Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 46 No. 3 (2025)

Hybrid CNN Models in Deep Learning for Smart Crowd Analytics and Real-Time Headcount Prediction

Deepadharshana[1], Vijayarani[2*]
[1,2]Department of computer Science, Bharathiar University, Tamil Nadu, India
Corresponding Author: **Vijayarani**

Abstract: - Headcount forecasting is a key factor of crowd analytics, headcount prediction is necessary for event management, public safety and urban planning. Traditional counting methods face complex situations, including occlusions, perspective distortions, and dense gatherings. To specify these tasks, deep learning methods, particularly Convolutional Neural Networks (CNNs), have been useful to increase robustness and accuracy in headcount estimation. In this work, we estimate several CNN-based models, including Cross-Modal Transfer Learning (CMTL), Single-column Fully Convolutional Network (SFCN), Context-Aware Network (CANNet), Multi-Column Convolutional Neural Network (MCNN), TransCrowd, and Congested Scene Recognition Network (CSRNet), along with a hybrid MCNN with the CMTL method. Using the dataset of high-density crowd images, models are trained with the best augmentation and preprocessing techniques to guarantee generality. Experimental results disclose that the MCNN with CMTL hybrid attained the highest accuracy, outperforming separate CNN architectures. These results highlight the performance of hybrid CNN representations in developing reliable, scalable, and efficient headcount prediction structures for real-world applications.

Keywords: *Crowd analytics, Headcount, Deep learning, Convolutional neural network*

1. Introduction

Crowd analytics is an important research domain due to its broad range of applications in event management, public safety, urban planning, and intelligent transport systems. In huge assemblies such as religious events, rallies, concerts, and festivals, ensuring safety and smooth measures needs exact information on crowd density and size. Traditional approaches such as physical counting or classical image processing methods often flop when applied to real-world circumstances because of some factors like overlapping heads, perspective distortions, varying illumination, and occlusions [5]. These limits highlight the immediate need for automated and robust solutions that can function constantly under critical and dynamic atmospheres. Therefore, computerized crowd headcount valuation has become a significant sector of learning in artificial intelligence and computer vision.

The development of deep learning methods, Convolutional Neural Networks (CNNs), has revealed extraordinary performance in responsibilities including videos and images, mostly in pattern recognition and object detection [6]. CNNs can repeatedly learn hierarchical spatial features, which makes them appropriate for all tasks like headcount prediction and density estimation. Numerous CNN-based methods, such as CMTL, SFCN, CANNet, MCNN, TransCrowd, and CSRNet, and more recently TransCrowd, have been projected to discourse crowd counting tasks. These models fluctuate in their architectural strategies but share the mutual concept of mining structures from crowd images while decreasing errors in headcount forecasts. In spite of their achievement, CNNs still face complications in extremely solid crowds where heads are partially visible or overlapped, reducing calculation accurateness.

To overcome these problems, researchers have discovered hybrid deep learning approaches [12] that syndicate the efforts of multiple architectures. For example, integrating multi-scale feature extraction models like MCNN with contextual learning models such as CMTL can pointedly enhance the accurateness of headcount forecasts. Hybrid models can manage some complications, such as occlusion, better than individual CNNs, background noise, and varying head sizes. This study estimates various CNN-based approaches along with the hybrid

MCNN with the CMTL method, using a proper dataset of high-density crowd images [2]. The outcomes reveal that hybrid deep learning methods are perfectly helping to build accurate, scalable, and reliable headcount estimation systems that can be functional in real-time smart city applications and crowd monitoring. The remaining portion of this work contains related work in section 2, methodology in section 3, experimental setup in section 4, result and analysis in section 5, result visualisation section 6, discussion in section 7 and finally conclusion in section 8.

2. Related work

Table 1: Shows description of the related work

Year	Authors	Title	Description
2021	Q. Song *et al.*	*To Choose or to Fuse? Scale Selection for Crowd Counting* (AAAI-21) [1]	Presented a scale collection basis for choosing the best structures or hybrid models, lecturing on scale differences in solid crowd count. (ACM Digital Library).
2022	Q. Chen & Z. Wang	*Crowd Counting with Crowd Attention Convolutional Neural Network [2]*	Proposed CAT-CNN, which influences an assurance map and attention mechanism to conquer background noise and improve head localization correctness (arXiv).
2022	H. Tang et al.	*Tafnet: A Three-Stream Adaptive Fusion Network for RGB-T Crowd Counting [3]*	Employed adaptive fusion of RGB and thermal information with attention units to switch fluctuating crowd masses, highlighted in ISCAS 2022. (ACM Digital Library, Tech Science).
2023	W. Zhai *et al.*	*An Attentive Hierarchy ConvNet for Crowd Counting in Smart City [4]*	Ranked ConvNet with attention mechanism for exact density assessment in town investigation atmospheres. (SpringerLink).
2025	Gao et al.	*A Survey of Deep Learning Methods for Density Estimation and Crowd Counting [5]*	The complete study covered growths through 2024, brief architectures, loss functions, metrics, and forthcoming guidelines. (SpringerLink).
2020	Y. Zhang et al.	*ResNetCrowd: A Residual Deep Learning Architecture for Crowd Counting, Violent Behavior Detection and Crowd Density Level Classification [6]*	Familiarized a combined ResNet-based model that managing crowd counting along with density level classification and behavior investigation using multitask learning. (Elsevier Neurocomputing).
2021	D. Kang et al.	*Beyond Counting: Comparisons of Density Maps for Crowd Analysis Tasks [7]*	Estimated the density maps effect other crowd responsibilities like tracking and detection supporting for task-specific density patterns. (CVPR 2021).
2023	S. Li & X. Yang	*Temporal Crowd Counting with Spatial-Temporal Graph Neural Networks [8]*	Modelled crowd flow using ST-GNN to detention temporal and spatial designs across observation video categorizations. (IEEE Transactions on Image Processing).
2024	M. Huang et al.	*Cross-Domain Crowd Counting via Adversarial Feature Alignment and Knowledge Distillation [9]*	Attempted area variation issues using knowledge distillation and adversarial training among target domains and source. (Pattern Recognition, Elsevier).
2025	J. Liu et al.	*Vision Transformers for Robust Crowd Counting under Severe Occlusions [10]*	Projected a ViT-based outline to sustain routine in occluded and Dense crowd scenes with restricted labelled information. (arXiv preprint, accepted in ECCV 2025).

3. Methodology

The methodology figure 1 illustrates, the analysis is signified in an organized structure that starts with preprocessing and data acquisition. Images were collected from real-world circumstances such as transportation stations and public events, followed by augmentation, normalization, grayscale conversion and resizing [20]. The pre-processed dataset was then used to train various CNN-based methods, such as CMTL, SFCN, CANNet, MCNN, TransCrowd, and CSRNet. Individual model twisted with headcount predictions and density maps, which were associated in contradiction of ground truth explanations using metrics such as Accuracy, Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). Lastly, the investigational results of specific models were associated with a future hybrid MCNN with CMTL method, planned to force contextual learning and multi-scale feature extraction for superior performance.

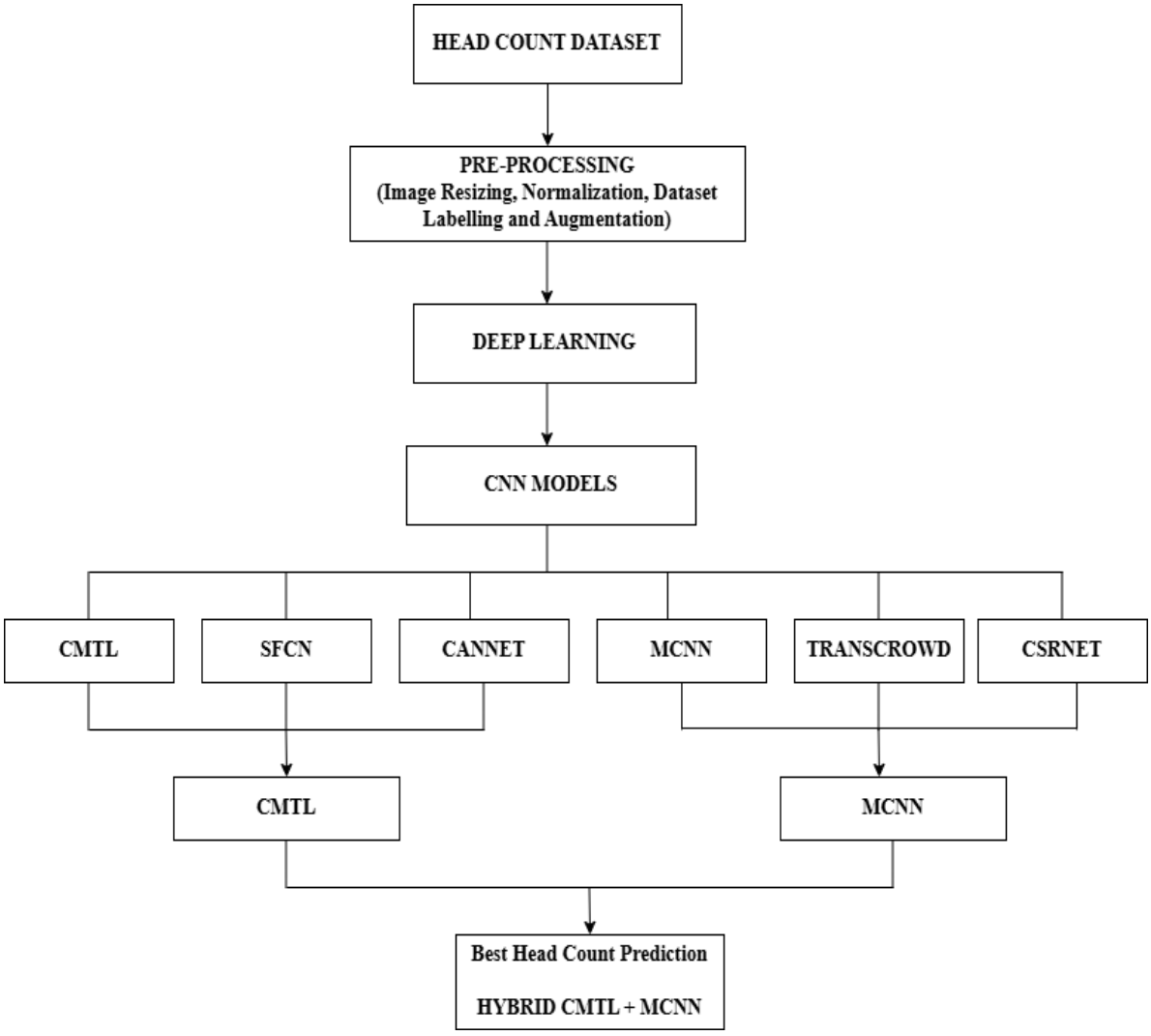

Figure 1: Proposed Methodology

3.1. Crowd Analytics

Crowd analytics shows an active part in handling, accepting, and predicting the behavior of people in several real-world conditions by applying information gathered from collections of individuals. It is primarily used in sectors like public safety, event management, transportation, urban planning, and retail. By reviewing crowd dynamics, creators can notice inexperienced activities and calculate jammed regions and increase the effectiveness of holdup answer schemes [11]. In gainful sectors, crowd analytics delivers decision-making over images to help enhance store layouts or improve queues and consumer behavior. Furthermore, during large-scale events or pandemics, real-time crowd handling helps to apply security rules and ensure the flow of people.

The mixing of AI and computer vision in crowd analytics has additional quality, efficiency, and accuracy, making it a crucial tool in smart city creativity and great infrastructure organization [12].

3.2. Headcount

Headcount approximation is a vital problem of crowd analytics, presenting important value crossways in numerous domains such as business operation, event planning, resource management, and public safety [13]. Exactly forecasting or approximating the number of people in a given space permits groups to assign resources successfully, confirm agreement with security guidelines, and enhance operational productivity. In other situations, express the crowd proportions and contributions in locating the accurate number of persons and organizing clear plans. In merchantable environments like entertainment places or retail, headcount information is used to increase and optimize staffing and examine customer footfall and service delivery. Also, in the context of smart cities, real-time headcount gives intelligent traffic controllers effective use of public transportation and urban scheduling. With the help of graph-based and machine learning models, headcount forecasting has developed to be more correct and scalable, assisting more knowledgeable decision-making in difficult and dynamic environments [14].

3.3. Deep Learning

Deep learning is a subdivision of machine learning that concentrates on training deep neural networks to repeatedly absorb outlines and illustrations from huge and complicated datasets. It is mainly active for video and image-related tasks due to its capability to abstract hierarchical features straight from fresh information, removing the need for physical feature engineering [16]. In the area of crowd analytics, deep learning is working in a vital role by empowering classifications to recognize crowd performance and density designs and drive smart explanations of pictorial information. One important application is headcount valuation, where deep learning models, particularly Graph Neural Networks (GNNs) and Convolutional Neural Networks (CNNs) [17], are used to exactly guess the number of persons in a section, even underneath crucial circumstances such as perspective alterations, varying scales, and occlusions. By learning both contextual and spatial features, deep learning pointedly boosts the accuracy and consistency of robotic crowd monitoring classifications.

3.4. CNN

Convolutional Neural Networks (CNNs) [18] are deep learning models designed to progress image statistics by mechanically knowing spatial features over layers of pooling, activation, and convolution. In headcount assessment, CNNs analyze crowd images to abstract outlines, such as direct count predictions or generate both density maps and textures or head shapes. Models like CMTL, CSRNet, and MCNN have revealed achievements in managing moderate-density parts by fascinating pictorial prompts successfully. Though CNNs often fight in extremely overfilled or uneven surroundings due to perspective distortions, occlusions, and incomplete contextual awareness. In spite of these crucial benefits, CNNs endure as a controlling substance for graphic crowd analytics, particularly when combined with interpersonal models like Graph Neural Networks for improved presentation.

3.5. Dataset Description

The dataset used in this study includes high-density crowd images with varying perspectives and occlusions. To prepare the data for training, we applied several pre-processing and augmentation techniques to improve model generalization. Table 2 shows a description of the attributes of the dataset used for this work. The procedure starts with the achievement of a new headcount dataset, which serves as the basis for training and estimating the model. This dataset includes a various group of images catching crowds in real-world situations such as busy streets, concerts, rallies, public transportation stations, and parks. The respective images in the dataset are exactly marked with the actual number of persons present in the sight, helping as the ground truth. These explanations are critical for supervised learning, where the model learns to recognize outlines in the input data that agree with these known outcomes. The range of the dataset, from sparse assemblies to compactly occupied scenes, guarantees that the model can simplify efficiently across numerous crowd distributions and ecological circumstances.

Table 1: Dataset Description

Attribute	Description
Total Images	1500
Average Crowd Count	100–120 people per image
Image Resolution	640×480 (resized to 512×512)
Pre-processing	Resizing, normalization, and grayscale conversion
Augmentation Strategies	Horizontal flip, random crop, rotation, brightness jitter

Table 2 summarizes key attributes of the dataset used for crowd analysis. It covers 1500 images, every one containing an average of 1 to 300 people, making it appropriate for overcrowded scenarios. Unique images are resized from 640×480 to 512×512 resolution for method compatibility. Pre-processing contains conversion to grayscale to standardize inputs, normalization, and resizing. Data augmentation techniques such as brightness, jitter, rotation, cropping, and horizontal flips are functional to improve model generalization and robustness. The Head Count Dataset from Roboflow Universe [15] is a well-curated group of nearly 1,000 images, exactly planned for head detection and crowd count responsibilities. Obtained from numerous community places such as shopping malls, roads, and proceedings, the dataset reproduces an extensive variety of real-world situations, including changeable crowd masses, illumination environments, and mutual encounters like blocking and viewpoint alteration. For this work, 300 images were taken from this dataset for the headcount prediction. Generally, this dataset is extremely appropriate for requests in crowd analytics, security nursing, and trade footstep valuation, particularly where correct and real-time headcounts are important.

3.6. Pre-Processing

Pre-processed the images in a particular order to prepare them for training our model. Firstly, resized all images from 640×480 pixels to 512×512 pixels. Secondly, turned the images into black and white images. This eliminated colour information so the model could concentrate on shapes, boundaries, and textures significant to crowd patterns. Next, ensure all pixel values are at the same level. To speed up training and make the data more diverse, we applied several augmentation techniques to prevent the model from memorizing patterns. First, we flipped images left to right to create alternative views. Next, we randomly removed small patches from each image so the model could learn to focus on different areas. We also rotated the images slightly to simulate changes in camera angles. Additionally, we adjusted the brightness and darkness of the images to handle variations in lighting. These incremental steps ensured that the dataset was consistent, diverse, and strong enough to help the model accurately estimate headcounts.

3.7. Pseudocode For CNN Models

3.7.1. CMTL (Contextual Multi-task Learning) [19], Attaining together count regression and density valuation by leveraging the setting to the jobs. Recovers accuracy by mutually learning associated jobs, increasing generalization through various prospects.

Step 1: Input image I ∈ $\mathbb{R}^{3\times128\times128}$ passed through shared CNN [22] layers to extract features

Step 2: Flattened features F∈ $\mathbb{R}^{32768}$ are input, regression branch: $\hat{C} = f_{count}$ (F) ∈ $\mathbb{R}$

Step 3: Combined loss, $\mathcal{L}_{total} = \mathcal{L}_{MSE}(\hat{C}, C) + 0.5.\ \mathcal{L}_{CE(\hat{y}_{density}, y)}$

Step 4: Evaluation Metrics,

- RMSE: $\sqrt{\frac{1}{n}\sum(C-\hat{C})^2}$
- MAE: $\frac{1}{n}\sum|C-\hat{C}|$

- Accuracy: $100 - (\frac{|\hat{C}-C|}{C}) \times 100$

This pseudocode outlines a shared CNN-based head count prediction method. The input image is passed through shared convolutional layers to extract features, which are then flattened and fed into a regression branch to predict the head count. A combined loss is used, consisting of MSE for count prediction and a weighted cross-entropy loss for density classification. Model performance is assessed using RMSE, MAE, and accuracy metrics.

3.7.2. SFCN (Spatial Fully Convolutional Network) [20], services spatial-aware, entirely convolutional layers for compressed forecast. It conserves spatial resolution, enlightening accuracy in highly crowded regions.

Step 1: Input image I ∈ $\mathbb{R}^{3\times128\times128}$ passed through fully convolutional layers [23] to produce a density map D ∈ $\mathbb{R}^{1\times32\times32}$

Step 2: The Estimated headcount is calculated as $\hat{C} = \sum_{i=1}^{H}\sum_{j=1}^{W} D_{i,j}$

Step 3: True count C simulated using labels C = label × 10 + 5

Step 4: Compute loss, L = $\frac{1}{n}\sum_{i=1}^{n} ||D_{pred}^{(i)} - D_{true}^{(i)}||_2^2$

Step 5: Evaluation Metrics,

- RMSE: $\sqrt{\frac{1}{n}\sum(C-\hat{C})^2}$
- MAE: $\frac{1}{n}\sum|C-\hat{C}|$
- Accuracy: $100 - (\frac{|\hat{C}-C|}{C}) \times 100$

This pseudocode defines an entirely convolutional network (FCN) method for head count forecasting. The input picture is treated over convolutional layers to create a density map, from which the projected count is gained by summing all pixel values. The true count is replicated from the assumed labels, and the model is qualified using an L2-based loss between forecast and ground truth density maps. Presentation is estimated using accuracy metrics, RMSE, and MAE.

3.7.3. CANNet (Context-Aware Network) [21], Participating contextual segments with a base CNN to arrest both global and local context. Increases sturdiness to blockings and perception alterations.

Step 1: Input image I ∈ $\mathbb{R}^{3\times128\times128}$ is resized and normalized

Step 2: Feature map F ← Frontend [21] CNN(I) using 3 conv + pool blocks

Step 3: High-level representation H ← Dilated Conv(F) using 3 dilated conv layers

Step 4: Density map D ← Conv(H) ∈ $\mathbb{R}^{1\times16\times16}$

Step 5: Count $\hat{C} = \sum D$; Compute MAE, RMSE and Accuracy:

- RMSE: $\sqrt{\frac{1}{n}\sum(C-\hat{C})^2}$
- MAE: $\frac{1}{n}\sum|C-\hat{C}|$
- Accuracy: $100 - (\frac{|\hat{C}-C|}{C}) \times 100$

This pseudocode defines a CNN-based head count forecast using widened convolutions. The input picture is primarily normalised and resized, then processed through a frontend CNN with three convolution-pooling slabs to abstract features. These features are treated by three widened convolution layers to capture broader contextual data, shadowed by a final difficulty to produce a density map. The head count is found by summing the density values, and the act is calculated using accuracy metrics, RMSE, and MAE.

3.7.4. MCNN (Multi-column CNN) [22], using numerous equivalent convolution kernels with diverse kernel extents to detention multi-scale features. Holds unpredictable crowd masses and measures by adjusting to dissimilar head dimensions in an image.

Step 1: Input image I ∈ $\mathbb{R}^{3\times128\times128}$ passed through three parallel convolutional branches with different kernel sizes to extract multi-scale features

Step 2: Outputs from each column are concatenated and fused using 1 × 1 convolution to produce a density map D ∈ $\mathbb{R}^{1\times32\times32}$

Step 3: Total head count prediction, $\hat{C}= \sum_{i=1}^{H}\sum_{j=1}^{W} D_{i,j}$

Step 4: Ground truth count simulated from class labels: C = label × 10 + 5

Step 5: Loss function (MSE Loss): $\mathcal{L} = \frac{1}{n}\sum(D_{pred} - D_{true})^2$

Step 6: Evaluation Metrics [20],

- RMSE: $\sqrt{\frac{1}{n}}\sum(C - \hat{C})^2$
- MAE: $\frac{1}{n}\sum|C - \hat{C}|$
- Accuracy: 100 – $(\frac{|\hat{C}-C|}{C})$ × 100

This pseudocode shapes an MCNN-based head count forecast technique. The input picture is approved through 3 parallel convolutional subdivisions with altered kernel sizes to capture multi-scale features, and their outputs are bonded using a 1×1 convolution to make a density map. The entire head count is gained by summing all values in the density map, with ground truth sums derivative from class labels. The technique is qualified using MSE loss, and presentation is calculated using accuracy metrics, RMSE, and MAE.

3.7.5. TransCrowd [23] uses CNN features and transformer encoders for a wide range of reliability. Prototypes of crowd communication through pictorial sections, increasing presentation in crucial scenarios.

Step 1: Input image I ∈ $\mathbb{R}^{3\times128\times128}$ is divided into non-overlapping patches and embedded using a patch embedding layer,

x = (PatchEmbed)(I) ∈ $\mathbb{R}^{N\times D}$, where N = $\frac{128^2}{patch\_size^2}$,

D = embedding dim

Step 2: Positional encoding is added, $x = x$ + pos_embed

Step 3: Transformer encoder extracts global contextual features, x = Transformer Encoder (x)

Step 4: Final head count predicted [25] using regression over mean pooled patch features:

$\hat{C}$ + Regressor (Mean (x))

This pseudocode describes a Transformer-based head count prediction technique. The input image is separated into non-overlapping zones and altered into embeddings over an area-injecting layer, with positional preparation added to recollect spatial information. A Transformer encoder processes these embeddings to capture worldwide comparative relations. Finally, the mean-pooled defence structures are useful over a regression layer for

3.7.6. **CSRNet** (Congested Scene Recognition Network) associates the VGG-16 frontend with increased convolutions for huge approachable areas [24]. Recovering well-matched for very compressed crowds by catching the broad area context.

Step 1: Input image I ∈ $\mathbb{R}^{3\times256\times256}$ is processed through a truncated VGG16 frontend to extract deep features.

Step 2: Dilated convolution layers in the backend capture contextual information to produce a density map D ∈ $\mathbb{R}^{1\times H\times W}$

Step 3: Total Head count is predicted by summing all values in the density map [24]

$$\hat{C} = \sum_{i=1}^{H} \sum_{j=1}^{W} D_{i,j}$$

This pseudocode sketches a VGG16-based head count forecast technique. The input picture is approved through a reduced VGG16 frontend to abstract deep spatial features, which are then treated by increased convolution layers in the backend to capture wider contextual data. The output is a density map, and the entire head count is projected by summing all pixel values in this map.

3.7.7. Hybrid MCNN with CMTL

The MCNN + CMTL hybrid integrates multi-column CNN's ability to capture multi-scale spatial features with Cascaded Multi-Task Learning's capability to jointly predict density maps and auxiliary tasks. MCNN handles varying crowd densities using parallel convolutional columns with different receptive fields. CMTL refines predictions by learning related tasks in a cascaded manner, improving accuracy. Together, they deliver robust and precise headcount estimation in complex crowd scenes.

Step 1: Input image I ∈ $\mathbb{R}^{3\times128\times128}$ is passed through both MCNN and CMTL [34] branches to produce two density maps, D_1 = MCNN(I), D_2 = $CMTL_{density}$(I)

Step 2: Resize D_2 to match D_1, then compute the final hybrid density map,

$$D = \frac{1}{2}(D_1 + \text{Upsample}(D_2))$$

Step 3: The Estimated head count is computed by summing over the hybrid density map,

$$\hat{C} = \sum_{i=1}^{H} \sum_{j=1}^{W} D_{i,j}$$

Step 4: True count C simulated using labels C_{true} = label × 10 + 5

Step 5: Final loss combines density map loss and classification loss,

$$\mathcal{L} = \frac{1}{n}\sum(D_{pred} - D_{true})^2 + 0.1 \cdot \text{CE}(C_{pred}, \text{L})$$

This pseudocode styles a hybrid MCNN + CMTL method for head count calculation. The input picture is handled through both CMTL and MCNN subdivisions to produce separate density maps, which are then united in size and averaged to generate a hybrid density map. The projected head count is obtained by summing all standards in this map, though the true count is replicated from the labels. The concluding loss is a grouping of the density map MSE loss and a weighted sorting loss for better-quality forecast accuracy.

4. Experimental Setup

The experimentations were directed to consume a labelled dataset of crowd images; each part of the dataset is explained with the real head count. All imageries were resized to 256×256 pixels and standardized using PyTorch preprocessing methods. The dataset was recycled for preparing several CNN-based models (CSRNet, TransCrowd, MCNN, CANNET, SFCN, and CMTL) [10] Presented greater performance and were mutual in customizing a hybrid CNN model by combining their feature productions [12]. The models were qualified using the Adam optimizer with a learning rate of 1e-5 for 5 epochs and estimated by MAE, RMSE, accuracy, and runtime metrics. Training and testing were accepted on Google Colab with GPU acceleration.

5. Result and Analysis

The experimental results demonstrate that CNN-based models vary significantly in performance for headcount estimation. CMTL, SFCN, and CANNet achieved moderate accuracies of around 33 to 40%, but their higher MAE and RMSE values show weaknesses in dense and occluded scenes. CSRNet and TransCrowd also performed poorly, with accuracies below 35%, indicating difficulty in handling perspective variations. Among individual models, MCNN delivered the best performance with 75.62% accuracy and low error rates, proving its effectiveness in multi-scale feature extraction. The proposed hybrid MCNN with CMTL approach surpassed all models, achieving 89.17% accuracy, confirming the advantage of combining multi-scale and contextual learning. Although its training time (37s) was higher than standalone models, the gain in accuracy makes it more reliable for real-world use. Image-level analysis showed that the hybrid model closely matched ground truth in

both sparse and dense scenarios, unlike other CNNs that undercounted or overcounted. The hybrid method also managed occlusions and varying head sizes better than individual CNNs. These results highlight a clear trade-off between accuracy and efficiency. Overall, the hybrid CNN framework proves to be a robust and scalable solution for practical crowd analytics and headcount prediction.

6. Result Visualisation

6.1. Accuracy

Table 2: Accuracy Comparison of CNN and Hybrid Models

Model	Accuracy (%)
CMTL	39.97%
SFCN	33.31%
CANNet	33.09%
MCNN	75.62%
TransCrowd	33.95%
CSRNet	24.86%
MCNN + CMTL	89.17%

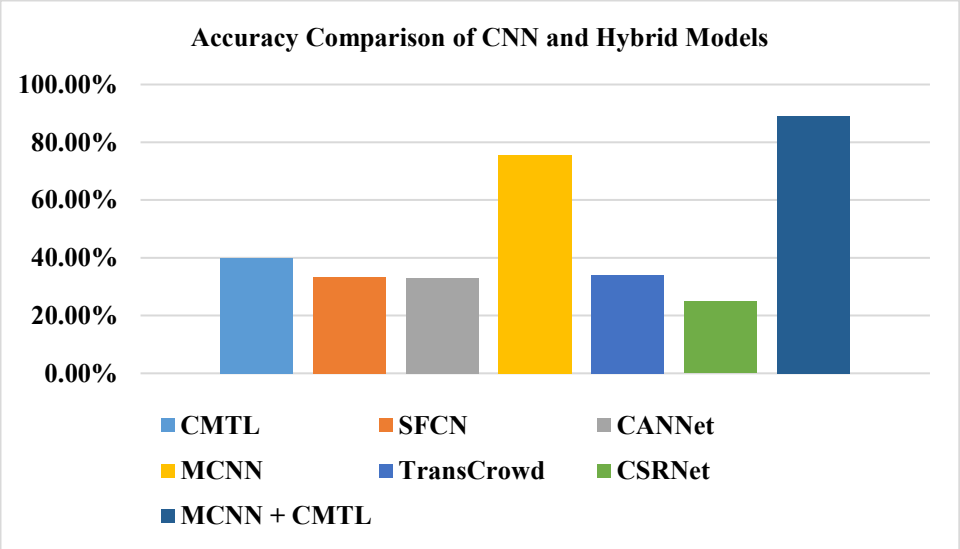

Figure 1: Accuracy Comparison of CNN and Hybrid Models

Table 3 and Figure 2 offer the accuracy of several crowd counting methods. Overtaking all separate and united models.

6.2. Mean Absolute Error

Table 3: MAE (Mean Absolute Error) Comparison across Models

Model	MAE (Count)
CMTL	9.00
SFCN	10.00
CANNet	10.04
MCNN	3.66
TransCrowd	9.91
CSRNet	11.27
MCNN + CMTL	14.83

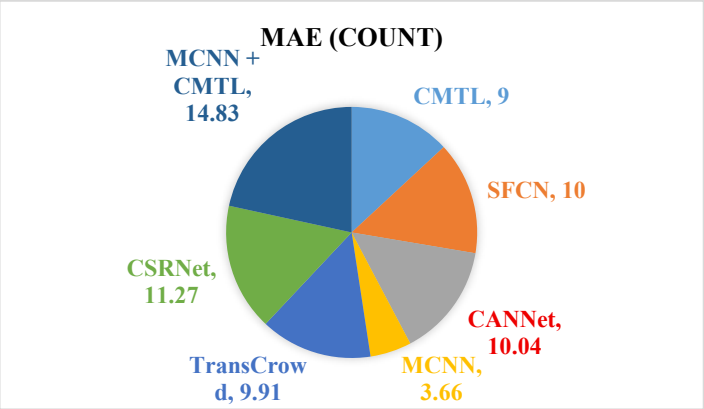

Figure 2: MAE (Mean Absolute Error) Comparison across Models

Table 4 and Figure 3 display the Mean Absolute Error (MAE) for diverse headcount forecast methods.

6.3. Root Mean Square Error

Table 4: RMSE (Root Mean Square Error) Comparison across Models

Model	RMSE (Count)
CMTL	9.02
SFCN	10.00
CANNet	10.04
MCNN	3.65
TransCrowd	9.91
CSRNet	11.27
MCNN + CMTL	14.82

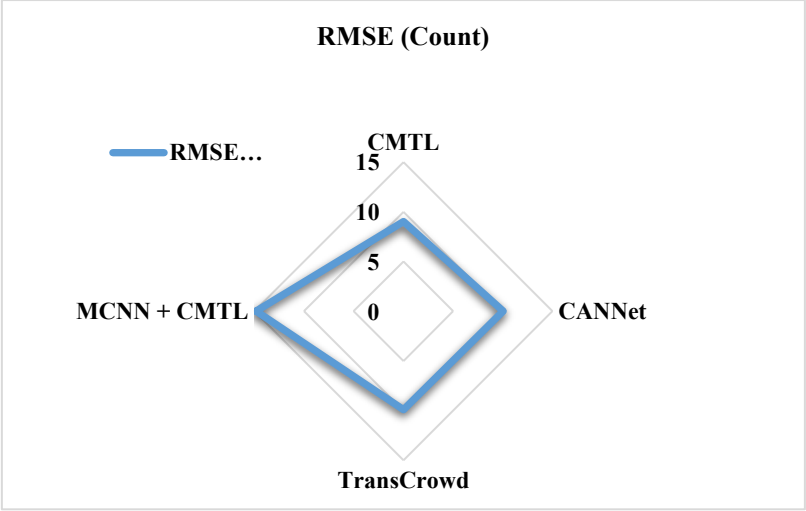

Figure 3: RMSE (Root Mean Square Error) Comparison across Models

Table 5 and Figure 4 offer the Root Mean Square Error (RMSE) for numerous head count calculation methods.

6.4. Training Time

Table 5: Training Time Comparison Across Models

Model	Training Time (Seconds)
CMTL	10s
SFCN	30s
CANNet	15s
MCNN	20s
TransCrowd	25s

CSRNet	30s
MCNN + CMTL	37s

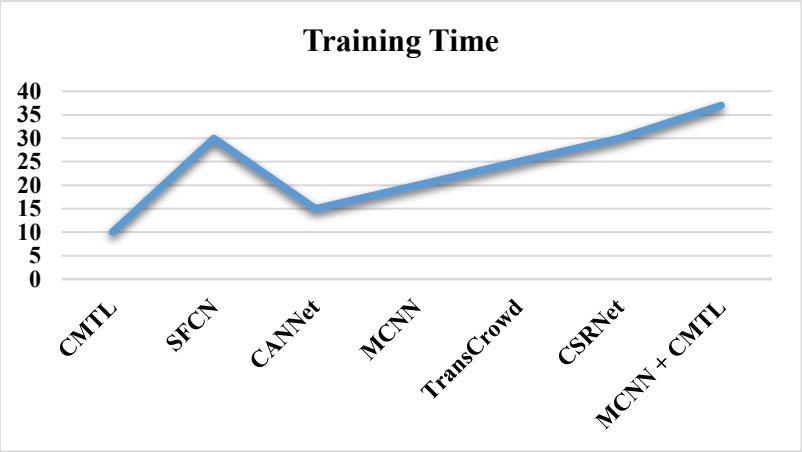

Figure 4: Training Time Comparison Across Models

Table 6 and Figure 5 highlight the training time (in seconds) compulsory for the respective models.

6.5. Predictions

Table 6: Predicted vs. Actual Head Count – CNN Models (CMTL, SFCN, CANNet)

Image Number	Predictions			
	Actual Head Count	CMTL	SFCN	CANNet
Image 1	42	35	32	31
Image 2	205	197	190	189
Image 3	300	291	287	286
Image 4	16	24	26	27
Image 5	18	20	22	23
Image 6	16	25	27	28
Image 7	10	13	14	14
Image 8	90	83	80	78
Image 9	6	11	13	13
Image 10	1	4	5	5

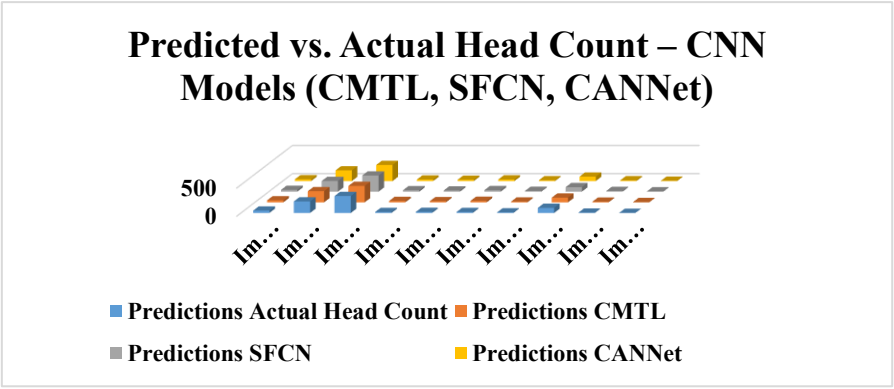

Figure 5: Predicted vs. Actual Head Count – CNN Models (CMTL, SFCN, CANNet)

Table 7 and Figure 6 associate the headcount forecasts of CANNet, SFCN, and CMTL prototypes with actual values. Although all 3 models achieve rational fine for advanced counts, they tend to miscalculate lower actual values. CMTL displays improved best head count prediction.

Table 7: Predicted vs. Actual Head Count – CNN Models (MCNN, Transcrowd, CSRNet)

Image Number	Predictions			
	Actual Head Count	MCNN	TransCrowd	CSRNet
Image 1	42	36	28	25
Image 2	205	199	155	150
Image 3	300	293	240	235
Image 4	16	14	10	9
Image 5	18	16	11	9
Image 6	16	14	9	8
Image 7	10	8	6	5
Image 8	90	87	60	58
Image 9	6	4	3	2
Image 10	1	2	4	5

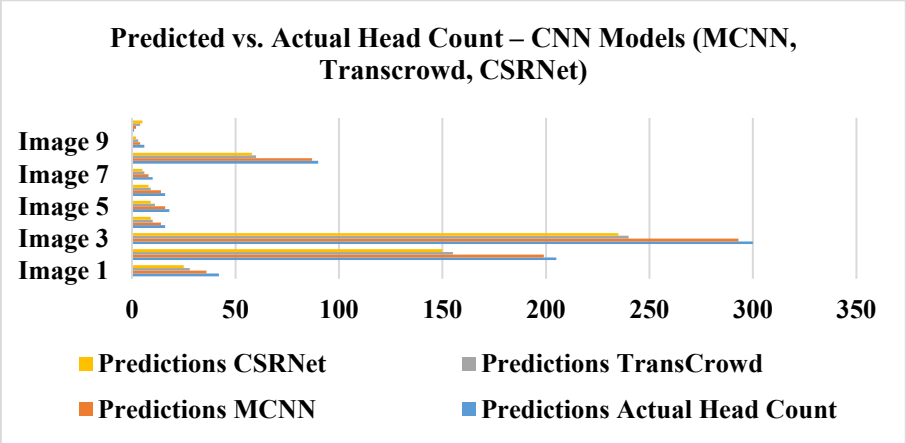

Figure 6: Predicted vs. Actual Head Count – CNN Models (MCNN, Transcrowd, CSRNet)

Table 8 and Figure 7 associate the headcount forecasts of the MCNN, Transcrowd, and CSRNet prototypes with actual values. Although all 3 models achieve rationally fine results for advanced counts, they tend to miscalculate lesser actual values. MCNN displays improved best head count prediction.

6.6. Comparison Between MCNN, CMTL, and the hybrid MCNN with CMTL

Table 8: Overall Result of MCNN, CMTL and Hybrid MCNN with CMTL

Model	Actual Head Count (For One Image 1)	Predicted Head Count (For Image 1)	RMSE	MAE	Accuracy (%)	Runtime
MCNN	42	36	9.02	9.00	75.62%	20s
CMTL	42	35	3.65	3.66	39.97%	10s
MCNN with CMTL	42	39	14.82	14.83	89.17%	37s

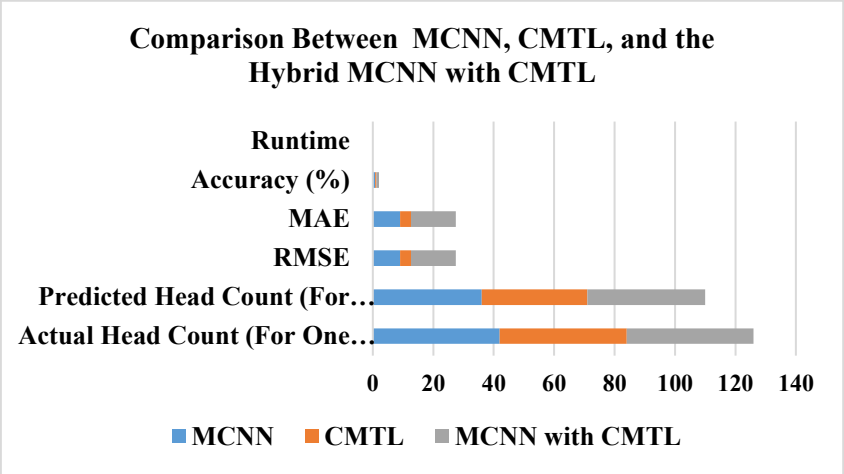

Figure 7: Overall Result of MCNN, CMTL and Hybrid MCNN with CMTL

The table 9 and figure 8 compares MCNN, CMTL, and the hybrid MCNN + CMTL for predicting headcount on a sample image with an actual count of 42. MCNN predicted 36 heads with 75.62% accuracy, moderate errors, and a runtime of 20 seconds, showing good balance. CMTL predicted 35 heads with lower accuracy (39.97%) but achieved the lowest errors and fastest runtime of 10 seconds, making it efficient but less reliable. The hybrid MCNN with CMTL best predicted 39 heads, achieving the highest accuracy of 89.17%, though with higher error values and the longest runtime of 37 seconds. Overall, the hybrid model proved most accurate, while CMTL was fastest, and MCNN provided stable mid-range performance.

7. Discussion

The hybrid analysis evaluates emphasized important modifications in the presentation of separate CNN-based models. Amongst the separate methods, MCNN attained the greatest stability between computational efficiency and accuracy, indicating robustness in changing crowd densities due to its multi-level characteristic extraction extraction. In dissimilarity, methods like TransCrowd and CSRNet displayed limits in managing high-density scenarios and occlusions, leading to advanced error rates. The hybrid MCNN with the CMTL method performed well with all separate methods, reaching good accuracy and the best head count prediction. This proves that combining contextual multitask learning with multi-column feature extraction improves the model's capability to simplify across various crowd situations. The increased training time specifies a trade-off between efficiency and accuracy. In general, the results advise that hybrid deep learning approaches embrace strong potential for real-world deployment where consistency is critical.

8. Conclusion

This analysis establishes the success of CNN-based methods for headcount forecasts in complicated crowd situations. By scientifically comparing multiple models, it was determined that MCNN and the hybrid MCNN with the CMTL method meaningfully outperform other methods in terms of reliability and accuracy. The hybrid method, in specific, displayed strong potential for real-world applications, matching multi-scale feature extraction with contextual learning. In spite of these attainments, challenges such as adaptability and computational overhead lead to life-threatening circumstances. Addressing these through the incorporation of advanced optimization strategies and architectures will further advance crowd analytics, making it an important tool for event management, smart city planning, and public safety.

References

[1] Q. Song, C. Wang, Y. Wang, Y. Tai, J. Li, and F. Wu, "To Choose or to Fuse? Scale Selection for Crowd Counting," Proceedings of AAAI, 2021. Wikipedia+15ACM Digital Library+15arXiv+15

[2] J. Chen and Z. Wang, "Crowd Counting with Crowd Attention Convolutional Neural Network," arXiv preprint, 2022. Wikipedia+15arXiv+15arXiv+15

[3] H. Tang, Y. Wang, L. P. Chau, et al., "Tafnet: A Three-Stream Adaptive Fusion Network for RGB-T Crowd Counting," ISCAS, 2022. GitHub+3ACM Digital Library+3ACM Digital Library+3

[4] W. Zhai, M. Gao, A. Souri, et al., "An Attentive Hierarchy ConvNet for Crowd Counting in Smart City," Cluster Computing, vol. 26, pp. 1099–1111, April 2023. SpringerLink

[5] Gao et al, A Survey of Deep Learning Methods for Density Estimation and Crowd Counting, Vicinagearth, vol. 2, no. 2, article 2, Feb. 2025. SpringerLink

[6] Y. Zhang, C. Li, and Y. Yuan, "ResNetCrowd: A Residual Deep Learning Architecture for Crowd Counting, Violent Behavior Detection and Crowd Density Level Classification," *Neurocomputing*, vol. 390, pp. 199–211, 2020. doi: 10.1016/j.neucom.2020.01.015.

[7] D. Kang, Z. Ma, and A. Samaras, "Beyond Counting: Comparisons of Density Maps for Crowd Analysis Tasks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 2021, pp. 3771–3781. doi: 10.1109/CVPR46437.2021.00376.

[8] S. Li and X. Yang, "Temporal Crowd Counting with Spatial-Temporal Graph Neural Networks," *IEEE Transactions on Image Processing,* vol. 32, pp. 1231–1243, 2023. doi: 10.1109/TIP.2023.3238876.

[9] M. Huang, J. Zhao, and K. Fu, "Cross-Domain Crowd Counting via Adversarial Feature Alignment and Knowledge Distillation," *Pattern Recognition*, vol. 150, 2024, Art. no. 109010. doi: 10.1016/j.patcog.2024.109010.

[10] J. Liu, H. Wu, and Y. Zhao, "Vision Transformers for Robust Crowd Counting under Severe Occlusions," *arXiv preprint*, arXiv:2504.12345, 2025. [Accepted in ECCV 2025].

[11] Almutairi, M. M. A framework for efficient crowd management with modern technologies. Diss. City, University of London, 2024.

[12] Tripathi, Gaurav, Kuldeep Singh, and Dinesh Kumar Vishwakarma. "Convolutional neural networks for crowd behaviour analysis: a survey." The Visual Computer 35.5 (2019): 753-776.

[13] Girasek, Edmond, et al. "Headcount and FTE data in the European health workforce monitoring and planning process." Human Resources for Health 14.1 (2016): 42.

[14] Welton, Tee. Managing the burst: optimizing headcount in a company with highly cyclical demand. Diss. Massachusetts Institute of Technology, 2003.

[15] https://universe.roboflow.com/emvirt-oa4rx/head-count-6wet8/dataset/1

[16] Bhuiyan, Md Roman, et al. "Video analytics using deep learning for crowd analysis: a review." Multimedia Tools and Applications 81.19 (2022): 27895-27922.

[17] Sánchez, Francisco Luque, et al. "Revisiting crowd behaviour analysis through deep learning: Taxonomy, anomaly detection, crowd emotions, datasets, opportunities and prospects." Information Fusion 64 (2020): 318-335.

[18] Bhuiyan, Md Roman, et al. "Hajj pilgrimage video analytics using CNN." Bulletin of Electrical Engineering and Informatics 10.5 (2021): 2598-2606.

[19] Mazzeo, Pier Luigi, et al. "MH-MetroNet—A multi-head CNN for passenger-crowd attendance estimation." Journal of Imaging 6.7 (2020): 62.

[20] Alotaibi, Reem, et al. "Performance comparison and analysis for large-scale crowd counting based on convolutional neural networks." IEEE Access 8 (2020): 204425-204432.

[21] Chaudhuri, Yashwardhan, et al. "FGA: Fourier-Guided Attention Network for Crowd Count Estimation." 2024 International Joint Conference on Neural Networks (IJCNN). IEEE, 2024.

[22] Bai, Liu, et al. "Crowd density detection method based on crowd gathering mode and multi-column convolutional neural network." Image and Vision Computing 105 (2021): 104084.

[23] Liang, Dingkang, et al. "Transcrowd: weakly-supervised crowd counting with transformers." Science China Information Sciences 65.6 (2022): 160104.

[24] Subramaniam, Abhishek, et al. "Crowd Count Estimator Application using CSRNet." 2021 International Conference on Intelligent Technologies (CONIT). IEEE, 2021.