International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

Deep Learning Algorithm using CSRNet and Unet for Enhanced Behavioral Crowd Counting in Video

B.Ganga [1] , Lata B T [2] , Rajshekar [3], Venugopal K R[4] Fellow, IEEE
[1]*Research Scholar, Department of CSE ,University of Visvesvaraya College of Engineering, India*
[2]*Associate Professor, Department of CSE ,University of Visvesvaraya College of Engineering, India*
[3]*Department of CSE ,University of Visvesvaraya College of Engineering, India*
[4]Former VC, Bangalore University, India
Correspondence: E-mail: gangamohanraj2003@gmail.com

Received 02Apr.2024, Revised ## Mon. 20##, Accepted ## Mon. 20##, Published ## Mon. 20##

Abstract: In crowd analysis, video data incurs challenges due to occlusion, crowd densities, and dynamic environmental conditions. To address these challenges and to enhance the accuracy we have proposed Behavioral Crowd Counting (BCC) that combines the Congested Scene Recognition Network (CSRNet) with Unet in video data. The CSRNet combines two networks namely a (1) frontend for feature extraction and (2) backend for the generation of a density map. It effectively tallies individuals within densely populated regions, offering a solution to the high crowd densities constraints. The Unet builds the semantic map and refines the semantic and density map of CSRNet. The Unet unravels complex patterns and connections among individuals in crowded settings, capturing spatial dependencies within densely populated scenes. It also offers the flexibility to incorporate attention maps as optional inputs to differentiate crowd regions from the background. We have also developed new video datasets namely Behavioral Video Dataset from the image dataset of the fine-grain crowd-counting to evaluate the BCC model. Datasets include standing vs sitting, waiting vs non-waiting, towards vs away, and violent vs non-violent videos, offering insights into posture, activity, directional movement, and aggression in various environments. The empirical findings illustrate that our approach is more efficient than others in behavioral crowd counting within video datasets, consisting of congested scenes as indicated by metrics MSE, MAE, and CMAE.

Keywords: Congested Scene Recognition Network (csrnet), Unet, Feature Extraction, Behaviour, and Crowd Analysis

1. Introduction

Crowd counting is an assessment of the number of individuals in a designated area through image or video, while crowd behavior focuses on studying and interpreting the actions, movements, and interactions of individuals within a crowd. The study of crowd behavior has evolved significantly due to the demand for detailed crowd analysis in fields such as Retail Analysis, Law Enforcement, DineSpace Analytics, Pedestrian Flow Monitoring, Traffic surveillance, and Public Safety [1]. Vibha et al., [2] explored methods for eliminating background elements to recognize moving objects in videos featuring a static background. In [3], Vibha et al., have developed a background registration method for detecting moving vehicles. Traditionally, crowd-counting algorithms have primarily focused on quantifying the number of individuals within an image. However, this *count-only* approach is inadequate in providing deeper insights into crowd dynamics and behaviors, which are critical for various practical applications, hence, there is a growing research interest in the detailed analysis of crowd videos.

Analyzing crowds using video technology is increasingly important, given the vast amount of crowd-related information available in video form. Traditional methods are inadequate for comprehensive understanding and interpreting the data in videos. Therefore, it is essential to focus on the immense potential of video data for in-depth crowd analysis. It goes beyond merely quantifying crowd counts in images and categorizing crowds in videos based on the action.

In this work, the focus is primarily on analyzing crowd behavior in applications such as Retail, Law Enforcement, DineSpace Analytics, and Pedestrian Flow Monitoring. The transition from image to video analysis has become crucial in these applications, reflecting the current trend where the majority of crowd data is now captured through video. Traditional retail analysis relies solely on static head counts,

E-mail: gangamohanraj2003@gmail.com

whereas video-based behavioral insights offer a dynamic perspective providing a deeper understanding of specific sub-categories, such as individuals in queues or leisurely browsing. Additionally, in law enforcement, challenges faced in crowd management are resolved in crowd behavioral video analysis by distinguishing violent and non-violent individuals within a crowd. Similarly, in restaurant or cafeteria settings, video analysis becomes an indispensable tool for distinguishing between standing and sitting people, enhancing the depth and precision of DineSpace analysis. In Pedestrian Flow Monitoring, our emphasis lies in leveraging video analysis to enhance crowd control strategies by moving beyond traditional static observations empowering to distinguish and manage the flow of pedestrians with greater depth and precision, addressing critical challenges in crowd dynamics.

1.1 Challenges/ Motivation

Enhanced Crowd Analysis: The increasing demand for in-depth crowd analysis in video data is driven by the growing need across various real-world contexts, such as retail, surveillance, and public safety. Traditional approaches to crowd counting have limitations when it comes to offering thorough insights into crowd behaviors, which are vital for addressing practical challenges in these domains. The demand serves as a driving force to develop more sophisticated crowd-analysis techniques that extend beyond the scope of basic crowd-counting methods to incorporate behavioral crowd-counting.

1.2 Contributions

(i) Development of BCC Architecture: The introduction of the Behavioral Crowd Counting (BCC) architecture to integrate the Congested Scene Recognition Network (CSRNet) with the Unet.

(ii) Adaptability to Congested Scenes: The CSRNet effectively counts individuals in densely populated areas which addresses the challenge of adapting to dense crowd densities.

(iii) Efficient Spatial Dependency Capture: The Unet deciphers intricate patterns and relationships among individuals in crowded environments to capture spatial dependencies within crowded scenes.

1.3 Organizational Structure

The rest of the paper is as follows. In Section 2, insights into existing research are provided in the Related Work. In Section 3, Background, Problem Definition, Objectives, and details of the BCC Architecture are discussed. In Section 4, Results and Discussions are presented. Section 5 contains Conclusion and Future Enhancements.

2. RELATED WORK

Related work on Crowd Behavior Analysis Models is presented in Table 1. Cem *et al.,* [4] have distinguished normal and abnormal crowd behaviors in surveillance videos by utilizing Motion Information Images (MIIs) derived from optical flow data. The merit is that the optical flow data improves the accuracy of identifying panic and escape behaviors. The demerit is that the real-time application is not explored in this work due to the resource-demanding nature of MII generation. Guo *et al.,* [5] have introduced a crowd anomaly detection method for video service robots, combining mean shift and *k*-means to identify abnormal behavior in crowded scenarios. The merit is that the method classifies categories with similar motion patterns and improves anomaly detection accuracy. The demerit is that the computational parameters for domain and spatial bandwidth require precise tuning.

Junyu *et al.,* [6] have developed Multi-level Feature-aware Adaptation (MFA) and Structured Density map Alignment (SDA) to address challenges in supervised learning for crowd counting and pixel-wise density estimation. The advantages are that it overcomes data scarcity issues and outperforms existing methods in cross-domain crowd counting. The challenge arises in distinguishing between background and foreground areas with similar textures, leading to inaccuracies in the estimated crowd count.

Hyojun *et al.,* [7] have addressed wildlife monitoring issues by introducing automated multi-class object counting, for endangered animal species. Fine-grained multi-class object counting dataset known as KR-GRUIDAE is presented in this work. The advantage is that EcoCountNet's network contributes to accurate and efficient counting processes. The disadvantage is that the network requires additional computational resources and real-time applications are not explored due to its complexity. Yongtuo *et al.,* [8] have presented crowd counting model adaptability across different domains by combining two modules Crowd Region Transfer (CRT) and Crowd Density Alignment (CDA). The merit is that the model exhibits promising performance in various adaptation scenarios. The drawback is that the point-level crowd-counting annotations for crowd images are still challenging problems and expensive.

Savchenko *et al.,* [9] have presented an efficient frame-level facial emotion analysis model that combines embeddings and scores from the EfficientNet architecture pre-trained on AffectNet. The merit is that the model outperforms the baseline on multiple tasks, including facial expression recognition, and valence-arousal estimation. The demerit is that generalizing the model to real-world scenarios is a challenging problem. Justin *et al.,* [10] have discussed fine-grained counting by using crowd-sourced annotations to estimate individuals in crowded scenes and also classify attributes. The merit is that *The Seal Watch dataset* contains eight fine-grained classes that advance research in animals. The detection-based methods for behavior analysis are not implemented in this work.

Pierre *et al.,* [11] have designed crowd behavior by exploring various approaches such as microscopic, macroscopic crowd modeling, motion-based crowd behavior analysis, and optical flow utilization. The advantage of crowd analysis is that it is applied in areas such as public safety, market analysis, urban planning, and entertainment, and the disadvantage is that it generates large volumes of data, which is challenging to store, process, and analyze efficiently. Shenjian *et al.,* [12] have presented a bi-level alignment framework for enhancing synthetic-to-real Unsupervised Domain Adaptation (UDA) crowd counting. The merit is that the model addresses domain adaptation problem, while demerit is that it involves increased computational complexity. Sachin et al., [13] have predicted and counted crowd behavior using a Multicolumn Convolutional Neural Network (MCNN) on the ShanghaiTech dataset. Merit is that enhanced crowd management in various real-world scenarios, demerit is that

* *E-mail:* gangamohanraj2003@gmail.com

diverse image sizes and resolution need to be addressed in this work.

Author /Year	Algorithm/ Model	Dataset	Advantages	Disadvantages
Cem *et al.,* [4], 2020	MIIs from optical flow	UMN and PETS2009	Improved accuracy	Computational intensity
Shuqiang *et al.,* [5], 2019	Mean shift and *k*--means	UMN data set	Improved accuracy	Parameter tuning
Junyu *et al.,* [6], 2020	MFA and SDA	Shanghai Tech Part B, WorldExpo'10, Mall and UCSD	Overcoming data scarcity	Error in similar textures
Hyojun *et al.,* [7], 2021	EcoCountNet	KR-GRUIDAE (fine-grained object counting)	Accurate counting	Additional resources and not real-time
Yongtuo *et al.,* [8], 2023	CRT and CDA	GCC, ShanghaiTech PartA	Promising adaptation	Annotation costly
Savchenko *et al.,* [9], 2022	EfficientNet	AffectNet	Outperforms baseline	Generalization challenge
Justin *et al.,* [10], 2022	Crowd-sourced fine-grained	Seal Watch dataset	Enhanced crowd management	Lack of size discussion
Pierre *et al.,* [11], 2019	Various approaches	The UCSD Anomaly Detection Dataset	Applicable in diverse areas	Large data challenge
Shenjian *et al.,* [12], 2022	Bi-level alignment	GTA5 Crowd Counting (GCC) dataset	Addresses domain adaptation	Increased complexity
Sachin *et al.,* [13], 2023	MCNN	ShanghaiTech dataset	Enhanced crowd management	Size and resolution challenges
Zhikang *et al.,* [14], 2019	Count attention	Shanghaitech dataset	Improved accuracy	Future applicability
Ye *et al.,* [15], 2019	DFFnetSeg	Test dataset from CDnet2014	Handles modification	Increased computational complexity
Adel *et al.,* [16], 2021	U-ASD Net	Haramain with three different scenes	Adapts to scenarios	More computational resources
Elizabeth *et al.,* [17], 2021	Integrated approach	Crowd datasets	Early detection	Data availability challenges
Xiaoheng *et al.,* [18], 2020	DANet and ASNet	ShanghaiTech Part A, UCF CC 50, UCF-QNRF, and WorldExpo'10	Alleviates density differences	More accurate counting
Naveed *et al.,* [19], 2021	CNN-based model	Shanghaitech (Part-A), Shanghaitech (Part-B), and Venice	Improved accuracy	Semantic segmentation expansion
Yadi *et al.,* [20l], 2023	AI-based analytics	Video records from a platform scenario	Accurate pedestrian counting	Advanced technology required
Reem et al., [22], 2023	Enhanced abnormal detection	Diverse Hajj dataset	Impressive results with scalability	Model complexity

Table 1. Literature Survey of Crowd Behavior Analysis Models.

Zhikang *et al.,* [14] have presented count attention and the adaptive model, to allocate different capacities to different regions in an image based on crowd density. The merit is that it improves crowd-counting accuracy in various scenarios. The broader ranges of scenarios like videos are not experimented in this work. Ye *et al.,* [15] have presented the foreground/background segmentation method DFFnetSeg for video analysis. The merit is that the model handles both unseen and changes in scene making it suitable for diverse video scenarios. The drawback is that the increase in computational complexity due to multiple frames. Adel *et al.,* [16] have

proposed U-ASD Net concatenating U-Net and Adaptive Scenario Discovery to address perspective distortions and scale variations in crowd counting. It shows adaptation to complex scenarios making it suitable for both dense and sparse datasets.

E-mail: gangamohanraj2003@gmail.com

The demerit of the model is that it requires more computational resources due to the number of parameters and training time.

Elizabeth *et al.,* [17] have explored crowd behavior analysis by integrating psychology theories, IoT, and cognitive computing for predictive crowd management. The advantages are that it enables early detection of crowd disasters and the potential for real- time data processing. However, there are challenges in data availability. Xiaoheng *et al.,* [18] have addressed the challenges of crowd counting using Convolutional Neural Networks (CNNs) by combining two networks namely Density Attention Network (DANet) and Attention Scaling Network (ASNet). It improves counting performance differences in regions with varying crowd density patterns, leading to more accurate crowd counting. Naveed *et al.,* [19] have proposed a CNN-based model comprising of three main modules *viz*., a backbone network for general features, Dense Feature Extraction Modules (DFEMs) with dense connections, and a Channel Attention Module (CAM) for class – specific responses. It improves counting accuracy in scenes with large perspective variations and varying density levels. The semantic segmentation is not explored in this work.

Yadi *et al.,* [20] have presented a comprehensive AI-based model for crowd analytics in rail transit stations, focusing on flow volume, crowd density, and walking speed analysis. It successfully achieves accurate pedestrian counting and practical applications, while requiring advanced technology and calibration equipment. Reem *et al.,* [21] have enhanced abnormal behavior detection in large crowds using the diverse Hajj dataset. While achieving impressive results, it faces challenges in real-world scalability and model complexity. Ahmed *et al.,* [22] have developed a cloud-based deep learning framework for early detection of crowding in event entrances. It demonstrates 87% accuracy but encounters real-time implementation challenges and environmental influences. Ganga *et al.,* [23] have proposed a crowd counting method combining Unet and GAN architectures to generate crowd density maps with minimized feature loss. Ganga *et al.,* [24] have presented AnomalyDetectNET for video anomaly detection.

3. BACKGROUND

The fine-grained crowd-counting [1] method features a two-branch architecture consisting of density-aware feature propagation and complementary attention mechanisms. In the density-aware feature propagation phase, the model iteratively propagates features to capture contextual information, with a specific emphasis on high-density areas and to predict the overall crowd density map. In the complementary attention mechanisms, the information is exchanged between the two branches and individual pixels are categorized effectively. Furthermore, the model during training combines three loss functions: counting loss, segmentation loss, and fine-grained loss, to optimize its performance. The method shows high accuracy of crowd counting, in scenarios where fine-grained categorization of crowd segments is necessary. Ground-truth density maps Y_j are generated by convolving dot maps D_j with a 2D Gaussian kernel K_σ as shown in equation (1).

$$Y_j = D_j * k_\sigma \quad (1)$$

Ground-truth segmentation maps S_j are obtained from the ground-truth density maps as shown in equation (2).

$$S_j = \frac{Y_j}{\eta + \sum_{j=1}^{k} Y_j} \quad (2)$$

Here η is a small number to prevent division by zero, and background segmentation (S_{K+1}) is defined by regions with low density [1]. The soft cross-entropy loss given in equation (3) is used for segmentation.

$$\text{soft CE Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C}\left(Y_{ij}.\log\left(\hat{Y}_{ij}\right)\right) \quad (3)$$

Where Y_{ij} represents the ground truth class probability, $\hat{Y}_{ij}$ is the predicted class probability for class, N is the total number of data points, and C is the number of classes.

3.1. Problem Definition

Given a Video clip/data consisting of images of a certain length and situations, the objectives are to explore Behavioral Crowd Counting (BCC), by combining the CSRNet and Unet.

3.2. Objectives

1. BCC Architecture: To design and construct the Behavioral Crowd Counting (BCC) architecture by integrating the CSRNet and Unet to enable behavioral crowd counting in video data.
2. Enhanced Accuracy and Segmentation: To improve the accuracy of crowd counting and crowd segmentation by accurately distinguishing crowd regions from the background and analyzing crowd behaviors in video data.
3. Optional Attention Map Integration: To integrate optional attention maps into BCC to focus on specific areas of interest within the crowd to refine density predictions.

3.3. Architecture of Behavioral Crowd Counting (BCC)

The Behavioral Crowd Counting (BCC) architecture as shown in Figure 1 is a concatenation of Congested Scene Recognition Network (CSRNet) and Unet to achieve better crowd counting by including behavior in video data. CSRNet is renowned for its precision in estimating crowd density within congested scenes, offering a deep understanding of the intricate details in densely populated areas. The merit of CSRNet is that it excels in capturing contextual information, allowing it to comprehend the spatial relationships among individuals within a crowd. The CSRNet consists of two major components *viz.*, a frontend and a backend network. The frontend network consists of several convolution and max-pooling operations that are used for feature extraction. The backend network incorporates dilated convolutions for feature extraction and comprises a series of convolution layers that process features extracted by the frontend network, producing the estimated crowd density map. Tables 2 and 3 show the Frontend Network (Feature Extraction) and Backend Network (Density Map Estimation) layers for their input and output channel dimensions, kernel sizes, stride values, padding, and activation functions.

The CSRNet is initialized with weights, and the front end can be pre-trained optionally on ImageNet.The frontend has the following methods namely forward method, make

E-mail: gangamohanraj2003@gmail.com

layers and initialize. The forward method processes input images, extracts features to produce crowd density and semantic segmentation maps. The make layers methods generate sequential layers based on the provided configuration, including convolution layers, batch normalization, and ReLU activations. The initialize method is responsible for initializing the weights of the network modules, viz., convolution layers and batch normalization. Overall, the architecture employs VGG16-inspired features for effective feature extraction in crowd analysis, with a specific focus on counting and semantic segmentation. The backend with dilated convolutions efficiently processes input images to produce accurate crowd density and segmentation maps.

The Unet incorporates a two-tiered structure namely encoders serve as feature extractors and decoders for processing input features for segmentation as shown in Table 4. The encoder consists of two convolution layers, *LeakyReLU* activation and *Max-pooling*. The convolution layers in each of its two blocks, followed by *Leaky ReLU* activation functions. Leaky ReLU activation functions initiate non-linearity to the network that allows a small, non-zero gradient for negative inputs, preventing dead neurons and facilitating the learning of more complex relationships in the data. Max-pooling operations follow each encoder block to down-sample and capture hierarchical and spatial information. The down-sampling is pivotal for progressive abstracting and concentrating relevant information from the input images, enabling the network to learn hierarchical representations while maintaining computational efficiency.

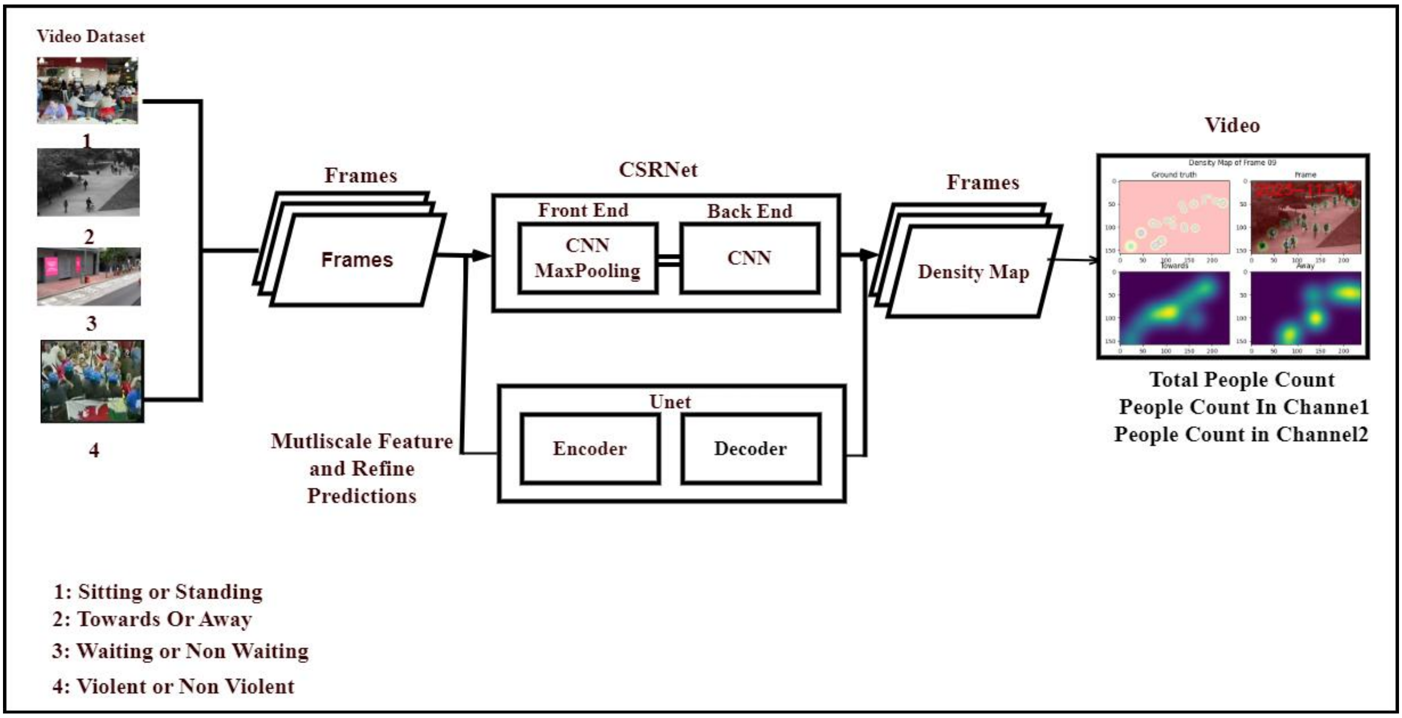

Figure 1: The Architecture of Behavioral Crowd Counting (BCC).

Table 2: Frontend Network (Feature Extraction).

Layer	Input Channels	Output Channels	Kernel Size	Stride	Padding	Activation Function
Convolutional Layer 1	1	64	5x5	1	2	Leaky ReLU (0.1)
Max-Pooling Layer 1	64	64	2x2	2	0	None
Convolutional Layer 2	64	128	5x5	1	2	Leaky ReLU (0.1)
Max-Pooling Layer 2	128	128	2x2	2	0	None
Convolutional Layer 3	128	256	5x5	1	2	Leaky ReLU (0.1)
Max-Pooling Layer 3	256	256	2x2	2	0	None
Additional Convolutional	Varies	Varies	Varies	Varies	Varies	Varies

E-mail: gangamohanraj2003@gmail.com

Table 3: Backend Network (Density Map Estimation).

Layer	Operation	Input Channels	Output Channels	Kernel Size	Stride	Padding
Encoder Conv1	Conv2d, LeakyReLU, Conv2d, LeakyReLU	i_cn	64	3x3	1	1
Max Pooling 1	MaxPool2d	64	64	2x2	-	-
Encoder Conv2	Conv2d, LeakyReLU, Conv2d, LeakyReLU	64	128	3x3	1	1
Max Pooling 2	MaxPool2d	128	128	2x2	-	-
Decoder Upconv1	ConvTranspose2d	128	64	2x2	2	0
Decoder Conv1	Conv2d, LeakyReLU, Conv2d, LeakyReLU	128	64	3x3	1	1
Decoder Upconv2	ConvTranspose2d	64	o_cn	2x2	2	0
Refinement Layer	Conv2d	64	o_cn	1x1	1	0

Layer	Input Channels	Output Channels	Kernel Size	Dilation	Activation Function
Convolutional Layer 1	512	512	3x3	Optional	None
Convolutional Layer 2	512	256	3x3	Optional	None
Convolutional Layer 3	256	128	3x3	Optional	None
Convolutional Layer 4	128	64	3x3	Optional	None
Output Layer	64	1	1x1	None	None

Table 4: Unet Architecture.

The decoder has transposed convolution layers which are used for up-sampling the feature maps, with two such layers in each decoder block. These layers contribute to the reconstruction of spatial details lost during the down-sampling process, aiding in the precise localization of features. The transpose convolution layers refine the feature representations, maintaining symmetry with the encoder and the final refinement occurs through a 1x1 convolution layer in the refinement block. The Unet benefits are particularly effective in tasks such as image segmentation where accurate spatial delineation is essential. It also effectively performs image segmentation tasks and its emphasis on accurate spatial delineation makes it potent in applications where precise localization of features is essential, such as medical image analysis, autonomous vehicle navigation, and satellite image processing

3.4. Algorithm

The algorithm 1 explains *Pre Processing, Segmentation of video* and *Crowd Counting*. The *Pre Processing* function is designed to handle video data. It takes a video path as input and performs the following tasks, such as frame extraction, saving frames to an output directory, and reading annotation data from a corresponding JSON file. The function utilizes *OpenCV* to read and store frames, organizing them based on the video number in a specified directory structure. The output of preprocessing is the total number of frames, the path to the frame outputs, extracted annotation data, and the video number. The *Segmentation* function focuses on video segmentation. It calls the *processing results* function to obtain a result and then iterates through each frame. For each frame, the data is prepared, and the

RefineSegmentation function is applied to refine the segmentation using an Unet model. The function returns the final segmented result for each frame in the video. It takes input data containing features and an attention map, concatenates them if an attention mechanism is used, initializes the Unet model, and performs segmentation refinement. The refined segmentation result is then returned.

In the *Crowd Counting* function, the video undergoes crowd counting processing. It first calls the *IntegratedProcessing* function to obtain a processing result. The function then iterates through each frame, prepares the data, and uses the *PredictCrowdCount* function to estimate the crowd count. The final result represents the crowd count information for each frame in the video.

The PredictCrowdCount function estimates crowd counts in images. It takes an image as input, initializes a CSRNet model for crowd counting, processes the image through its frontend and backend, and predicts the crowd count using the output layer of the model. The result is the predicted crowd count for the given image. The algorithm Enhanced Behavioral Crowd Counting (EBCC) accurately counts individuals in crowded scenes by considering their behavior such as the direction of movement (towards or away), posture (standing or sitting), state of people (waiting or non waiting), and nature of the activity (violent or non-violent). It preprocesses video frames to extract and refine segmentation data and then employs a combination of Unet and CSRNet models to analyze these frames. The process counts

`*E-mail:* gangamohanraj2003@gmail.com

the number of people by integrating behavioral analysis, leading to a more comprehensive understanding of crowd dynamics. The versatility of EBCC allows it to be adapted for other applications, such as monitoring public safety and enhancing surveillance systems for behavioral anomaly detection. The algorithm's time complexity is *O(n),* with *n* representing the total frames in a video, as it processes each frame separately. The space complexity is *O(m),* where *m* is the memory requirements of the Unet and CSRNet models during the algorithm's runtime.

Function 1: PreProcessing

```
Function 1:Pre Processing (video_ path)
  Input : video_ path , file_ name
  Output: total_ frames, output_ path, annotation_ data, video_ num
  dataset_ path ← 'D:/Final_ yr_ project/Final_ yr_ project/dataset_ frames'
  file_ name ← video_ path
  video_ num ← substring ( f ile_ name)
  frames_ dir ← frames + video_ num
  output_ path ← Call join (dataset_ path, frames_ dir)

      if not exists (output_ path) then
      |     makedirs (output_ path)
      end
        video ← VideoCapture (video_ path)
        total_ frames ← int ( video.get ( CAP_PROP_FRAME_COUNT ) )
        for frame_ count ← 0 to total_ frames do
            success, frame ← readvideo
            If not success then
            | break
            end
            frame_ path ← output_ path, frame_ count
            Call imwrite (frame_ path, frame ) frame_ count ← frame_ count  + 1
         end
      release (video )
      print "Video extraction complete! "
      annotation_ file ← open D : /Final_ yr_ project/Final_ yr_ project/annotations/video_ {video_ num}. json
      annotation_data ← json.load (annotation_ file)
      return total_ frames, output_ path, annotation_ data, video_ num
```

Function 2: Segmentation of Video

```
Function 2: Segmentation (video path)
  Input : video path
  Output: Segmentation result
  processing result ← video path
  Extract information from processing result
     for frame index ← 0 to total frames do
     |  frame path ← join(output path, frame_ index)
     |  input data ← PrepareData(frame_ path)
     |  segmentation result ← RefineSegmentation(input_ data)
     |  Process segmentation_result
     end
  return Segmentation result
Function RefineSegmentation()
  Input: InputData
  Output: RefinedSegmentation
  // Extract features and attention map from InputData
  // fea: Extracted features from the input.
```

E-mail: gangamohanraj2003@gmail.com

```
// att: Attention map used in the segmentation process
  fea ← ['fea'] att ← ['att'] // Assuming att is the attention map
// Check if attention mechanism is used
    if att is not None then
     | fea ← torch.cat((fea, att), 1)
    end
// Initialize Unet model for segmentation refinement
  unet_ model ← Unet Model(input_ channels, output_ channels)
// Perform segmentation refinement using the Unet model
  RefinedSegmentation ← unet_ model(fea)
return RefinedSegmentation
```

Algorithm 1 EBCC: Enhanced Behavioural Crowd Counting

```
Input : video_ path
Output: Crowd Count
Function 1 : Pre processing (video_ path)
Function 2: Segmentation  (video_ path)
Function 3: Crowd Counting (video_ path)
   Input : video_ path
   Output: Crowd Count  processing _result ← IntegratedProcessing(video_ path)
   Extract information from processing_ result
   for frame index ← 0 to total frames do
      frame_ path ←  join(output_ path, frame_ index)
      input data  ←  PrepareData(frame_ path)
      crowd count ←  input data Process crowd_ count
   end
return Crowd Count
Function 4: PredictCrowdCount(x)
   Input: Image x
   Output: Predicted crowd count dmap
   // Initialize CSRNet model
   i_ cn = number of input channels
   o_ cn = number of output channels
   csrnet model ← CSRNet(i_cn, o_cn)
   // Frontend Processing
   x ← csrnet_ model.frontend(x)
   // Density Map Feature Extraction
   dmap_f ea ← csrnet_ model.backend(x)
   // Density Map Prediction
   dmap ← csrnet_ model.output_ layer(dmap_f ea)
   return dmap
```

4. EXPERIMENTS SETUP

The loss functions such as Mean Squared Error (MSE) and Mean Absolute Error (MAE) are the metrics used for evaluation. In Table 5, details of four behavior video datasets such as frame rate in seconds, length of video in seconds, and resolutions of the videos are shown.

4.1. Datasets

1) Standing vs Sitting Video Dataset: The dataset presents videos derived from a fine-grained image dataset, illustrating individuals either standing or sitting in various urban settings. It provides a valuable resource for analyzing and distinguishing between static postures in different environmental contexts.

2) Waiting vs Non Waiting Video Dataset: The dataset presents videos built from a fine-grained image dataset capturing the people waiting (e.g., at bus stops) versus non waiting those engaged in other activities. It's an excellent tool for studying patterns of stationary and transient behaviors in public spaces.

3) Towards vs Away Video Dataset: The dataset presents videos built from a fine-grained image dataset featuring people walking towards or away from the camera. It aids in understanding directional movement and pedestrian dynamics, offering insights into approach and departure behaviors in various settings.

4) Violent Vs Non-Violent Dataset: The dataset presents videos built from a fine-grained image dataset having both violent and non-violent video scenes. This resource

E-mail: gangamohanraj2003@gmail.com

differentiates aggressive and non-aggressive behaviors in different contexts.

Table 5. Details of Four Behaviour Video Dataset.

Dataset Name	Frame Rate(sec)	Length(sec)	Resolution(dpi)
Violent vs Non-violent	1	1	96
Towards vs Away	1	1	96
Standing vs Sitting	1	1	96
Waiting vs Non-Waiting	1	1	96

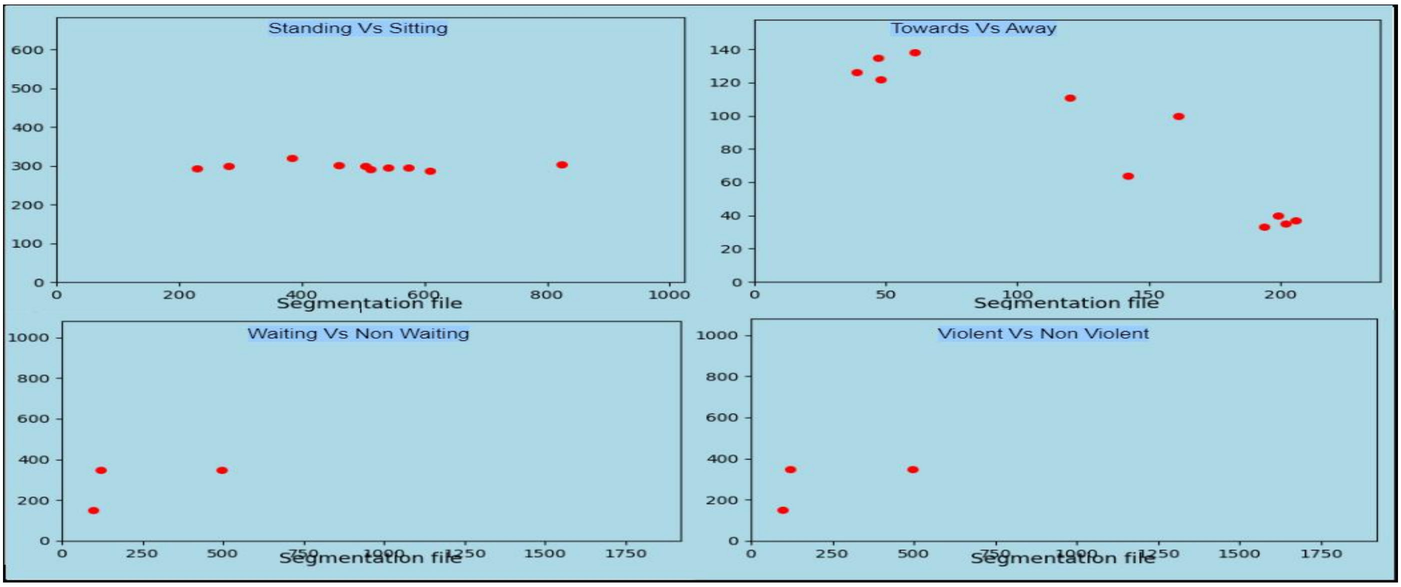

Figure 2: Segmentation Results of Four Video Dataset

4.2. Metrics

(i) Mean Square Error (MSE) is calculated as the average error between the predicted density values and the ground-truth density values as shown in equation (4).

$$\text{MSE Loss} = \frac{1}{N}\sum_{i=0}^{N}(Y_i - \widehat{Y}_i)^2 \qquad (4)$$

Where Y_i represents the ground truth density map, $\widehat{Y}_i$ is the predicted density map, and N is the total number of data points.

(ii) Mean Absolute Error (MAE) is calculated as the average error between the predicted density values and the ground-truth density values as shown in equation (5).

$$MAE = \frac{1}{n}\sum |\sum Y_{ij} - \sum \widehat{Y}_{ij}| \qquad (5)$$

Where n is the total number of test images, Y_{ij} represents the ground-truth density map for the i^{th} test image and the j^{th} category , $\widehat{Y}_{ij}$ represents the predicted density map for the i^{th} test image and the j^{th} category.

4.3 . Performances

Figure 2 shows the Segmentation Results of the four video datasets with segmentation masks generated. A segmentation mask is a pixel-wise labeling of an image, where each pixel is assigned a category or class based on certain characteristics. The segmentation mask is generated to highlight specific regions of interest within images, guided by annotated points provided in the annotation data. The resulting segmentation mask provides a spatially detailed representation of the annotated features within the image, effectively delineating these features from the background.

Figure 3 shows the mask results of the four video datasets with the binary masks generated. Each mask isolates specific regions of interest within the corresponding images by assigning binary values to pixels. The regions of interest are determined by annotated points obtained from the annotation data associated with the images. The masks are created to selectively highlight and distinguish particular features within the images, as dictated by the annotated points. The grayscale intensity in the mask corresponds to the binary values assigned to pixels, where white (or lighter shades) typically represent annotated regions (binary value 1), and black (or darker shades) represent non-annotated 10 areas (binary value 0). The binary

E-mail: gangamohanraj2003@gmail.com

masks play a vital role in the precise examination of object detection, segmentation, and feature analysis.

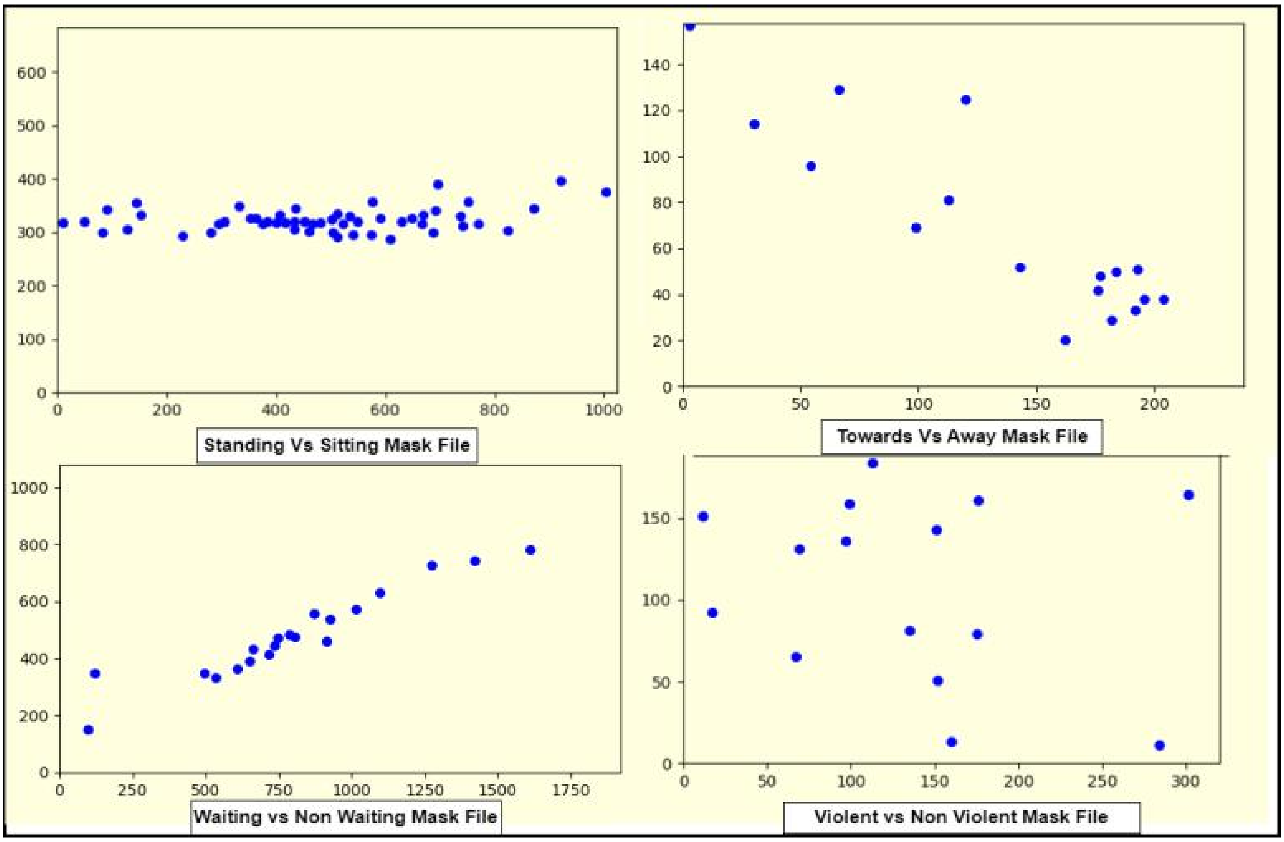

Figure 3. Mask Results of Four Video Dataset.

5. Results and Discussions

In Figure 4, the *Video Information* provides details about the video, including the frames per second (1 FPS), total frames (6), and timestamp. The *Frame Extraction Results* in Figure 5 shows the extraction of each frame. In Figure 6, *the Frame Analysis Results* displays frame numbers and their respective Comparative Mean Absolute Error (CMAE) values. Lower CMAE values indicate more accurate predictions. Figures 4, 5, and 6 offer a comprehensive overview of the video-to-frame conversion process, presenting key metadata, analytical outcomes, and the process of converting a video into frames.

Tables 6 and 7 provide a comprehensive summary of video analysis metrics, such as processing speed(Frames per Second), total frames, video creation time, average object speed, total number of people, average count of people moving towards and away, Comparative Mean absolute Error (CMAE), Mean Absolute Error (MAE), Mean Squared Error (MSE), and a PATCH metric for all four datasets. These metrics offer insights into the dynamics,

Property	Value
Frames per second	1
Total frames	6
Video created at	2023-06-29 20:39:56

Figure 4. Video Information.

accuracy, and characteristics of crowd behavior across diverse scenarios and dataset

Figure 7 presents the experimental results for four datasets i.e., Towards/Away, Standing/Sitting, Waiting/Non Waiting and Violent/Non-Violent using four separate figures for each dataset. In the Towards/Away dataset, the figures visualize crowd movement direction featuring annotated ground truth, crowd frames, and the number of people moving towards and away. For the Standing/Sitting dataset, the results focus on postures (standing or sitting), displaying annotated ground truth, crowd frames, and the number of

`E-mail: gangamohanraj2003@gmail.com

people standing and sitting. In Waiting/Non Waiting dataset figures show the annotated ground truth, crowd frames, and number of waiting and non waiting people. The Violent/Non-Violent dataset figures illustrate instances of violence, showcasing annotated ground truth, crowd frames,and the number of violent and non-violent actions within the crowd.

In Table 8, Comparative Mean Absolute Error (CMAE) results of the proposed BCC model compared with the fine grain crowd counting model of Jia [1] across diverse video datasets are presented. Across various human behaviors such as Standing/Sitting, Waiting/Non-Waiting, Towards/Away, and Violent/Non-Violent categories, the BCC model consistently outperforms the base paper, with lower CMAE values. For instance, in the Standing/Sitting category, the BCC model achieves a CMAE of 5.06, excelling the base paper's 8.01. This trend persists across Waiting/Non-Waiting, Towards/Away, and Violent/Non-Violent categories, with the BCC model demonstrating CMAE values of 4.12, 3.16, and 4.25, respectively, compared to the base paper's 2.99, 2.29, 2.29,and,4.35.

In Table 9, comprehensive overview of key metrics with crowd behavior analysis across four datasets is presented. Each row corresponds to a specific category, namely Standing vs Sitting, Waiting vs Non-Waiting, Towards vs Away, and Violent vs Non-Violent. The provided metrics include frames per second, total frames, the date and time of video creation, average object speed, total people count, counts in two designated channels, and several evaluation metrics (CMAE, MAE, MSE, and PATCH). These metrics present insights into the characteristics, accuracy, and dynamics of crowd behavior within diverse scenarios and behavioral categories .

A contributing factor to the BCC model's enhanced accuracy lies in the incorporation of the Unet architecture. The Unet captures spatial dependencies and intricate patterns in crowd behavior. Its features such as a contracting path for context capture and an expansive path for precise localization identify even small differences in scenes with many people. The integration of the Unet with CSRNet architecture boosts the BCC model's ability to analyze complex spatial relationships, emphasizing the importance of an advanced neural network for superior performance in crowd analysis applications.

Figure 8 illustrates the outcomes of three key metrics—Mean Absolute Error (MAE), Comparative Mean Absolute Error (CMAE), and Mean Squared Error (MSE) across four distinct video datasets. The *x*-axis corresponds to the individual datasets, while the *y*-axis to the metric values for each dataset. The data points visually represent the model performance based on MAE, CMAE, and MSE, with lower values indicating more accurate predictions. The visual summary enables a comparison of model effectiveness across the various video datasets concerning these specific evaluation metrics.

Extracted frame: 1/6
Extracted frame: 2/6
Extracted frame: 3/6
Extracted frame: 4/6
Extracted frame: 5/6
Extracted frame: 6/6
Extraction complete!

Figure 5. Frame Extraction Results.

Frame Number: 1, CMAE: 3.36
Frame Number: 2, CMAE: 2.66
Frame Number: 3, CMAE: 3.50
Frame Number: 4, CMAE: 2.21
Frame Number: 5, CMAE: 1.78
Frame Number: 6, CMAE: 1.64

Figure 6. Frame Analysis Results

Table 6. Comparison of Metrics (CMAE, MAE, and MSE)

Table 6(a). Waiting vs Non Waiting

Metric	**Value**
Frames per second	1
Total frames	2
Video created at	2023-11-20 19:25:06
Average Speed	60.44 pixels per frame
Total no of people	12.50
Avg no of people in Channel1	10.50
Avg no of people in Channel2	2.00
CMAE	4.12
MAE	2.56
MSE	9.63
PATCH	5.78×10^{-6}

Table 6(b). Standing vs Non Standing

Metric	**Value**
Frames per second	1
Total frames	7
Video created at	2023-11-20 19:16:18
Average Speed	129.92 pixels per frame
Total no of people	56.57
Avg no of people in Standing	39.43
Avg no of people in Sitting	17.14
CMAE	5.06
MAE	3.64
MSE	15.12
PATCH	3.85×10^{-5}

`*E-mail:* gangamohanraj2003@gmail.com

Table 7. Comparison of Metrics (CMAE, MAE, and MSE)

Table 7(a). Violent vs Non Violent Data

Metric	Value
Frames per second	2
Total frames	10
Video created at	2023-11-20 19:19:40
Average Speed	0.56 pixels per frame
Total no of people	21.10
Avg no of people in Towards	12.60
Avg no of people in Away	8.50
CMAE	3.16
MAE	3.09
MSE	6.20
PATCH	0.0005573

Table 7(b). Towards vs Away Data

Metric	**Value**
Frames per second	1
Total frames	6
Video created at	2023-11-20 19:21:44
Average Speed	20.18 pixels per frame
Total no of people	28.67
Avg no of people in Violent	21.67
Avg no of people in Non Violent	7.00
CMAE	4.25
MAE	3.26
MSE	10.64
PATCH	0.0003048

Table 8. CMAE Results on four Video Datasets

Model	Standing/Sitting	Waiting/NonWaiting	Towards/Away	Violent/NonViolent
BCC(Proposed Model)	5.06	4.12	3.16	4.25
Jia(Base Paper) [1]	8.01	2.99	2.29	4.35

Table 9. Comprehensive overview of BCC Achitecture metrics across four datasets

Category	Frames (sec)	Total Frames	Date Created	Time Created	Avg Speed	Total Speed	Channel 1	Channel 2	CMAE	MAE	MSE	PATCH (10^{-5})
Standing vs Sitting	1	7	2023-11-20	19:16:18	129.92	56.57	39.43	17.14	5.06	3.64	15.12	3.85
Waiting vs nonWaiting	1	2	2023-11-20	19:25:06	60.44	12.50	10.50	2.00	4.12	2.56	9.63	5.78
Towards vs Away	2	10	2023-11-20	19:19:40	0.56	21.10	12.60	8.50	3.16	3.09	6.20	5.57
Violent vs NonViolent	1	6	2023-11-20	19:21:44	20.18	28.67	21.67	7.00	4.25	3.26	10.64	3.04

Figure 9 (a) shows the Comparative Mean Absolute Error (CMAE) results across four categories, each representing different aspects within the visual data. The *x*-axis denotes the individual categories, while the *y*-axis the corresponding CMAE values. Lower CMAE values on the graph signify higher accuracy in the model's predictions for each category. The graph shows the comparison, highlighting the model behaviors within each specific category based on the calculated CMAE values.

Figure 9 (b) presents the performance between the BCC (Proposed Model) and Jia (Base Paper) in predicting crowd behaviors. In Standing/Sitting, BCC boasts a CMAE of 5.06 compared to Jia's 8.01. For Waiting/Non-Waiting, BCC records a CMAE of 4.12, surpassing Jia's 2.99. In Towards/Away, BCC achieves a CMAE of 3.16 against Jia's 2.29. Lastly, in Violent/Non-Violent, BCC demonstrates a CMAE of 4.25, outperforming Jia's 4.35. These numerical comparisons proves the accuracy of the BCC model over the Jia model across diverse crowd behavior categories.

E-mail: gangamohanraj2003@gmail.com

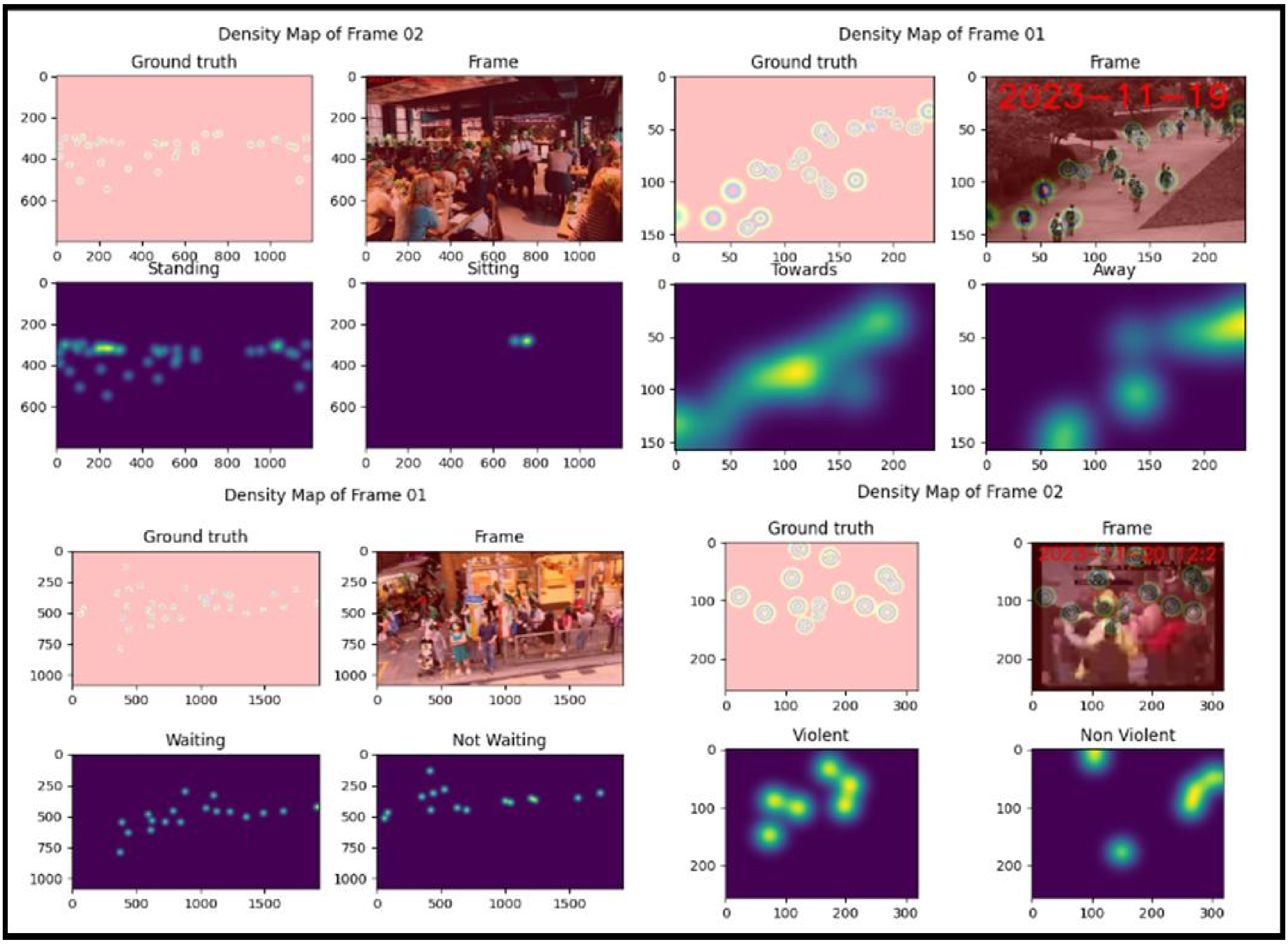

Figure 7. Results of BCC architecture with four Dataset

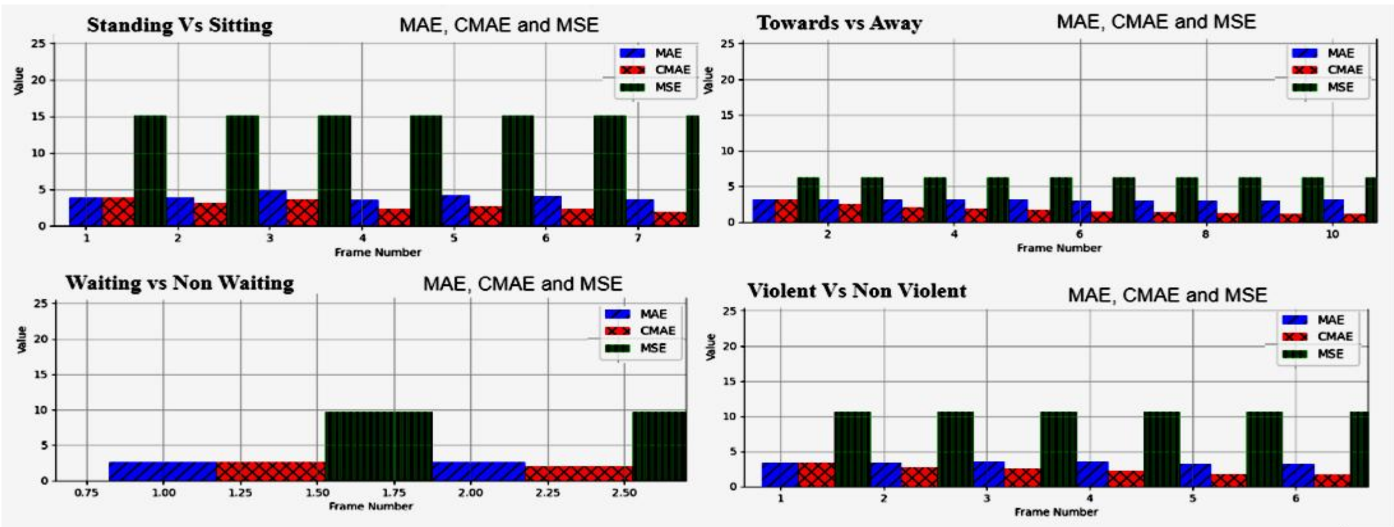

Figure 8. Results of MAE, CMAE, MSE Metric of Four Datasets

E-mail: gangamohanraj2003@gmail.com

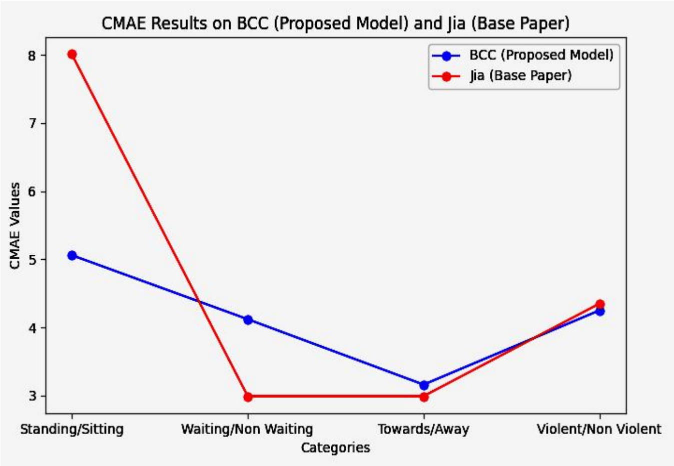

Figure 9(a). Bar Graph Results of CMAE for each Category

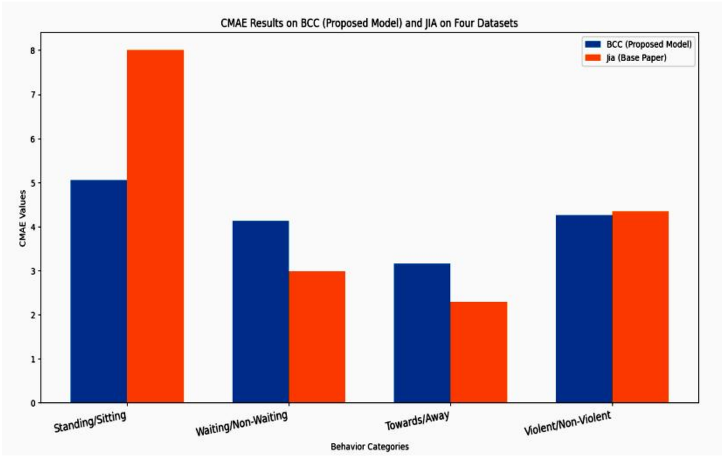

Figure 9 (b). CMAE between BCC and JIA models

6. CONCLUSION

The Behavioral Crowd Counting (BCC) architecture combines a Congested Scene Recognition Network (CSRNet) with an Unet for enhanced behavioral crowd counting. The CSRNet consists of a frontend and a backend network for feature extraction and generation of a crowd density map. The Unet produces a density map and refines an attention-based map. It operates on video features and, attention maps, refining the density map through several iterations. The refined density maps provide behavior-based crowd segmentation, separating crowd regions from the background with improved accuracy. The experimental results validate the effectiveness of the approach in behaviour crowd counting in video data consisting of congested scenes. This synergy empowers the system to perform behavioral crowd counting, offering unprecedented insights into crowd dynamics within video datasets.

Extending BCC to recognize, and analyze emotions or sentiments within crowd enables applications in marketing, entertainment, and event management. Incorporating multi-modal inputs from different data sources, such as audio, text, or social media data, provides a more comprehensive understanding of crowd behavior and improves analysis accuracy.

REFERENCES

[1] J. Wan, N. Senthil Kumar, and A. B. Chan, "Fine-Grained Crowd Counting," *IEEE Transactions on Image Processing*, vol. 30, pp. 2114–2126, 2021. doi: 0.1109/TIP.2021.3052815.
[2] L. Vibha, C. Hegde, P. D. Shenoy, K. R. Venugopal, and L. M. Patnaik, "Dynamic Object Detection, Tracking and Counting in Video Streams for Multimedia Mining," *IAENG International Journal of Computer Science*, vol. 35, no. 3, pp. 16–21, 2008. IJCS 35 3 16.
[3] L. Vibha, C. Hegde, P. D. Shenoy, K. R. Venugopal and L. M. Patnaik, "Moving Vehicle Identification using Background Registration Technique for Traffic Surveillance," in Proc. of *The International MultiConference of Engineers and Computer Scientists*, vol. 1, 2008. ISBN: 978-988-98671-8-8.
[4] C. Direkoglu, "Abnormal Crowd Behavior Detection using Motion Information Images and Convolutional Neural Networks," *IEEE Access,* vol.8, pp. 80408–80416, 2020. doi: 10.1109/ACCESS.2020.2991447.
[5] S. Guo, Q. Bai, S. Gao, Y. Zhang, and A. Li, "An Analysis Method of Crowd Abnormal Behavior for Video Service Robot," *IEEE Access*, vol. 7, pp. 169577–169585, 2019. doi: 10.1109/ACCESS.2019.2953774.
[6] J. Gao, Y. Yuan, and Q. Wang, "Feature-Aware Adaptation and Density Alignment for Crowd Counting in Video Surveillance," *IEEE Transactions on Cybernetics*, vol. 51, no. 10, pp. 4822–4833, 2020. doi:10.1109/TCYB.2020.2983862.
[7] H. Go, J. Byun, B. Park, M.A. Choi, S. Yoo, and C. Kim, "Fine-Grained Multi-Class Object Counting," *IEEE Transactions on Cybernetics in 2021 IEEE International Conference on Image Processing (ICIP)*, pp. 509–513, 2021. doi:10.1109/ICIP42928.2021.9506384.
[8] Y. Liu, D. Xu, S. Ren, H. Wu, H. Cai, and S. He, "Fine-Grained Domain Adaptive Crowd Counting via Point-Derived Segmentation," *IEEE International Conference on Multimedia and Expo (ICME),* pp. 2363–2368, 2023. doi:10.1109/ICME55011.2023.00403.
[9] A. V. Savchenko, "Video-Based Frame-Level Facial Analysis of Affective Behavior on Mobile Devices using Efficientnets," *in Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2359–2366, 2022. doi: 10.1109/CVPRW56347.2022.00263.
[10] J. Kay, C. M. Foley, and T. Hart, "Fine-Grained Counting with Crowd-Sourced Supervision," *arXiv preprint arXiv:2205.11398*, 2022. arXiv:2205.11398.
[11] P. Bour, E. Cribelier, and V. Argyriou, "Crowd Behavior Analysis from Fixed and Moving Cameras," in *Multimodal Behavior Analysis in the Wild*, pp. 289–322, 2019. doi:10.1016/B978-0-12-814601-9.00012-1.

E-mail: gangamohanraj2003@gmail.com

[12] S. Gong, S. Zhang, J. Yang, D. Dai, and B. Schiele, “Bi-Level Alignment for Cross-Domain Crowd Counting,” in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7542–7550, 2022. doi:10.32628/CSEIT23903104.

[13] S. Bhardwaj, A. Dwivedi, A. Pandey, Y. Perwej, and P. R. Khan, “Machine Learning-Based Crowd behavior Analysis and Forecasting,” in *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 9, no. 3, pp. 418-429, 2023. doi:10.32628/CSEIT23903104

[14] Z. Zou, Y. Cheng, X. Qu, S. Ji, X. Guo, and P. Zhou, “Attend to Count: Crowd Counting with Adaptive Capacity Multi-Scale Cnns,” *Neurocomputing*, vol. 367, pp. 75–83, 2019. doi: 10.1016/j.neucom.2019.07.082.

[15] Y. Tao, Z. Ling, and I. Patras, “Universal Foreground Segmentation based on Deep Feature Fusion Network for Multi-Scene Videos,” *IEEE Access*, vol. 7, pp. 158326–158337, 2019. doi: 10.1109/ACCESS.2019.2952003.

[16] A. Hafeezallah, A. Al-Dhamari, and S. A. R. Abu-Bakar, “U-ASD Net: Supervised Crowd Counting based on Semantic Segmentation and Adaptive Scenario Discovery,” *IEEE Access*, vol. 9, pp. 127444–127459, 2021. doi: 10.1109/ACCESS.2021.3110632.

[17] E. B. Varghese and S. M. Thampi, “Towards The Cognitive and Psychological Perspectives of Crowd Behaviour: A Vision-Based Analysis,” *Connection Science*, vol. 33, no. 2, pp. 380–405, 2021. doi: 10.1080/09540091.2020.1770341.

[18] X. Jiang, L. Zhang, M. Xu, T. Zhang, P. Lv, B. Zhou, X. Yang, and Y. Pang, “Attention Scaling for Crowd Counting,” in *Proc. of The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4706–4715, 2020. doi:10.1109/CVPR42600.2020.00476.

[19] N. Ilyas, B. Lee, and K. Kim, “HADF-Crowd: A Hierarchical Attention-Based Dense Feature Extraction Network for Single-Image Crowd Counting,” *Sensors*, vol. 21, no. 10, pp. 3483, 2021. doi:10.3390/s21103483.

[20] Y. Zhu, K. Ni, X. Li, A. Zaman, X. Liu, and Y. Bai, “Artificial Intelligence Aided Crowd Analytics in Rail Transit Station,” *Transportation Research Record*, pp. 03611981231175156, 2023. SAGE Publications Sage CA: Los Angeles, CA. doi:10.1177/03611981231175156.

[21] R. Alharthi, A. Alhothali, B. Alzahrani, and S. Aldhaheri, “Massive Crowd Abnormal Behaviors Recognition using C3D,” in *Proc. of the 2023 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 01-06, 2023. doi:10.1109/ICCE56470.2023.10043437.

[22] A. Alia, M. Maree, M. Chraibi, A. Toma, and A. Seyfried, “A Cloud-Based Deep Learning Framework for Early Detection of Pushing at Crowded Event Entrances,” *IEEE Access*, vol. 11, pp. 45936-45949, 2023. doi:10.1109/ACCESS.2023.3273770.

[23] B. Ganga, B.T.Lata, Spoorthi. Admani, K R. Venugopal, and L. M. Patnaik, “Generation of High Quality Density Map using USkipGAN,” in *Proc. of the 2022 IEEE International Conference for Women in Innovation, Technology Entrepreneurship (ICWITE),* pp. 1-6, 2022. doi:10.1109/ICWITE57052.2022.10334424

[24] Ganga B, N Navya Shree, Dr. Lata B T, Dr. Venugopal K R, “AnomalyDetectNet: A Deep Learning Framework for Anomaly Detection in Video Data,” *IJCRT*, Vol 12, Issue 1, January 2024. ISSN: 2320-2882

AUTHORS

B. Ganga is a research scholar in the Department of Computer Science and Engineering at the University of Visvesvaraya College of Engineering (UVCE), Bangalore University, Bengaluru, India. She received B.E degree in Computer Science and Engineering from Karunya Institute of Technology, Tamilnadu, and an M.Tech in Software Engineering from M.S.Ramaiah Institute of Technology, Bangalore. She has 10 years of teaching experience. Her area of interest is Artificial intelligence, Deep learning, and Data Mining.

Dr. Lata B T is an Associate Professor in the Department of Computer Science and Engineering at the University of Visvesvaraya College of Engineering (UVCE), Bangalore University, Bengaluru, India. She obtained her B.E in Computer Science and Engineering from Karnataka University, Dharwad, and M.Tech degree in Computer Network Engineering from Visvesvaraya Technological University, Belgaum. Ph.D. degree in the area of Wireless Sensor Networks from Bangalore University. Her research interest is in the area of Sensor Networks, IoT, and Image processing.

K Rajshekar is a B.Tech student in the Department of Computer Science and Engineering at the University of Visvesvaraya College of Engineering (UVCE), Bengaluru, India. His areas of interest are machine learning, deep learning, and image processing.

Dr. Venugopal K R Former Vice-Chancellor, of Bangalore University, has served Bangalore University and UVCE for over the last five decades. He has 11 degrees with two Ph.Ds., one in Economics from Bangalore University, and another in Computer Science Engineering from IIT Madras. He received his ME degree in Computer Science Engineering from the Indian Institute of Science, Bangalore. He has authored and edited 84 books, he has published more than 1200 Research Papers, he has a Google Scholar citations H-index of 39 and holds 40 patents.. He received the IEEE Fellow and ACM distinguished educator award from USA for his outstanding contributions to the field of Computer Science Engineering

E-mail: gangamohanraj2003@gmail.com