Jurnal Ilmu Komputer dan Informasi (Journal of Computer Science and Information)
18/1 (2025), 67-76. DOI: http://dx.doi.org/10.21609/jiki.v18i1.1314

Estimating Passenger Density in Trains through Crowd Counting Modeling

Bryan Tjandra[1], Oey Joshua Jodrian[2], Nyoo Steven Christopher Handoko[3], Alfan Farizki Wicaksono[4]
Faculty of Computer Science, Universitas Indonesia
Depok, Indonesia
Email : [1]bryan.tjandra@ui.ac.id, [2]oey.joshua@ui.ac.id, [3]nyoo.steven@ui.ac.id, [4]alfan@cs.ui.ac.id

Abstract

The Greater Jakarta Commuter Rail, also known as the KRL Commuter Line, is one of the primary transportation choices for many people due to its comfort and efficiency. However, the level of user dissatisfaction is still relatively high, particularly regarding the frequent and unpredictable overcrowding of trains. To address this issue, our research develops an Artificial Intelligence-based model to predict train passenger density through crowd counting. By utilizing the proposed k-F1 metric By utilizing the proposed k-F1 metric, which balances the impact of False Positives and False Negatives in crowd density predictions by measuring the proximity of predicted points to the nearest ground truth within a scaled threshold and a constructed dataset of train density, we compare three object detection approaches: bounding box prediction (YOLOv5), density map (CSRNet), and proposal point (P2PNet). In our experiments, YOLOv5 surpassed other models in performance, achieving a Mean Absolute Error (MAE) of 1.41 and a k-F1 score of 0.91, while maintaining a fast inference speed of 300 milliseconds per frame. This model's strength lies in scenarios with fewer people and larger objects, such as passengers, within the frame. Conversely, P2PNet and CSRNet were less successful under these conditions, achieving MAEs of 3.49 and 4.98, and k-F1 scores of 0.77 and 0.35 respectively. However, it is important to note that P2PNet and CSRNet are better suited for denser and more congested environments, such as peak hours or at major transit hubs, where trains typically experience high crowd densities. The proposed density estimation method can be applied to real-time image-based CCTV systems to predict train congestion and facilitate transportation management decisions.

Keywords: Commuter Line, Density Estimation, Crowd Counting, P2PNet, Individual Localization

1. Introduction

Commuter train transportation (Kereta Rel Listrik/KRL) has become one of the most popular modes of transportation for many residents of the Jakarta Metropolitan Area. According to Statistics Indonesia, the non-departmental Indonesian government institute responsible for conducting statistical surveys, in February 2023 alone, there were over 20.8 million KRL passengers, with an average of 743 thousand passengers per day[1]. PT. Kereta Api Indonesia, the state-owned company that manages the KRL, predicts that within the next 5 years, the number of KRL passengers can reach 1.1 million per day[2]. Despite the increasing number of train users, the overall level of satisfaction between user expectations and reality in 2019 reached only 65.47% [1]. Based on our study (as seen in Section 4.1), from 2019 to June 2023, the level of user dissatisfaction, as observed through sentiment analysis of social media posts, reached 26.7%.

Based on our analysis of user satisfaction, we identified several main issues related to KRL. One of the critical issues is the frequent overcrowding of KRL trains. Despite the wide passenger-to-capacity ratio, reaching 63% as of 2023, KRL trains still experience overcrowding, especially during peak hours[2]. Real-time passenger density detection devices can be an alternative solution to avoid

[1]Source: Central Statistics Agency

[2]Source: PT. Kereta Api Indonesia

overcrowding. This allows for immediate response and management decisions to effectively control passenger flow and enhance safety [3]. Our conceptual framework hinges on establishing a nuanced relationship between passenger counting and density prediction, particularly in correlation with varying times of the day. This strategic linkage not only aids in optimizing train schedules but also facilitates the efficient management of passenger flow throughout the day. We have also conducted interviews with 10 KRL officers from various commuter lines, and they confirmed that no tool is currently available for predicting train density. This lack of prediction tools makes it difficult for passengers to prepare for and avoid overcrowded trains. It also poses a challenge for train operators to manage train schedules and balance the availability of trains with passenger demand.

In addressing the complexities of crowd counting and density estimation, there are three existing state-of-the-art approaches that are particularly relevant: density-based prediction, bounding box prediction, and proposal point detection [2]. CSRNet, which employs density map prediction, is renowned for its precise density estimation in dense crowds but struggles with the exact localization of individuals [3]. YOLOv5 uses bounding box prediction to achieve high-speed and accurate detection in less crowded environments, yet its performance diminishes in high-density settings due to overlap issues [4]. Lastly, P2PNet focuses on proposal point detection, providing fine-grained localization even in dense areas. However, it requires complex post-processing to resolve closely situated detections [2]. These models were selected to be the basis for our evaluation of crowd density and individual localization within the KRL commuter environment.

We propose developing a framework for estimating KRL passenger density to address these issues. Firstly, we conduct sentiment analysis and topic modeling on a collection of microblog Twitter posts to ascertain the importance of addressing density-related issues. Secondly, we develop a crowd-counting model and individual location detection model. In its development, we also constructed a dataset consisting of images directly captured from KRL, which is helpful for model evaluation. We propose a new metric to evaluate the model's ability to determine individual point localization. Lastly, we suggest a method to calculate the quantity of passenger density based on the results of the crowd-counting modeling.

[3]https://redresscompliance.com/instant-decision-making-ai-in-real-time-video-processing/

The success of this development is expected to enhance passenger density management and support the development of a more advanced and sustainable Indonesia. With the implementation of this density detection system, PT. KCI can directly observe the impact of changes in the KRL system on public interest. These effects can be observed through the trends in KRL usage density over time.

Contribution. This work contributes academically by developing a density estimation model for KRL, creating a new dataset for evaluating crowd-counting models in KRL, and introducing a novel metric for assessing the quality of the model in predicting individual point localization.

This research also holds several practical benefits, such as: (1) assisting PT. KCI in assessing the impact of changes in the KRL system through trends in KRL usage density over time, (2) improving efficient and sustainable transportation infrastructure, supporting the growth of the commuter train industry (in line with SDG 11[4]), (3) enhancing the performance of KRL transportation in line with the development of an advanced and sustainable Indonesia, and (4) supporting the development of a smart city with an automated KRL density monitoring center.

The methodology in this research paper has several limitations: (1) all crowd image datasets are captured from an overhead perspective of CCTV cameras, (2) the dataset used as training data originates from various locations in China, and (3) the tweet data collected is limited to those containing tags related to KRL in the past four years.

2. Background

This section reviews recent literature methods for density estimation (Section 2.1). These methods utilize crowd counting to estimate the density percentage. Several approaches, such as bounding box (Section 2.2), density map (Section 2.3), and proposal point (Section 2.4), are also discussed.

2.1. Density Estimation

Density estimation is a branch of computer vision used to estimate the density within an image. This method can utilize object detection-based crowd counting [5], where the results are masked to produce object segmentation and help estimate density [6]. Three approaches are used in the crowd counting methods in this research: bounding box, density map, and proposal point. Bounding box generates bounding boxes around detected objects [7].

[4]https://sdgs.un.org/goals/goal11

Proposal point predicts the location points at the center of objects [8]. The density map estimates the crowd counts by assigning density values to each grid cell in the image [3]. In the case of crowd density in public transportation, there are object characteristics such as a limited number of people up to 50 in a single image and variations in object sizes due to camera perspective. To determine the best method, the bounding box, density map, and proposal point methods are compared using the YOLOv5, CSRNet, and P2PNet models.

2.2. Bounding Box Method

One of the well-known state-of-the-art models for efficient and accurate bounding box computations is YOLOv5 [9].

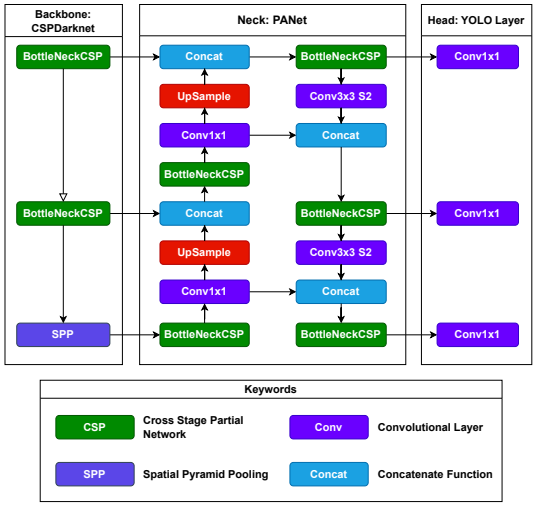

Figure 1. YOLOv5 Architecture

In Figure 1, YOLOv5 combines CSPNet with Darknet as the backbone to reduce the model size and to ensure better inference speed and accuracy [4]. YOLOv5 also applies PANet as the neck to enhance information flow. The head layer generates three feature maps, allowing the model to handle small, medium, and large objects [10]. However, the bounding box method can involve expensive annotation computations and can be challenging to precisely locate objects in distant, ambiguous, and complex images [11].

2.3. Density Map Method

One of the models for density map computation is CSRNet [3]. CSRNet utilizes dilated convolutional neural networks to extract deeper and retain output resolution while capturing critical information. Dilated convolution layers use sparse kernels to replace pooling layers to prevent resolution reduction and information loss.

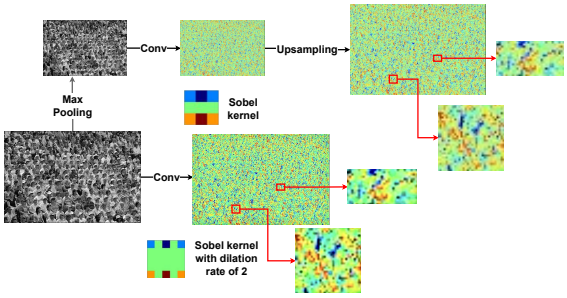

Figure 2. CSRNet Workflow

In Figure 2, the Front-end network extracts features from the image using dilated convolution, while the Back-end network maps these features to density values to predict the crowd density map [3]. Although density maps are suitable for estimating dense objects, their accuracy in determining exact object locations is limited [2].

2.4. Proposal Point Method

Density estimation can also be performed by predicting proposal points in crowd counting, as in the P2PNet method that utilizes localization techniques by directly predicting the coordinates of object points [2]. This method combines regression to detect locations and classification to determine the confidence score of key points. If the predicted point is close to the ground truth (GT) point, the model updates the confidence to be higher and the location closer to the GT.

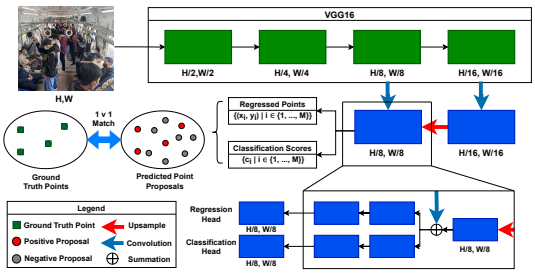

Figure 3. P2PNet Architecture

P2PNet uses the VGG-16_bn backbone to extract feature maps (Figure 3). These feature maps are divided into two branches for regression and classification. The points and confidence scores obtained from the predictions of these branches are connected

with the ground truth points. The resulting loss updates the weights in the next iteration, making the obtained point locations more accurate. Although the proposal point method can predict precise locations, it is susceptible to False Negatives in significant object size variations [2].

3. Methodology

This research consists of three main components, as shown in the scheme depicted in Figure 4. Section 3.2 aims to identify the most frequently discussed topics by the public regarding the KRL issue. Section 3.3 aims to compare the performance of three crowd-counting approaches and estimate individual locations. Lastly, Section 2.1 aims to estimate passenger density inside the KRL using the proposed estimation method by the research team. We also explain the dataset used (Section 3.1) and introduce the k-F1 evaluation method proposed in this research (Section 3.4).

3.1. Dataset

As shown in Table 1, the dataset consists of two image datasets and one text dataset. We scraped Twitter for 20K tweets related to KRL from 2019 to 2023. The crowd-counting training data images were obtained from ShanghaiTech educational institution [12] and train stations in China [13]. Additionally, we collected 150 captured images of passenger density in KRL trains under various conditions.

Table 1. Datasets used in the study

Dataset	Purpose	Size
Population density	Model training	3,323
KRL passenger density	Model evaluation	150
Twitter messages	Satisfaction analysis	20,000

3.2. Satisfaction Analysis

As a preliminary step, we conducted topic analysis on the sentiment-oriented messages in the relevant Twitter dataset of size 20K of the KRL context, as mentioned earlier. Manually labeling all the tweets would be time-consuming. Therefore, we applied a Semi-Supervised Learning approach with Pseudo-labeling [14]. Initially, we manually labeled the sentiment orientation (positive, negative, and irrelevant) of 400 tweets. Then, a BERT-based model [15, 16] was trained on these labeled 400 tweets and used to predict labels for the remaining 19,600 tweets.

Tweets with prediction confidence levels above 0.9 were merged with the initial annotated data for retraining purposes. After retraining the model and using it to relabel the 19.600 tweets, we selected tweets with negative sentiment labels (either from manual labeling or pseudo-labeling) for topic modeling to identify negative issues related to the use of KRL. Based on the generated topics, the main issue that emerged is the frequent and unpredictable overcrowding in KRL. We employ crowd counting and individual localization methods to estimate this overcrowding. As an additional note, we used BERTopic [17, 18] for topic extraction and processed KRL-related tweets using the API of GPT-3.5 [19, 20] and the tweet-preprocessor library[5].

3.3. Crowd Counting & Localization

As mentioned in Section 4.1, the analysis of the Twitter dataset suggests the need for a system that can predict passenger density in KRL. Two approaches can be utilized in the case of density estimation, which uses crowd counting in KRL. The first approach involves cameras above the doors, where the number of people is incremented when someone enters and decremented when someone exits. However, this approach has some drawbacks, such as the possibility of people moving between train carriages, resulting in variations in the number of passengers in each car. Additionally, this approach can accumulate errors if there are detection failures, leading to less accurate estimates of passenger count.

The second approach involves crowd counting using cameras inside the KRL to capture images of the entire train. With this approach, estimating train density can be done more effectively as the crowd images are visible. This approach also employs only head-based detection, as passengers' bodies may be occluded by other objects [21]. In this research, we focus on the second approach mentioned above.

We compare several methods that can be used to count crowds and localize the detected objects. It is important to note that counting the detected people alone is insufficient; accurately localizing the objects is also crucial. The image data undergoes preprocessing steps, including cropping, resizing, and image augmentation. Subsequently, these images are fine-tuned using three crowd detection approaches: (1) Proposal Points with P2PNet, (2) Density Map with CSRNet, and (3) Bounding Box with YOLOv5, as illustrated in Figure 5.

In order to ascertain the optimal object detection approach, we evaluate their performance using the

[5]https://pypi.org/project/tweet-preprocessor/

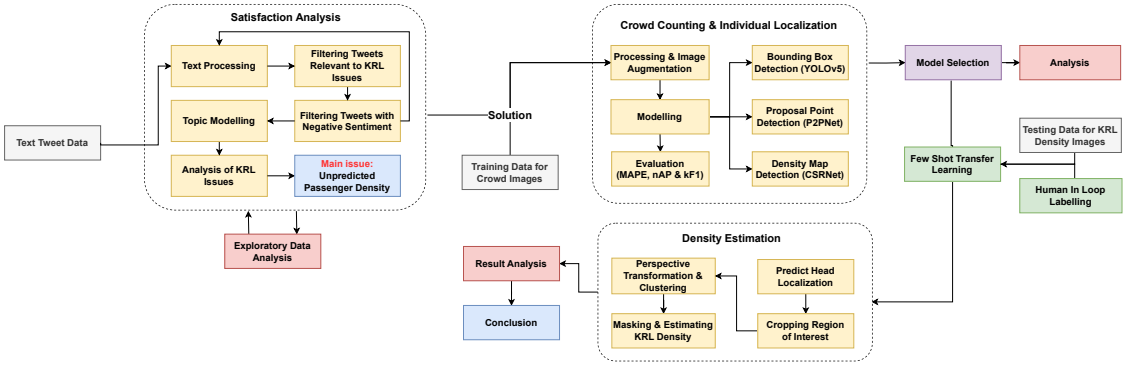

Figure 4. Experimental Flow Diagram Scheme

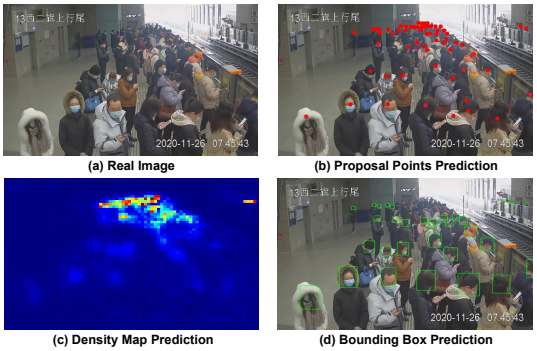

Figure 5. Comparison of prediction methods for crowd counting using the bounding box with YOLOv5, density map with CSRNet, and proposal point with P2PNet

Mean Absolute Percentage Error (MAPE) metric to estimate the error in the predicted number of people and the k-Nearest F1 Score (k-F1) metric, which we propose to measure the performance of point localization estimation. The use of MAPE as a representation of the average error percentage of objects is preferred over MAE because the number of people in a single image can vary. Therefore, measuring only the absolute error is insufficient, and it is better to represent the error as a percentage.

We use the midpoint to determine the precise location within the bounding box so that when density estimation is performed, the resulting mask has a circular shape that conforms to the shape of the head rather than a rectangular shape. As for the density map, we apply thresholding to determine the discrete predicted locations.

3.4. k-Nearest F1 Score

The evaluation of individual point localization often relies on the Normalized Average Precision (nAP) metric [2]. However, nAP can be overly sensitive to False Positives (FPs), potentially misrepresenting performance, particularly in scenarios with a high number of False Negatives (FNs). In such cases, nAP might yield seemingly acceptable results despite overlooking the significant presence of FNs.

Recognizing the need for a metric that balances both FP and FN in density estimation, we propose a new effective metric for evaluating point localization estimation to reduce the number of detections (FP and FN) in a balanced manner, called the k-Nearest F1 Score (k-F1). This metric considers the distance differences between the predicted points and the ground truth (GT) and utilizes the confidence score from the prediction results.

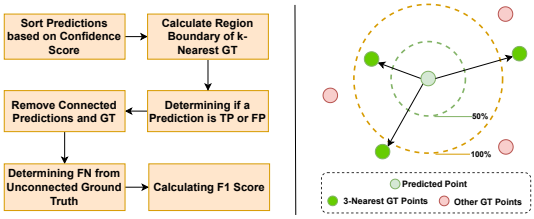

Figure 6. Calculation process of k-F1 Score (left) and illustration of region calculation based on 3-nearest GT (right) with restriction factor 0.5

Figure 6 on the left illustrates the working process of the k-F1 metric. First, the predicted set points $\mathcal{X}$ are sorted based on the confidence score. Second, for each predicted point $x \in \mathcal{X}$ in sequence, the k-nearest GT points, along with the average Euclidean distance to the predicted point x, denoted as $d_k(x)$, are found. This is illustrated in Figure 6 on the right. Finally, $d_k(x)$ is used as a threshold to determine whether $x \in \mathcal{X}$ is a True Positive (TP) or False Positive (FP), formally defined as:

$$I(x) = \begin{cases} 1 & \text{if } d_p(x) < \lambda . d_k(x) \\ 0 & \text{otherwise}\,, \end{cases}$$

where $I(x)$ is an indicator function that takes a value of 1 if the predicted point x is considered TP and 0 if it is considered FP; $d_p(x)$ is the Euclidean distance between the predicted point x and the nearest GT point; and λ is a restriction factor. This factor adjusts the threshold for classifying a prediction as a True Positive by scaling the average distance to the k-nearest ground truth points, thereby allowing for flexibility in how strict the detection criteria are based on the scenario's requirements.

If the Euclidean distance between the prediction and the nearest GT point $d_p(x)$ is smaller than the λ times $d_k(x)$, the prediction is considered similar to the GT point and classified as TP; otherwise, it is considered FP. If multiple GT points are within the restricted region, the GT point closest to the prediction is selected.

Once a prediction has been considered similar to a GT point, it is not considered again for subsequent k-nearest calculations. After evaluating predictions with the highest confidence, the process continues with the following prediction with the highest confidence, and so on. After all predictions have been evaluated, GT points that are not connected are considered False Negatives (FN). From the values of TP, FP, and FN, the F1 Score can be calculated, which is then referred to as the k-F1 Score.

3.5. Modeling Density Estimation

Several approaches can be used for density estimation. The first approach is to count the number of people inside the train and divide it by the maximum capacity of the train. However, this approach has limitations because the position of the CCTV cameras is limited to the ends of the train cars, so the images cannot accurately represent all the people inside the cars, especially those in the middle. This is also due to the limited height of the train carriages and the presence of plate structures that can obstruct the view of people far away.

Given the varying perspectives, areas closer to the camera tend to appear less crowded, while those farther away appear more crowded. We hypothesize, based on current observations, that density trends might differ significantly towards the back of the crowd regions; however, this remains a preliminary assumption pending further research. To mitigate this bias, we focus on the specific middle area when estimating the density and assume the density estimation value applies to a single train car. The density estimation is done by considering only the area covered by the head since using the entire body can introduce bias, as other objects may potentially occupy the space covered by the body from the camera's perspective.

We use few-shot transfer learning [22] by utilizing the three pre-trained crowd-counting models mentioned earlier due to small data examples and the density data of Indonesian trains collected by the research team. We begin by utilizing a pre-trained model that has already learned general features from a large dataset. Then, we fine-tune this model using a smaller dataset specific to our task. A total of 150 images are used and undergo a human-in-the-loop labeling process [23], where $1/3$ of the data is for transfer learning and the remainder for validation.

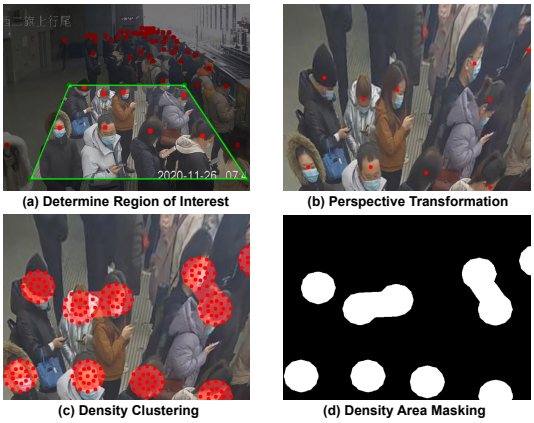

Figure 7. Process for density estimation

Figure 7 illustrates the general steps involved in estimating the density. First, we determine the region of interest (ROI) that is transformed to a top-down perspective using the Homography matrix [24] so that the camera perspective appears as if it were taken from above. This transformation ensures that the head sizes become uniform [25]. After the transformation, the predicted point size is adjusted to match the head size.

Next, we take the average radius of points that can cover the head area to obtain the constant used as the point scaling factor to cover the head from 50 observation images. Since the sizes of male and female heads generally have minor standard deviations, it can be assumed that the head sizes are relatively similar [26]. Therefore, scaling with a constant is still relevant. Subsequently, clustering is performed using the DBSCAN algorithm [27] to generate areas with no space between closely located heads, as objects can no longer occupy these spaces. To ensure that the points can adapt to the head size, 12 points with a radius of r pixels and 9 points with a radius of $r/2$ pixels are added, as shown in Figure 7 (c).

After obtaining the existing clusters, we use the concave hull algorithm to form the segmentation areas for each cluster [28]. Then, the density estimation is calculated by dividing the number of pixels

in the segmentation areas by the number of pixels in the region of interest. The number of pixels is obtained using the Shoelace Formula, which calculates the area of a polygon in the formed concave hull. Suppose this method is used for real-time density estimation through CCTV with video detection. In that case, the average percentage of density can be taken from 10 consecutive frames to avoid possible estimation errors.

4. Experimental Results and Analysis

Based on the results of the grouped topics from tweet texts, it is found that the main issue that frequently arises is the unpredicted overcrowding of KRL capacity (4.1). Therefore, we developed a method to estimate density based on crowd counting. Based on the results of the MAPE, nAP metrics, and the proposed k-F1 metric, it is found that P2PNet achieves the best performance on images with a large number of people (4.2). After crowd counting, we estimate the density using the percentage of the segmented area masked by the head. For the case of KRL, the YOLOv5 method achieves the best results due to the relatively large size of the objects.

4.1. Passenger Satisfaction

In the initial stage, we constructed the IndoBERT model to predict the relevance of KRL issues with a F1 validation score of 0.86. The predicted results were filtered to include only tweets relevant to KRL issues, resulting in 10,128 relevant tweets. Next, we utilized two BERT-based sentiment models, namely ayameRushia and w11wo. The results indicated that the ayameRushia model achieved an F1 score of 0.803, while the w11wo model achieved a score of 0.95. After fine-tuning, the w11wo model's F1 score improved to 0.965, as shown in Table 2.

Table 2. The performance results of sentiment prediction

Model	F1 score	Accuracy
ayameRushia/bert-base-indonesian-1.5G-sentiment-analysis-smsa	0.803	0.819
w11wo/indonesian-roberta-base-sentiment-classifier	0.950	0.960
Fine tuned w11wo/indonesian-roberta-base-sentiment-classifier	0.965	0.969

Out of the 10,128 tweets analyzed using the fine-tuned w11wo model, 26.7% of them were found to have negative sentiment. We conducted further analysis to identify the topics contributing to these negative sentiments. We successfully identified relevant topics by employing the BERTopic model, as displayed in Table 3.

Table 3. Top five most discussed KRL issues

Topic	Keyword	Percentage
Overcrowding of KRL	*krl_padet*	19.2%
Delayed arrivals	*krl_datang_lambat*	13.9%
Cases of sexual harassment	*seksual_pelecehan*	9.1%
Prone to virus spread	*virus_covid*	7.9%
Misuse of priorities	*duduk_prioritas*	6.5%

Based on the obtained results, the most frequently mentioned topic in the tweets is the overcrowding of KRL capacity. This forms the basis for developing a model for estimating KRL overcrowding density.

4.2. Crowd Counting Effectiveness

In order to calculate the density of passengers, we applied the methods of crowd counting and individual localization by evaluating three distinct approaches on two types of ShanghaiTech datasets: (1) Dataset A, with a maximum of $2,000$ individuals per frame, and (2) Dataset B, with a maximum of 500 individuals per frame.

Table 4. Comparison of results for Dataset A and B

Method	Dataset	MAPE	nAP	k-F1	Time (s)
Yolov5	A	0.58	0.65	0.64	0.307
	B	0.21	0.87	0.85	0.204
CSRNet	A	0.30	0.36	0.34	0.332
	B	0.17	0.40	0.37	0.221
P2PNet	A	0.15	0.92	0.90	0.408
	B	0.16	0.94	0.91	0.276

Table 4 presents the evaluation results for Dataset A and Dataset B, utilizing the GPU T4 for running the experiment. From these results, it can be observed that the P2PNet method achieves the best MAPE, (nAP), and k-F1 ($k = 3$, $\lambda = 0.75$). Here, k represents the number of nearest Ground Truth (GT) points considered for calculating the average as a limit for a point to be counted as True Positive (TP). λ denotes the restriction factor multiplied by the radius of a region to narrow down the threshold for a prediction to be categorized as TP. We set k as the three nearest GT points and λ as 0.75 to reduce False Positives (FP). Note that the higher the value

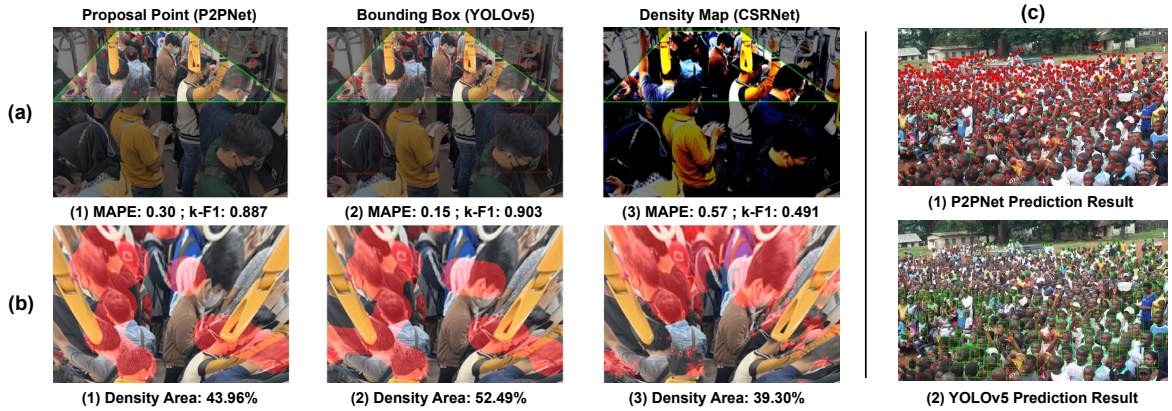

Figure 8. (a) Crowd Counting and (b) Density Estimation Results, and (c) Example Comparison Results of YOLOv5 and P2PNet Predictions on Dense Objects

of k, the more precise the evaluation results as more nearest points are considered. However, the higher the value of k, the greater the computational time. Therefore, we choose k = 3 to balance the trade-off between these two factors.

The CSRNet model exhibits a slightly worse MAPE than P2PNet due to its regression-based prediction results, which estimate the count of detected objects. However, it outperforms YOLOv5 because the regression method employed by CSRNet is more suitable for densely populated objects. Furthermore, CSRNet demonstrates lower performance in nAP and k-F1 scores compared to P2PNet and YOLOv5. We suspect that the localization capability of CSRNet is compromised due to high noise levels during thresholding. On the other hand, the YOLOv5 model exhibits the worst MAPE performance. However, it has the fastest inference time among the models.

The comparison of discrete object detection between YOLOv5 and P2PNet can be seen in Figure 8 (c), where it is evident that YOLOv5 still has a significant number of False Negatives (FNs), particularly undetected heads. Table 4 presents the differences in results between the k-F1 score and nAP. The nAP metric focuses more on penalizing False Positives (FPs), while the k-F1 score emphasizes penalties on both FPs and FNs. This penalty is because detection errors can lead to inaccurate density estimation.

Table 5. Comparison of results for the KRL datasetL

Method	MAE	MAPE	k-F1
Yolov5	1.41	0.129	0.91
CSRNet	4.98	0.927	0.35
P2PNet	3.49	0.309	0.77

Furthermore, we evaluated all three models using the KRL density dataset. We employed the few-shot transfer learning method, involving 50 KRL images for training and 100 for validation. The prediction results and evaluation metrics can be observed in Figure 8 (a) and Table 5. The evaluation results show that YOLOv5 performs the best in terms of MAE, MAPE, and k-F1 scores. YOLOv5 excels in accurately detecting relatively large objects, such as those found in KRL, but not excessively numerous.

As the camera angle in the train car remains static, it ensures consistency in data capture. However, variations in camera resolution and technical aspects could potentially affect these outcomes. On the other hand, P2PNet exhibits inferior performance in detecting larger objects but performs well in densely populated areas, as demonstrated by the crowded end of the KRL. CSRNet, while better at estimating densely packed objects, struggles with accurate localization, resulting in a lower k-F1 score.

4.3. Density Estimation

The three models are then evaluated to determine the best model for density estimation. Based on the results presented in Table 5, it is evident that the YOLOv5 model provides the best performance. We also showcase the density estimation results in Figure 8 (b) to support this finding. It can be observed that P2PNet still has uncovered head areas, CSRNet exhibits some noise, resulting in inaccurate density estimation, while YOLOv5 adequately covers the areas. Based on these findings, it can be concluded that the bounding box method (YOLOv5) is the best approach for density estimation in the context of KRL compared to other methods.

We classify the density percentage into low, moderate, and crowded categories. We conducted interviews with 30 respondents using 50 reference images. The interview results were then calculated to obtain a 90% confidence interval, and the outcomes were adjusted accordingly, as shown in Table 6.

We classified the crowd density into three categories—low, moderate, and crowded—to simplify the model's application in future app integration. We interviewed 30 respondents with 50 reference images, asking them to categorize each image according to these density levels. Their assessments were then statistically analyzed to establish a 90% confidence interval for each category, the results of which are adjusted and displayed in Table 6.

Table 6. The range of crowd density categories in KRL

Result	Low		Moderate		Crowded	
Interview	21.2% 39.7%	–	45.3% 66.4%	–	71.2% 86.8%	–
Adjusted	0% 42.0%	–	42.0% 68.8%	–	68.8% 100%	–

After obtaining the interview results, we calculated the average interval limits for each category. The adjusted outcomes can also be found in Table 6. Examples of density estimation results for moderate and crowded-density KRL conditions are shown in Figure 9.

Figure 9. Categories and percentages of crowd density in KRL

Based on the results obtained, this method can be further implemented with CCTV surveillance cameras to monitor the density levels inside KRL.

5. Conclusion

This work investigated the analysis of sentiment orientation and topics related to transportation issues. We achieved an F1 validation score of 0.965 using a fine-tuned `w11wo`. Analyzing a dataset of 10,128 relevant tweets, we found that 26.7% expressed negative sentiment. Among these negative tweets, our topic modeling revealed that the most discussed issue is KRL overcrowding, accounting for 19.2%. We then explored the application of density estimation approaches using different crowd-counting methods. The implemented methods achieved promising results, demonstrating their effectiveness in crowd density estimation through various crowd-counting techniques.

Our experiments revealed trade-offs between the three compared methods: YOLOv5, CSRNet, and P2PNet. CSRNet excelled in crowd counting for dense images due to its density map approach. However, its inability to precisely localize individuals introduced significant noise into the estimations. On the other hand, P2PNet, utilizing the proposal point method, effectively handled both crowd counting and individual localization in dense scenarios with small objects. However, it struggled with detecting larger-sized objects. CSRNet and P2PNet achieved MAEs of 4.98 and 3.49 and k-F1 scores of 0.35 and 0.77 respectively. YOLOv5, on the other hand, emerged as a balanced choice, achieving good performance in crowd counting and individual localization for our specific scenario involving fewer people and larger objects. It achieved the best performance of the three models, with an MAE of 1.41 and a k-F1 score of 0.91.

In the context of train passenger density, characterized by a relatively small number of individuals but with larger sizes, the YOLOv5 method performs well in crowd counting and individual localization for medium to large-sized objects. However, YOLOv5 may produce false negatives for small-sized objects. By utilizing the density estimation method based on YOLOv5 in CCTV systems, passengers can gain valuable insights into real-time crowd density, facilitating informed decisions about boarding and waiting times. Furthermore, we highlight the `k-F1` metric's effectiveness in evaluating individual point localization accuracy. Overall, the findings presented here hold promise for improving passenger experience and informing future transportation management strategies.

References

[1] D. I. Nurcahya, "Tingkat kepuasan pengguna terhadap pelayanan krl commuter line rute bogor-jakarta kota," Ph.D. dissertation, 2019.

[2] Q. Song, C. Wang, Z. Jiang, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Wu, "Rethinking counting and localization in crowds:a purely point-based framework," 2021.

[3] Y. Li, X. Zhang, and D. Chen, "Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes," 2018.

[4] M. Horvat, L. Jelečević, and G. Gledec, "A comparative study of yolov5 models perfor-

mance for image localization and classification," 09 2022.

[5] G. Gao, J. Gao, Q. Liu, Q. Wang, and Y. Wang, "Cnn-based density estimation and crowd counting: A survey," 2020.

[6] X. Zhang, Y. Sun, Q. Li, X. Li, and X. Shi, "Crowd density estimation and mapping method based on surveillance video and gis," *ISPRS International Journal of Geo-Information*, vol. 12, no. 2, p. 56, 2023.

[7] N. Nufus, D. M. Ariffin, and A. S. Satyawan, "Sistem pendeteksi pejalan kaki di lingkungan terbatas berbasis ssd mobilenet v2 dengan menggunakan gambar 360° ternormalisasi," *Prosiding SENASTINDO*, vol. 3, pp. 123–134, December 2021.

[8] H. Chen and H. Zheng, "Object detection based on center point proposals," *Electronics*, vol. 9, p. 2075, 12 2020.

[9] K. Sharma, S. Rawat, D. Parashar, S. Sharma, S. Roy, and S. Sahoo, "State of-the-art analysis of multiple object detection techniques using deep learning," *International Journal of Advanced Computer Science and Applications*, vol. 14, pp. 527–534, 07 2023.

[10] R. Xu, J. Li, and Y. Liu, "An improved forest fire and smoke detection model based on yolov5," *Forests*, vol. 14, p. 833, April 2023.

[11] X.-T. Vo and K.-H. Jo, "Accurate bounding box prediction for single-shot object detection," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 9, pp. 5961–5971, 2022.

[12] W. Liu, D. L. W. Luo, and S. Gao, "Future frame prediction for anomaly detection – a new baseline," in *2018 IEEE CVPR*, 2018.

[13] J. Yang and M. Gong, "Metrostation dataset," 2022. [Online]. Available: https://doi.org/10.6084/m9.figshare.20521848.v1

[14] D.-H. Lee, "Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks," *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 07 2013.

[15] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "Indolem and indobert: A benchmark dataset and pre-trained language model for indonesian nlp," 2020.

[16] K. S. Nugroho, A. Y. Sukmadewa, H. Wuswilahaken DW, F. A. Bachtiar, and N. Yudistira, "Bert fine-tuning for sentiment analysis on indonesian mobile apps reviews," in *Proceedings of the 6th International Conference on Sustainable Information Engineering and Technology*, ser. SIET '21, 2021, p. 258–264. [Online]. Available: https://doi.org/10.1145/3479645.3479679

[17] M. Grootendorst, "Bertopic: Neural topic modeling with a class-based tf-idf procedure," *arXiv preprint arXiv:2203.05794*, 2022.

[18] S. Sawant, J. Yu, K. Pandya, C.-K. Ngan, and R. Bardeli, "An enhanced bertopic framework and algorithm for improving topic coherence and diversity," 2022, pp. 2251–2257.

[19] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[20] J. Ye, X. Chen, N. Xu, C. Zu, Z. Shao, S. Liu, Y. Cui, Z. Zhou, C. Gong, Y. Shen, J. Zhou, S. Chen, T. Gui, Q. Zhang, and X. Huang, "A comprehensive capability analysis of gpt-3 and gpt-3.5 series models," *ArXiv*, vol. abs/2303.10420, 2023.

[21] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[22] A. Parnami and M. Lee, "Learning from few examples: A summary of approaches to few-shot learning," 2022.

[23] E. Mosqueira-Rey, E. Hernández-Pereira, D. Alonso-Ríos, and et al., "Human-in-the-loop machine learning: a state of the art," *Artificial Intelligence Review*, vol. 56, pp. 3005–3054, 2023. [Online]. Available: https://doi.org/10.1007/s10462-022-10246-w

[24] T. Sung and H. J. Lee, "Images alignment using homography transformation matrix," 10 2018.

[25] S. Basalamah, S. Khan, and H. Ullah, "Scale driven convolutional neural network model for people counting and localization in crowd scenes," *IEEE Access*, pp. 1–1, 05 2019.

[26] A. Ormeci, H. Gürbüz, A. Ayata, and H. Cetin, "Adult head circumferences and centiles," *Journal of Turgut Ozal Medical Center*, vol. 3, pp. 261–264, 01 1997.

[27] D. Deng, "Dbscan clustering algorithm based on density," 2020, pp. 949–953.

[28] A. Moreira and M. Santos, "Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points." 01 2007, pp. 61–68.