IEEE Computer Society, pp. 5388–5396.

Marana, A. N., Cavenaghi, M. A., Ulson, R. S. & Drumond, F. (2005), Real-time crowd density estimation using images, *in* 'Advances in Visual Computing, First International Symposium, ISVC 2005, Lake Tahoe, NV, USA, December 5-7, 2005, Proceedings', Springer, pp. 355–362.

Miao, Y., Han, J., Gao, Y. & Zhang, B. (2019), 'St-cnn: Spatial-temporal convolutional neural network for crowd counting in videos', *Pattern Recognition Letters* **125**, 113–118.

Newell, A., Yang, K. & Deng, J. (2016), Stacked hourglass networks for human pose estimation, *in* 'Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII', Springer, pp. 483–499.

Noh, H., Hong, S. & Han, B. (2015), Learning deconvolution network for semantic segmentation, *in* '2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015', IEEE Computer Society, pp. 1520–1528.

Onoro-Rubio, D. & López-Sastre, R. J. (2016), Towards perspective-free object counting with deep learning, *in* 'Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VII', Springer, pp. 615–629.

Oñoro-Rubio, D., Niepert, M. & López-Sastre, R. J. (2018), Learning short-cut connections for object counting, *in* 'British Machine Vision Conference 2018, BMVC 2018, Northumbria University, Newcastle, UK, September 3-6, 2018', BMVA Press, p. 262.

Ouyang, W. & Wang, X. (2013), Joint deep learning for pedestrian detection, *in* 'IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013', IEEE Computer Society, pp. 2056–2063.

Pham, V.-Q., Kozakaya, T., Yamaguchi, O. & Okada, R. (2015), Count forest: Co-voting uncertain number of targets using random forest for crowd density estimation, *in* '2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015', IEEE Computer Society, pp. 3253–3261.

Pinheiro, P. H. & Collobert, R. (2014), Recurrent convolutional neural networks for scene labeling, *in* 'Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014', JMLR.org, pp. 82–90.

Rabaud, V. & Belongie, S. (2006), Counting crowded moving objects, *in* '2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2006, New York, NY, USA, 17-22 June 2006', Vol. 1, IEEE Computer Society, pp. 705–711.

Ren, S., He, K., Girshick, R. & Sun, J. (2015), Faster r-cnn: Towards real-time object detection with region proposal networks, *in* 'Annual Conference on Neural Information Processing Systems 2015, Montreal, Quebec, Canada, December 7-12, 2015', pp. 91–99.

Ryan, D., Denman, S., Fookes, C. & Sridharan, S. (2009), Crowd counting using multiple local features, *in* 'Digital Image Computing: Techniques and Applications, 2009. DICTA'09.', IEEE, pp. 81–88.

Saleh, S. A. M., Suandi, S. A. & Ibrahim, H. (2015), 'Recent survey on crowd density estimation and counting for visual surveillance', *Engineering Applications of Artificial Intelligence* **41**, 103–114.

Sam, D. B., Sajjan, N. N., Babu, R. V. & Srinivasan, M. (2018), 'Divide and grow: Capturing huge diversity in crowd images with incrementally growing cnn', pp. 3618–3626.

Sam, D. B., Surya, S. & Babu, R. V. (2017), Switching convolutional neural network for crowd counting, *in* '2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017', IEEE Computer Society, pp. 4031–4039.

Saunders, C., Gammerman, A. & Vovk, V. (1998), 'Ridge regression learning algorithm in dual variables', pp. 515–521.

Saxena, A., Sun, M. & Ng, A. Y. (2009), 'Make3d: Learning 3d scene structure from a single still image', *IEEE transactions on pattern analysis and machine intelligence* **31**(5), 824–840.

Schwarz, M., Schulz, H. & Behnke, S. (2015), Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features, *in* 'IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015', IEEE, pp. 1329–1335.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R. & LeCun, Y. (2014), 'Overfeat: Integrated recognition, localization and detection using convolutional networks'.

Shen, Z., Xu, Y., Ni, B., Wang, M., Hu, J. & Yang, X. (2018), Crowd counting via adversarial cross-scale consistency pursuit, *in* '2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018', IEEE Computer Society, pp. 5245–5254.

Sheng, B., Shen, C., Lin, G., Li, J., Yang, W. & Sun, C. (n.d.), 'Crowd counting via weighted vlad on dense attribute feature maps', *IEEE Transactions on Circuits and Systems for Video Technology* **28**(8), 1788–1797.

Shi, M., Yang, Z., Xu, C. & Chen, Q. (n.d.), Revisiting perspective information for efficient crowd counting, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019', Computer Vision Foundation / IEEE.

Shi, Z., Zhang, L., Liu, Y., Cao, X., Ye, Y., Cheng, M.-M. & Zheng, G. (2018), Crowd counting with deep negative correlation learning, *in* '2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018', IEEE Computer Society, pp. 5382–5390.

Sidla, O., Lypetskyy, Y., Brandle, N. & Seer, S. (2006), Pedestrian detection and tracking for counting applications in crowded situations, *in* 'IEEE International Conference on Video and Signal Based Surveillance, AVSS'06, Sydney, Australia, 22-24 November 2006', IEEE Computer Society, pp. 70–70.

Silberman, N., Hoiem, D., Kohli, P. & Fergus, R. (2012), Indoor segmentation and support inference from rgbd images, *in* 'Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V', Springer, pp. 746–760.

Simonyan, K. & Zisserman, A. (2015), Very deep convolutional networks for large-scale image recognition, *in* '3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings'.

Sindagi, V. A. & Patel, V. M. (2017*a*), 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, AVSS 2017, Lecce, Italy, August 29 - September 1, 2017', IEEE, IEEE Computer Society, pp. 1–6.

Sindagi, V. A. & Patel, V. M. (2017*b*), Generating high-quality crowd density maps using contextual pyramid cnns, *in* 'IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017', IEEE Computer Society, pp. 1879–1888.

Sindagi, V. A. & Patel, V. M. (2018), 'A survey of recent advances in cnn-based single image crowd counting and density estimation', *Pattern Recognition Letters* **107**, 3–16.

Stauffer, C. & Grimson, W. E. L. (1999), Adaptive background mixture models for real-time tracking, *in* '1999 Conference on Computer Vision and Pattern Recognition, CVPR '99, Ft. Collins, CO, USA, 23-25 June 1999', IEEE Computer Society, pp. 246–252.

Subburaman, V. B., Descamps, A. & Carincotte, C. (2012), Counting people in the crowd using a generic head detector, *in* 'Ninth IEEE International Conference on Advanced Video and Signal-Based Surveillance, AVSS 2012, Beijing, China, September 18-21, 2012', IEEE Computer Society, pp. 470–475.

Tompson, J. J., Jain, A., LeCun, Y. & Bregler, C. (2014), Joint training of a convolutional network and a graphical model for human pose estimation, *in* 'Annual Conference on Neural Information Processing Systems 2014, Montreal, Quebec, Canada, December 8-13 2014', pp. 1799–1807.

Topkaya, I. S., Erdogan, H. & Porikli, F. (2014), Counting people by clustering person detector outputs, *in* '11th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2014, Seoul, South Korea, August 26-29, 2014', IEEE Computer Society, pp. 313–318.

Varior, R. R., Shuai, B., Tighe, J. & Modolo, D. (2019), 'Scale-aware attention network for crowd counting', *CoRR* **abs/1901.06026**.

Vedaldi, A. & Lenc, K. (2015), Matconvnet, convolutional neural networks for matlab, *in* 'Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, MM '15, Brisbane, Australia, October 26 - 30, 2015', ACM, pp. 689–692.

Viola, P. & Jones, M. J. (2004), 'Robust real-time face detection', *International journal of computer vision* **57**(2), 137–154.

Walach, E. & Wolf, L. (2016), Learning to count with cnn boosting, *in* 'Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II', Springer, pp. 660–676.

Wang, C., Zhang, H., Yang, L., Liu, S. & Cao, X. (2015), Deep people counting in extremely dense crowds, *in* 'Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, MM '15, Brisbane, Australia, October 26 - 30, 2015', ACM Press, pp. 1299–1302.

Wang, Q., Gao, J., Lin, W. & Yuan, Y. (2019), Learning from synthetic data for crowd counting in the wild, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019', Computer Vision Foundation / IEEE, pp. 8198–8207.

Wang, Y. & Zou, Y. (2016), Fast visual object counting via example-based density estimation, *in* '2016 IEEE International Conference on Image Processing, ICIP 2016, Phoenix, AZ, USA, September 25-28, 2016', IEEE, pp. 3653–3657.

Wu, B. & Nevatia, R. (2005), Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors, *in* '10th IEEE International Conference on Computer Vision,

ICCV 2005, Beijing, China, 17-20 October 2005', Vol. 1, IEEE Computer Society, pp. 90–97.

Wu, X., Zheng, Y., Ye, H., Hu, W., Yang, J. & He, L. (2019), Adaptive scenario discovery for crowd counting, *in* 'IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019', IEEE, pp. 2382–2386.

Xie, W., Noble, J. A. & Zisserman, A. (2018), 'Microscopy cell counting and detection with fully convolutional regression networks', *Computer methods in biomechanics and biomedical engineering: Imaging & Visualization* **6**(3), 283–292.

Xiong, F., Shi, X. & Yeung, D. (2017), Spatiotemporal modeling for crowd counting in videos, *in* 'IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017', IEEE Computer Society, pp. 5161–5169.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R. & Bengio, Y. (2015), Show, attend and tell: Neural image caption generation with visual attention, *in* 'Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015', JMLR.org, pp. 2048–2057.

Zhang, C., Li, H., Wang, X. & Yang, X. (2015), Cross-scene crowd counting via deep convolutional neural networks, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015', IEEE Computer Society, pp. 833–841.

Zhang, C. & Woodland, P. C. (2015), Parameterised sigmoid and relu hidden activation functions for dnn acoustic modelling, *in* 'INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015', ISCA, pp. 3224–3228.

Zhang, L., Shi, M. & Chen, Q. (2018), Crowd counting via scale-adaptive convolutional neural network, *in* '2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018, Lake Tahoe, NV, USA, March 12-15, 2018', IEEE Computer Society, pp. 1113–1121.

Zhang, Q. & Chan, A. B. (2019), Wide-area crowd counting via ground-plane density maps and multi-view fusion cnns, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019', Computer Vision Foundation / IEEE, pp. 8297–8306.

Zhang, S., Wu, G., Costeira, J. P. & Moura, J. M. (2017*a*), Understanding traffic density from large-scale web camera data, *in* '2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017', IEEE Computer Society, pp. 4264–4273.

Zhang, S., Wu, G., Costeira, J. P. & Moura, J. M. F. (2017*b*), Fcn-rlstm: Deep spatio-temporal neural networks for vehicle counting in city cameras, *in* 'IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017', IEEE Computer Society, pp. 3687–3696.

Zhang, Y., Zhou, D., Chen, S., Gao, S. & Ma, Y. (2016), Single-image crowd counting via multi-column convolutional neural network, *in* '2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016', IEEE Computer Society, pp. 589–597.

Zhao, M., Zhang, J., Zhang, C. & Zhang, W. (2018), Towards locally consistent object counting with constrained multi-stage convolutional neural networks, *in* 'Computer Vision - ACCV 2018 - 14th Asian Conference on Computer Vision, Perth, Australia, December 2-6, 2018, Revised Selected Papers, Part VI', Springer, pp. 247–261.

Zhao, T., Nevatia, R. & Wu, B. (2008), 'Segmentation and tracking of multiple humans in crowded environments', *IEEE transactions on pattern analysis and machine intelligence* **30**(7), 1198–1211.

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C. & Torr, P. H. S. (2015), Conditional random fields as recurrent neural networks, *in* '2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015', IEEE Computer Society, pp. 1529–1537.

Zhou, J., Pei, H. & Wu, H. (2018), Early warning of human crowds based on query data from baidu maps: analysis based on shanghai stampede, *in* 'Big Data Support of Urban Planning and Management', Springer, pp. 19–41.

Zhou, Y.-T. & Chellappa, R. (1988), Computation of optical flow using a neural network, *in* 'Proceedings of International Conference on Neural Networks, ICNN'88, San Diego, CA, USA, July 24-27, 1988', IEEE, pp. 71–78.