https://doi.org/10.2352/EI.2023.35.9.IPAS-293

CROWD COUNTING USING DEEP LEARNING BASED HEAD DETECTION

Maryam Hassan[1], *Farhan Hussain*[3], *Sultan Daud Khan*[2], *Mohib Ullah*[2], *Mudassar Yamin, Habib Ullah* [2]

[1] NUST College of Electrical & Mechanical Engineering, Rawalpindi, Pakistan
[2] Department of computer science, National university of technology, Islamabad, Pakistan
[3] Norwegian University of Science and Technology, 2815 Gjøvik, Norway
[4] Faculty of Science and Technology (REALTEK), Norwegian University of Life Sciences (NMBU), 1430 Ås, Norway

ABSTRACT

Scale invariance and high miss detection rates for small objects are some of the challenging issues for object detection and often lead to inaccurate results. This research aims to provide an accurate detection model for crowd counting by focusing on human head detection from natural scenes acquired from publicly available datasets of Casablanca, Hollywood-Heads and Scut-head. In this study, we tuned a yolov5, a deep convolutional neural network (CNN) based object detection architecture, and then evaluated the model using mean average precision (mAP) score, precision, and recall. The transfer learning approach is used for fine-tuning the architecture. Training on one dataset and testing the model on another leads to inaccurate results due to different types of heads in different datasets. Another main contribution of our research is combining the three datasets into a single dataset, including every kind of head that is medium, large and small. From the experimental results, it can be seen that this yolov5 architecture showed significant improvements in small head detections in crowded scenes as compared to the other baseline approaches, such as the Faster R-CNN and VGG-16-based SSD MultiBox Detector.

Index Terms— Object detection, Convolutional Neural Networks, Deep Learning, YOLO, Yolov5, Precision, Mean average Precision.

1. INTRODUCTION

Increase in population has led to overcrowding in many places such as parades, station departures and entrances, political protests, and strikes etc. These situations raise a number of security concerns. The key task in carrying out crowd surveillance is accurate crowd counts. For applications such as video surveillance and traffic management, crowd counting is essential. Due to strong occlusions, scene perspective distortions, and a wide range of crowd distributions, crowd counting is a difficult process. Vision based security systems are becoming increasingly common in modern society. Mostly every public area has its own security system, as it is vital to use such systems to protect public security. Due to the availability of low-cost security video cameras and high-speed computer networks, it is now technologically viable and financially reasonable to install such a system for crime reduction and detection [1]. The global population has been rapidly growing in recent decades. As a result of global urban population growth, the crowd problem has become more prevalent. In the field of crowd analysis [2], automated crowd counting is a popular issue. Many notable articles have been published in this topic over the last few decades, and it has been and continues to be a difficult problem for autonomous visual surveillance for many years [3, 4]. Due to the advent of autonomous vehicles, smart video monitoring, facial detection, and a variety of people counting applications, robust and precise object detection models are in high demand. Many practical applications of computer vision rely significantly on object detection, such as human face identification, pedestrian detection, vehicle detection, and video surveillance. One of the most important tasks for crowd counting is Human Head detection in any scenario. Human detection is essential in a variety of real-world applications, including enhanced human-machine interactions, video surveillance, and crowd analysis. The most common approaches used are regression, segmentation, image processing, machine learning techniques, counters and sensor-based models. Initially, hand-crafted features were employed in machine learning algorithms for computer vision problems. In comparison to learnt features, these features are less reliable and discriminative. Deep learning approaches have recently been used to successfully apply features learning to major computer vision problems such as segmentation [5, 6], classification [7, 8], detection and identification [9, 10]. For many years, deep learning approaches have consistently won classification and detection contests such as ImageNet [11]. Several approaches have been presented for estimating the persons in a picture or video. Scientists and researchers initially developed fundamental ML and computer vision algorithms such

as density-based techniques, regression and object detection to estimate crowd density and density maps [12].

An implementation of detection-based object counting may be done using any state of art object detection approach, such as Faster RCNN or Yolo, in which the model finds the target items in the input images. To retrieve the count, we may return the count of the detected objects (the bounding boxes). This topic, in particular, has drawn the attention of the scientific community for automated identification of aberrant crowd behavior during public events [13]. represented three-dimensional features to represent the human body, and a random method to compute the number of persons [14]. To retrieve head features, several studies employed the Haar wavelet transform [15]. A research was performed in 2013 which presented a crowd counting model using Dirichlet Process Mixture Model(DPMM) in parallel with clustering, local clusters of people were acquired and the proposed measure allowed them to estimate the number of persons in the clusters. DPMMs allow for the totally automatic detection and tracking of an unknown and variable number of objects without initial tagging. This enabled them to use person detectors without segmenting distinct persons one by one [16].This technique provides acceptable detection in sparse environments, however it fails to perform accurate in crowded scenarios with interference and congestion even in surveillance applications where image resolution influences accuracy. Without employing explicit object segmentation or tracking, in 2008 a study was carried out which proposed a privacy-preserving approach for assessing the size of inhomogeneous crowds full of individuals traveling in separate directions. From each segmented region, a collection of simple integrative features is retrieved, and the correlation among features and the number of individuals per segment is computed using Gaussian Process regression [17]. Some researchers in 2013, proposed the improved method of crowd counting using regression. To highlight the feature set's describable potential, we suggest a new low-level feature, the number of corner points. Then, to improve the performance of the suggested algorithm, they introduced a fusion scheme combining relevance vector regression (RVR) and Gaussian process regression (GPR) [18]. In 2018 a research suggested that network of generative adversaries can produce high quality crowd density maps of various crowd density scenarios. They used the discriminator's adverse loss to increase the efficiency of the estimated density map that is essential to predict crowd numbers properly. Multiple aspects of the hierarchy from the crowd image can be extracted [19]. The density-based crowd counting approaches provide density values that are estimated using low-level features including pixels or areas, which resolves the disadvantage of regression-based methods while simultaneously preserving location information. Density estimation methods can integrate spatial information and build a mapping link between object characteristics and density maps. FF-CNN (Feature Fusion of Convolutional Neural Network), a deep convolutional neural network technique based on feature fusion was presented in 2018. Before including the head count, the proposed FF-CNN localized the crowd image to its crowd density map. High-quality density maps that served as ground truths for network training were created using geometry adaptive kernels. The deconvolution approach was utilized to combine high and low-level functions for joint optimization, and two loss functions were employed: the loss of density maps and the absolute count loss. [20]. [21] presented a layered technique in which training is performed in stages. They added CNNs continuously, so that each new CNN is trained to estimate the residual error of the previous prediction. Following the training of the first CNN, the next CNN is trained on the gap between the estimation and the ground reality. The procedure is then repeated for the third CNN. This research paper uses yolov5 for real-time object detection of objects which outperforms all the previous versions of yolo in terms of training time and speed. YOLO (You Only Look Once) is one of the most prominent algorithms which performs real time detection of objects with highest accuracy.

2. MODELS'S ARCHITECTURE

We have used the YOLOv5 version for training our model due to three reasons. First, in Yolov5 architecture, the CSP-Net [22] network was integrated into the Darknet which created CSPDarknet [22]. CSPNet handles the problems related to gradient information and incorporates gradient changes into the feature map which decrease models' parameters and FLOPS (floating point operations per second). It decreases the model size and when trained on ImageNet database, it reduces the computations by 20% as compared to the other state-of-art detectors [22]. Second, to improve information flow, the Yolov5 used a path aggregation. network (PANet) [23] as its neck. PANet uses a novel feature pyramid network (FPN) topology with an improved bottom-up path, which increases low-level feature propagation. Simultaneously, adaptive feature pooling, which connects the feature grid and all feature levels, is used to extract useful information in each feature level and then it propagates directly to the next subnetwork. PANet improves the utilization of accurate localization signals in lower layers, which obviously improves the object's location accuracy. Third, the Yolo layer i-e the head of Yolov5, creates three distinct sizes (18x18, 36x 36, 72 x72) of feature maps to provide multi-scale prediction, allowing the model to handle smaller, medium, and large objects. In Yolov5 official code, there are 4 object detection models namely Yolov5s, Yolov5m, Yolov5l and Yolov5x. We have used the yolov5vs as it is the smallest network. Like other single-stage detectors, YOLO v5 has three important parts which are:

1 . Backbone.

2 . Neck.

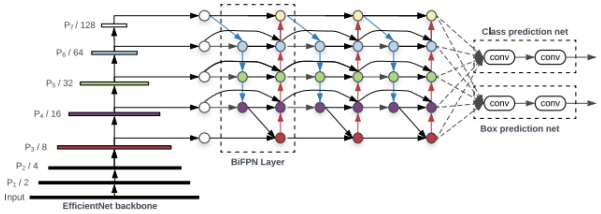

Fig. 1: Block diagram of pre-processing with an adaptive threshold technique

3 . Prediction.

2.1. Backbone

Model Backbone is mostly used to extract key features from an input image. Backbone is divided into these segments i.e., Focus and CSP structure. The focus structure basically performs the slicing operation, and there is no such structure in previous yolo versions. The focus structure takes an image as an input and slice it into multiple feature maps. We have used the Yolov5s architecture. Suppose the image of 608*608*3 shape is provided as input into the Focus structure of Yolo5s. It will first create a 304*304*12 feature map by concatenating all the feature maps. Then it is fed to convolution layer which performs convolution operation of 32 kernels to produce 304*304*32 feature map.

Backbone of YOLO v5 uses CSPNet to extract rich and meaningful features from an input image. With deeper networks, CSPNet has shown a considerable reduction in processing time. Deeper neural networks have been shown to be more powerful in feature extraction which brings up more computations and increased training time. Thus, making object detection tasks unaffordable. On the other hand, the accuracy of neural networks decreases on reducing layers. The CSPNet basically strengthens the feature learning capability of convolutional neural networks. Therefore, high accuracy can be obtained with reduced computations. Yolov5 toYolov5s network has two CSP structures i.e., CSP1X and CSP2X. CSP1 X structure is used in the Backbone network, whereas the CSP2 X structure is used in the Neck. Because Backbone has five CSP modules and the input image is 608*608, the feature map change rule is as follows: 608-¿304-¿152-¿76-¿38-¿19. A 19*19 feature map is obtained after 5 CSP modules. In Backbone, the author only employs the Mish activation function, while the Leaky relu activation function is employed behind the network. Backbone also includes an important module i.e., SPP. In the SPP module, the author uses the maximum pooling method with kernels of sizes k=1*1,5*5,9*9,13*13, and then performs concatenation operation on feature maps of different scales.

Table 1: Comparison of the results obtained from merge dataset with other datasets.

Dataset	Training mAP (%)	Casabalanca mAP (%)	Hollywood mAP (%)	Scuthead mAP (%)
Casabalanca	0.83	83.87	70.92	6.0
Hollywood head	0.98	31.85	56.00	0.0
Scuthead	0.82-	30.70	11.25	66.85
Merge	0.81	72.7	75.50	78.0

Table 2: Dataset Description

Datasets	Images	Pixels	Head size
Casablanca	1466	464x464	All mostly medium
Hollywood heads	224,740	384x864, 640x864,480x864	Various sizes
Scuthead (Part A)	67321	384x640	Small
Merge dataset	15466	Variable	Small, medium, large

2.2. Model Neck

The Model Neck creates feature pyramids which detects the small and occluded objects accurately which is quite challenging. Feature pyramids aid models in generalizing successfully when it comes to object scaling. These pyramids are used for identification of the same object appearing in different sizes and scales. They are quite beneficial in assisting models to perform effectively on previously unseen data. Other models, such as FPN, BiFPN, and PANet, employ various sorts of feature pyramid approaches. The FPN and PAN module is used as a neck in YOLO v5 to generate feature pyramids. FPN is made up of two pathways: bottom-up and top-down. For feature extraction, it uses the standard convolutional network.

2.3. Model Head

The model Head is mostly responsible for the final detection step. It uses anchor boxes to construct the final vectors consisting of output class along with class probabilities objectiveness scores and parameters defining the boundary box coordinates. Head is the last part of object detection model. It takes feature maps generated from neck as an input and performs the prediction step and gives the co-ordinates of the bounding box.

3. RESULTS AND DISCUSSIONS

We have evaluated the performance of our model on the three publicly available datasets which are Casablanca, Scuthead, and Hollywood Heads dataset. Indoor, outdoor, and crowded scenarios are all included in these datasets. Aside from that, there are many other head sizes, such as small, medium, very small, and very huge. All these datasets are annotated and the dimensions of bounding boxes are provided in the form of ground truth values. Table 2 shows the brief description of the datasets used. The detailed description is discussed below in detail.

3.1. Casablanca

The Casablanca dataset includes pictures from the film Casablanca. There are 1466 frames in all, with annotated head bounding boxes. The Hollywood dataset is annotated similarly to the Casablanca dataset, with the exception that the frontal head annotation has been reduced to faces. Casablanca is an old video with an unusual scenario for head detection, with visuals that are greyscale, have low illumination, and are crowded. Monochromic photos with a resolution of 464x640 pixels constitute the dataset. Due of the dense background, large differences in scales, positions, and appearances of human heads, the dataset is extremely complicated to handle.

3.2. Hollywood Heads dataset

The dataset is the largest in the world, with 224,740 photos drawn from 21 distinct Hollywood films. The collection contains 369,846 annotations (bounding boxes) that cover human heads of various sizes, shapes, and looks. The data is separated into three groups. The set includes 216,719 photos for training the models, which span 15 different films. The second batch contains 6,719 photos from three films for validation, while the third set has 1,302 images from the remaining three films for testing.

3.3. Scuthead dataset

Part A and Part B are the two portions of the dataset. Part A contains 2000 image samples taken from a zoom-out camera mounted in the corner of a university classroom. Part A has 67,321 human head annotations. Human heads usually have similar positions and orientations inside the classroom, thus the photographs are carefully picked to minimize the resemblance and increase the variance across the images. Part A's population density ranges from 0 to 90 people per picture, with an average count of 51.8.

3.4. Merge dataset

All other datasets contain limited size of heads i-e Casablanca contain medium size of heads, Hollywood heads contain variable but colored heads and Scuthead contain only small heads. In this research in order to improve the accuracy of the head detection all these three datasets are combined to form one dataset containing every type of head. This dataset is named as merge dataset and it is further trained on yolov5. It contains 15466 images. This dataset is divided into training, testing and validation in the ratio of 70% training, 20% Val and 10% testing. In other words, 10826 are used for training, 1546 for testing and 3093 for validation.

3.5. Results

Several experiments are performed to analyze the performance of our proposed methodology. Casablanca dataset

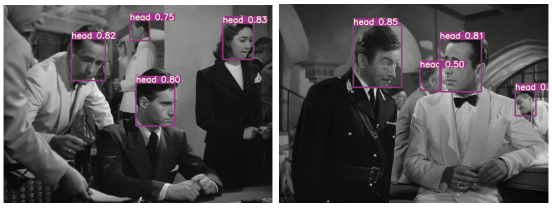

Table 3: Parameters and hyper parameters of the model.

Parameters	Casablanca	Hollywood	Scuthead	Merge dataset
Input size	650x650	650x650	650x650	850x820
Output size	Variable	Variable	384x640	Variable
Batch	10	15	10	15
Epoch	100	100	100	100
Learning rate	0.01	0.01	0.01	0.01
Confidence threshold	0.4	0.4	0.4	0.4
Weight decay	0.0005	0.0005	0.0005	0.0005
Data split (training)	70%	70%	70%	70%
Data split (Val)	20%	20%	20%	20%
Data split (Testing)	10%	10%	10%	10%

is considered to be the most commonly available and used dataset for human detection or crowd counting. As Casablanca includes the black and white images and include mostly large and medium heads. Table 3 shows the parameters used for training Casablanca, Scuthead and Hollywood dataset. For Casablanca dataset, batch size is varied from 10 to 16 and the better results are obtained on batch size 10. Epochs size was varied from 50 to 200 and the best result was obtained at size 100. Hollywood heads contains all the colored images. As the dataset is large, batch size was varied from 10 to 15 and at 16 best results were achieved. Scuthead dataset comprises of small heads only. In this research, only Dataset Part A is used to analyze the results as both the dataset contain like images. The accuracy rate for this dataset is not satisfactory as it misses the medium and large heads. Different datasets contain different size of heads. In order to create a generalized model that can detect any type of head from a crowd, we have combined the three datasets and formed another dataset named as merge dataset.

Fig. 3: Head detection results using Casablanca dataset for training yolov5 and testing the (a) Casablanca (b) Hollywood and (c) Scuthead dataset

3.6. Results when trained Casablanca Dataset on yolov5

The mAP of the Casablanca is high i.e 83%. Hollywood heads mAP is 70.9% and the Scuthead dataset contain all small heads so its mAP is very less i-e 6%. Therefore, the draw back in using this dataset was its accuracy rate detecting small heads is very less and we cannot use for a stable model. The mean average precision of out this experiment is 83% when the step size is 100 and the precision obtained is 95%.

3.7. Results when Yolov5 is trained on Hollywood Heads Dataset:

Training mAP of this dataset is very high. It contains mostly medium size heads and it lags the accuracy rate in detecting small heads. The mAP of the Scuthead is 0% as it contains all the small heads, Casablanca dataset is 31.8% as these are all black and white images while the mAP of the Hollywood Heads is 56%. We cannot consider it the most stable dataset as it is not detecting small heads. From the results, it can be seen that the detection rate is less as it is not able to detect very small and very large heads. When Scuthead dataset is tested, none of the small heads are detected. Hence, it is proved that this method is not accurate for detecting small heads. The detecting accuracy of this method is very less. It detects medium heads with greater precision but strongly ignore the small heads. The detection time is greater as compared to the Casablanca dataset training.

3.8. Results when trained Yolov5 on Scuthead Dataset:

As the Scuthead dataset comprises of only small heads, it completely ignores the medium and large size heads. The mAP of Casablanca and Hollywood Heads is very less but it detects the scuthead dataset with 66% mAP. The detection rate is very less in this as compared to the other methods. When images of Casablanca dataset are tested, only small heads are detected as this network is trained on Scuthead dataset that include only small heads that is why it is not capable of detecting other head sizes. When Hollywood Heads dataset is tested on the pre-trained network it is analyzed that it is only detecting some small heads and accuracy is very less.

3.9. Results when trained Merge dataset on yolov5

This dataset comprises of the images of all previous dataset used. Therefore, we can say it contains every kind of Head size in different scenarios from multiple angles. The mAP of this dataset is improved and more stable as compared to the other datasets. Its detection rate is also high as compared to the previous experiments. When we have trained the yolov5 on Merge dataset and tested the Casablanca dataset, it is detecting almost all the heads. So we can assume that this approach is much accurate as compared to the other experiments. The merge dataset yields the best and the most accurate results so far. The mAP of the Merge Dataset is higher than the Scuthead and Hollywood heads while less than the Casablanca itself that is when we train our model using the Merge dataset give more accurate results than Hollywood heads and Scuthead dataset. It means that our precision rate of improved and this pre-trained model head detection rate is more than our previous experiments.

4. CONCLUSION

Crowd counting offer the wide variety of applications. In the field of computer vision human head detection in overcrowded scenes is a significant challenge. One such use-case may be crowd counting based on head detection, where a deep learning-based approach tends to produce more reliable results than earlier crowd counting techniques. In past Fast RCNN was considered as the most used technique for the object detection, but it lacks in identifying the small heads in crowded region. Our research proves that it is possible to improve the accuracy rate by using yolov5. The main difference is that the features are extracted at the last layer in Fast RCNN while in yolov5 due to the focus operation feature maps are extracted in the starting layers. Crowd Counting via head detection using deep learning is a new approach in this area. In this research multiple datasets are used to evaluate the architecture performance. After obtaining the results another dataset is formulated which contain all the images of other dataset and hence result in the stabilized pertained model. Other main goal achieved by this research is the improvement in the accuracy rate as compared to the other architectures. The rate of detecting the Small heads in raised from 66% to 78%. From the results shown above we can assume that this method is providing the stable and accurate pre-trained model for detecting every kind of heads. The rate of detecting the Small heads in raised from 66% to 78%. The accuracy of detecting Hollywood heads dataset is improved from 56% to 75% and Scuthead mean average precision is increased to 66% to 78%.

5. FUTURE WORK

In future the idea is to extend the same idea to High density crowds for example in Haram there are billions of people. In such scenarios it is approximately impossible to detect every head. Therefore, we cannot achieve the accurate results. For Crowd Counting in such scenarios dot annotations are used. The idea is to develop another architecture to carry out this research and to overcome the challenges in detecting high density crowd with better accuracy. The future work involves detections combined with the tracker for enhanced performance. As it is obvious from the above results that the

detection rate of yolov5 is higher as compared to the previous detection methods therefore we can deploy this method in tracker for detection in real time scenarios.

6. REFERENCES

[1] Sami Abdulla Mohsen Saleh, Shahrel Azmin Suandi, and Haidi Ibrahim, "Recent survey on crowd density estimation and counting for visual surveillance," *Engineering Applications of Artificial Intelligence*, vol. 41, pp. 103–114, 2015.

[2] Mohib Ullah, Habib Ullah, Nicola Conci, and Francesco GB De Natale, "Crowd behavior identification," in *2016 IEEE international conference on image processing (ICIP)*. IEEE, 2016, pp. 1195–1199.

[3] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[4] Sultan Daud Khan, Habib Ullah, Mohammad Uzair, Mohib Ullah, Rehan Ullah, and Faouzi Alaya Cheikh, "Disam: Density independent and scale aware model for crowd counting and localization," in *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 4474–4478.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[6] Mohib Ullah, Ahmed Mohammed, and Faouzi Alaya Cheikh, "Pednet: A spatio-temporal deep convolutional neural network for pedestrian segmentation," *Journal of Imaging*, vol. 4, no. 9, pp. 107, 2018.

[7] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[8] Xiangjun Chen, Zhaohui Wang, Yuefu Zhan, Faouzi Alaya Cheikh, and Mohib Ullah, "Interpretable learning approaches in structural mri: 3d-resnet fused attention for autism spectrum disorder classification," in *Medical Imaging 2022: Computer-Aided Diagnosis*. SPIE, 2022, vol. 12033, pp. 611–618.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[10] Sareer Ul Amin, Mohib Ullah, Muhammad Sajjad, Faouzi Alaya Cheikh, Mohammad Hijji, Abdulrahman Hijji, and Khan Muhammad, "Eadn: An efficient deep learning model for anomaly detection in videos," *Mathematics*, vol. 10, no. 9, pp. 1555, 2022.

[11] Brian E Moore, Saad Ali, Ramin Mehran, and Mubarak Shah, "Visual crowd surveillance through a hydrodynamics lens," *Communications of the ACM*, vol. 54, no. 12, pp. 64–73, 2011.

[12] Tao Zhao, Ram Nevatia, and Bo Wu, "Segmentation and tracking of multiple humans in crowded environments," *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 7, pp. 1198–1211, 2008.

[13] Ibrahim Saygin Topkaya, Hakan Erdogan, and Fatih Porikli, "Detecting and tracking unknown number of objects with dirichlet process mixture models and markov random fields," in *International Symposium on Visual Computing*. Springer, 2013, pp. 178–188.

[14] Min Li, Zhaoxiang Zhang, Kaiqi Huang, and Tieniu Tan, "Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection," in *2008 19th international conference on pattern recognition*. IEEE, 2008, pp. 1–4.

[15] Bo Wu and Ram Nevatia, "Tracking of multiple, partially occluded humans based on static body part detection," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*. IEEE, 2006, vol. 1, pp. 951–958.

[16] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *2008 IEEE conference on computer vision and pattern recognition*. IEEE, 2008, pp. 1–7.

[17] Xinyu Chen, Mingzhe Liu, Jun Ren, and Chuan Zhao, "An overview of crowd counting on traditional and cnn-based approaches," in *2020 6th International Conference on Robotics and Artificial Intelligence*, 2020, pp. 126–133.

[18] Victor Lempitsky and Andrew Zisserman, "Learning to count objects in images," *Advances in neural information processing systems*, vol. 23, 2010.

[19] Viet-Quoc Pham, Tatsuo Kozakaya, Osamu Yamaguchi, and Ryuzo Okada, "Count forest: Co-voting uncertain number of targets using random forest for crowd density estimation," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3253–3261.

[20] Deepak Babu Sam, Skand Vishwanath Peri, Mukuntha Narayanan Sundararaman, Amogh Kamath, and R Venkatesh Babu, "Locate, size, and count: accurately resolving people in dense crowds via detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 8, pp. 2739–2751, 2020.

[21] Xin Zeng, Yunpeng Wu, Shizhe Hu, Ruobin Wang, and Yangdong Ye, "Dspnet: Deep scale purifier network for dense crowd counting," *Expert Systems with Applications*, vol. 141, pp. 112977, 2020.

[22] Min Fu, Pei Xu, Xudong Li, Qihe Liu, Mao Ye, and Ce Zhu, "Fast crowd density estimation with convolutional neural networks," *Engineering Applications of Artificial Intelligence*, vol. 43, pp. 81–88, 2015.

[23] Bolei Xu and Guoping Qiu, "Crowd density estimation based on rich features and random projection forest," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2016, pp. 1–8.