

RESEARCH ARTICLE

Training a Regression-Based Model for Crowd Counting in Transit Cars Using Ranked Image Pairs and Triplets

HOJUN LEE¹, KYEONGJUN LEE², JIWON KANG^{ID2}, AND KEEMIN SOHN^{ID1,2}¹Department of Smart City, Chung-Ang University, Seoul 06974, South Korea²Department of Urban Engineering, Chung-Ang University, Seoul 06974, South Korea

Corresponding author: Keemin Sohn (kmsohn@cau.ac.kr)

This work was supported in part by the Chung-Ang University Research Scholarship Grant, in 2022; and in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government under Grant 2021R1A2C2003842.

ABSTRACT Accurately measuring the level of crowding in transit cars is crucial for ensuring passenger safety and efficient operation. However, applying object detection algorithms to crowd counting in transit cars poses difficulties due to the low viewpoint of the cameras and the labor-intensive task of image labeling. Although some researchers have explored regression-based crowd counting methods without labeling with bounding boxes, their approaches still necessitate manual counting of passengers for image labeling. To overcome these challenges, we propose a novel calibration method for regression-based models that minimizes the number of labeled images required for training. Our approach employs image pairs and triplets with ranks for reinforcing the model training. Subsequently, the training task requires a minimal number of images labeled with exact passenger counts. Experimental results demonstrate that our proposed calibration approach considerably enhances the crowd counting performance of the conventional regression-based model. Specifically, our method reduces the mean absolute error (MAE) by 76.5% and 34.3% for conventional detection- and regression-based calibration methods, respectively.

INDEX TERMS Passenger load in transit, crowd counting, computer vision, regression-based model, ranking model.

I. INTRODUCTION

Measuring passenger load in a transit car is an important aspect of enhancing safety and optimizing operation of public transit systems. Passenger load data can also be used to manage crowding to provide better passenger service and are inevitable to plan for future infrastructure improvements. Transit operators estimate revenue, and plan budgets according to the passenger load data. This information can also be used to justify funding requests for future improvements or expansions to the transit system.

There are several methods that can be used to measure the passenger load in a transit car. Automatic fare collecting

The associate editor coordinating the review of this manuscript and approving it for publication was Yongjie Li.

(AFC) systems use sensors or other devices to charge transit fares and detect when passengers enter or exit the transit station. Using this fare-charging information, the passenger load data can be automatically collected on a real-time basis. For example, radio Frequency Identification (RFID) technology uses radio waves to identify and track passengers with RFID-enabled cards or mobile devices [1], [2]. This technology can track the number of passengers boarding and alighting at each station, allowing for real-time passenger load information. Some transit systems have sensors that measure the weight of transit cars [3]. The passenger load can be estimated by the difference between loaded and empty cars, but the challenge is that the estimation might be affected by many factors and leads to an inaccurate measurement. The computer vision technology has recently been highlighted leveraging the recent advancements in deep learning. This

involves analyzing video footage from cameras mounted inside the transit car to determine the number of passengers on board.

Each passenger crowd counting method has its own advantages and disadvantages, and the most appropriate method will depend on the specific needs and requirements of the transit operator. However, the trend is towards computer vision-based counting systems, as they become more accurate, efficient and cost-effective. There are two typical types of computer vision technologies for passenger counting such as the detection- and regression-based passenger counting methods.

The detection-based passenger counting typically involves two steps. The first step is to use a computer vision algorithm to detect individual passengers. This can be done using various deep learning-based object detection models like YOLO [4] and EfficientDet [5]. Once the individual passengers have been detected, the next step is to simply count the number of detected passengers. The detection-based approach, however, has a serious disadvantage that the accuracy deteriorates as crowding level increases.

The regression-based passenger counting could be a more plausible approach whereby only passenger counts are required to label images [6]. Although training regression-based crowd counting models does not require bounding boxes, the models still necessitate manual endeavor to count passengers for labeling images. Furthermore, regardless of model specifications, applying the regression-based model to crowd counting in transit cars poses difficulties due to the problem of varying object scales according to positions within an image. Recently, the density map enhanced the counting accuracy of regression-based crowd counting models, which adjust the scale by passenger positions within an image [7], [8]. Preparing the ground truth density maps for training, however, entails the extra burden.

Reducing the human effort to annotate training images is the most critical challenge when working with a computer vision technology. We propose a novel calibration method that minimizes the human effort to annotate images for training. Our approach uses image pairs and triplets with ranks for training the crowd counting algorithm. Subsequently, the approach demands a minimal number of images labeled with exact passenger counts. Ranking crowding levels of paired or tripled images significantly takes much less time than that for positioning or delineating individual passengers in each image. By effectively leveraging regression-based techniques, our proposed training method offers improved accuracy in crowd counting for a transit car.

The present paper is organized as follows. The next section introduces the research history of crowd counting based on the computer vision technology. The third section set up two models and describes how to calibrate them using a minimal number of images labeled with exact counts. The fourth section describes how to obtain and annotate video images for training. The fifth section shows the test results and compare them with those from other passenger counting

algorithms. The last section draws conclusions and suggests further extensions of the proposed crowd counting approach.

II. LITERATURE SURVEY

Following the significant advancements in deep learning applied to computer vision technology, the prevailing paradigm in crowd counting has shifted towards the use of Convolutional Neural Networks (CNNs). Contemporary “state-of-the-art” methodologies for crowd counting predominantly rely on density maps [7], [8], [9] which modulate object density to accommodate differing perspectives. These CNN-based methodologies can be classified into several categories based on various criteria.

Certain studies have leveraged patch-based inputs for crowd counting via CNNs [8], [10], while others have utilized entire images as input [11], [12]. The architecture of the employed network also differs across studies. Single column models have been utilized in several works [13], [14], while multi-column models have been employed in others [15], [16]. Though the majority of researchers have opted for a fully supervised approach for model training, some have utilized self-supervised [17] or semi-supervised methodologies [18]. Furthermore, there are instances where an unsupervised training scheme has been adopted [19]. The training scheme devised in our research aligns more closely with these latter methodologies, being less dependent on intensive supervision.

Besides CNN-based models, attention-based models were developed for crowd counting. A weakly-supervised transformer was used for crowd counting with adaptive scene consistency attention [20]. A multi-scale attention network was devised for crowd counting [21]. An attention-based crowd counting model was developed to alleviate the problem of uneven distribution of crowd density [22].

The object detection in satellite images has the same problem that individual objects cannot be detected using the conventional object detectors. A semantic segmentation-guided method was mobilized to detect objectives in remote sensing images [23]. Some researchers used pseudo instance soft labels for weakly supervised object detection in remote sensing images [24]. They also suggested various technologies such as the unbiased proposal filtration and the oriented bounding box regression to enhance the object detection in remote sensing images [25], [26].

The regression-based approach has also employed a CNN-based backbone for feature extraction from images [14], [27], [28]. Some researchers have concentrated on directly assessing passenger load using images obtained by cameras mounted within transit cars [8]. Conversely, an alternative strategy involves counting the number of boarding and alighting passengers using images captured by cameras focusing on gates of transit cars [28]. The latter methodology offers an advantage in terms of bolstering detection accuracy, albeit with the necessity to compute passenger load indirectly from gate counts. Computing the current

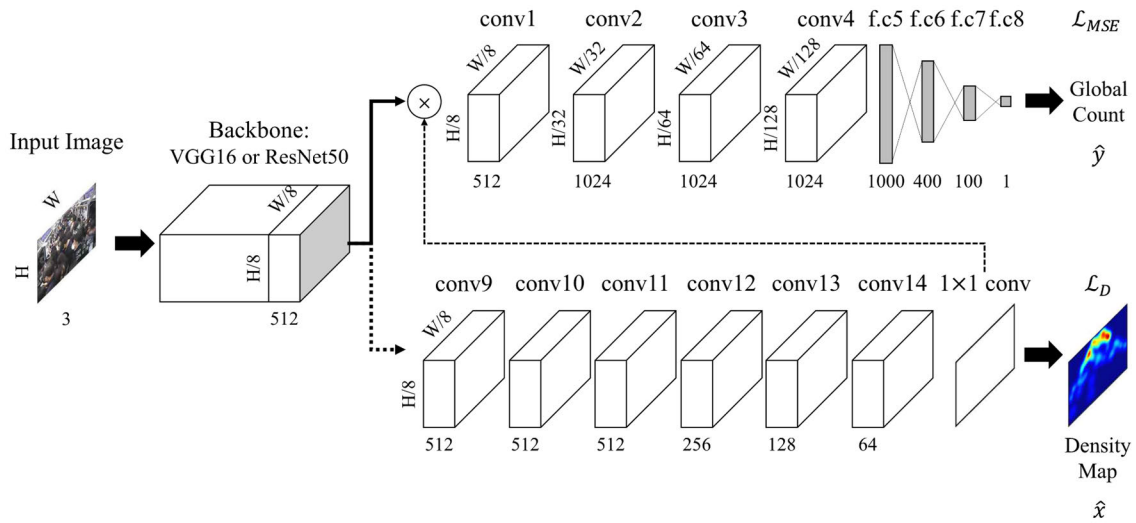


FIGURE 1. Architecture of the proposed regression-based crowd counting model for transit cars.

crowding level, thus, depends on the previous crowd level. This presents a challenge in situations where the detection mechanism malfunctions, as restoring the passenger count under such circumstances is difficult. In the present study, we have opted for the former approach, devising a novel training scheme designed to reduce the human effort to label images and to enhance the accuracy of direct passenger load measurement.

III. MODELING FRAMEWORK

A. MODEL SET-UPS

We implemented two distinct types of regression-based crowd counting models. The first model, referred to as the simple regression-based model, estimates passenger numbers directly from the features abstracted by a CNN from an image. This model requires only passenger counts to be a label for training. The second model, the augmented regression-based model, produces dual outputs: a density map and a passenger count. Given the variable scale in an image taken from a camera onboard, and the low viewpoint leading to numerous occlusions, the augmented model applies unique scales to each position within the image. However, this necessitates additional effort in preparing ground truth density maps for training.

We evaluated how much the performance of both models can enhance when using paired and tripled images with ranks. For comparison, we utilized EfficientDet [5] and YOLO5 [4], “state-of-the-art” object detection models, as a baseline, which were fine-tuned using the same training images with complete bounding box labels.

Both regression-based models utilize two different backbones, VGG-16 and ResNet50, respectively, for abstracting features from an image. The difference between the two models lies in the configuration of their respective head components, which share backbones. Specifically, the simple

regression-based model’s head directly outputs the passenger count within an input image. In contrast, the augmented regression-based model incorporates an additional auxiliary output for the density map. Thus, one head returns the passenger count while the other provides a density map, aiding in the fitting of passenger counts by considering differing scales for each position according to the camera perspective. The architecture of the head designated for fitting the density map is derived from a previous study that successfully executed crowd counting [14]. A notable feature of our augmented model structure is the utilization of the estimated density map to feed the main head for passenger counting. This arrangement allows for simultaneous training of both pipelines, which stands in contrast to the alternate training of the two heads as conducted in a previous study [8]. The architecture of both models is illustrated in Fig. 1.

B. RANKING DATA TO CALIBRATE THE REGRESSION-BASED MODEL

Ranking data is composed of sequence of ranks chosen by respondents and have widely been used to calibrate choice models in academia [29]. The present study used the ranking of crowdedness amongst images of passengers in a transit car. Given three different images (A, B and C), a respondent makes ranks in order of the passenger crowd level. In this case, the rank ACB would be converted to the respondent’s perception that A is more crowded than C and B, and C is also more crowded than B. If the passenger count that the respondent perceives for each image is assumed implicit score, the ranking observation can be interpreted as two statistically independent choices. If expanding this concept into the general case of N images, $N - 1$ independent choices are represented as follows [30]. Y_i denotes the implicit count of the i^{th} image and is assumed to be separated by the systematic part (\hat{y}_i) and the random part (ϵ_i). The former can be modeled

by a neural network, and the latter is assumed to follow the Gumbel distribution, which leads to the multinomial logit choice model.

$$(Y_1 \geq Y_n, n = 1, 2, \dots, N); (Y_2 \geq Y_n, n = 2, 3, \dots, N); \dots; (Y_{N-1} \geq Y_N) \quad (1)$$

The probability of observing a ranking sequence can be decomposed using the Bayes' theorem.

$$\begin{aligned} & \text{Prob}(r_1, r_2, r_3, \dots, r_N) \\ &= \text{Prob}(r_1|r_2, r_3, \dots, r_N) \text{Prob}(r_2, r_3, \dots, r_N) \end{aligned} \quad (2)$$

In Eq.(2), $\text{Prob}(r_1, r_2, r_3, \dots, r_N)$ denotes the joint probability of observing that the ranking indicates that r_1 is preferred to r_2 , r_2 is preferred to r_3 and so on. The joint probability is decomposed into the conditional probability $[\text{Prob}(r_1|r_2, r_3, \dots, r_N)]$ and the marginal probability $[\text{Prob}(r_2, r_3, \dots, r_N)]$. The marginal probability is recursively decomposed in the same manner and so on, as shown in the following formula [31].

$$\begin{aligned} & \text{Prob}(r_1, r_2, r_3, \dots, r_N) \\ &= \text{Prob}(r_1|r_2, r_3, \dots, r_N) \text{Prob}(r_2|r_3, r_4, \dots, r_N) \\ & \dots \text{Prob}(r_{N-1}|r_N) \end{aligned} \quad (3)$$

Finally, a multinomial logit choice model can then be used to denote each conditional probability assuming that the random part of count is distributed by the Gumbel distribution. For the case where a respondent chooses a more crowded image between a given pair of images, the probability reduces to a simple binary logit choice probability.

C. CALIBRATING THE SIMPLE REGRESSION-BASED MODEL

Three distinct loss functions are established to train the simple regression-based model. The first loss function reflects the disparity between observed and estimated passenger counts. It is denoted by the squared L2 loss (MSE) in Eq. (4), where \mathcal{A} represents the set of training images, $N_{\mathcal{A}}$ denotes the set size, y_i corresponds to the number of passengers within the i^{th} training image, and \hat{y}_i represents the estimated count.

$$\mathcal{L}_{MSE} = \frac{1}{N_{\mathcal{A}}} \sum_{i \in \mathcal{A}} (y_i - \hat{y}_i)^2 \quad (4)$$

Eq. (5) outlines the second loss function that takes the form of a binary logit choice model, representing the log-likelihood that chooses a more crowded image out of two. More intuitively, the loss function indicates that probability that the estimated count (\hat{y}_{i_1}) of a more crowded image exceeds that (\hat{y}_{i_2}) of a less crowded image. Here, the estimated passenger counts are regarded as random utilities for a binary logit choice model. The model output, \hat{y}_{i_1} , is derived from an input image with more passengers, and \hat{y}_{i_2} is the output from the other image with fewer passengers. \mathcal{P} is the set of training image pairs, $N_{\mathcal{P}}$ is the set size, equivalent to the number of image pairs in \mathcal{P} , and each pair in the set is ordered such that the first image in each pair contains more passengers. The

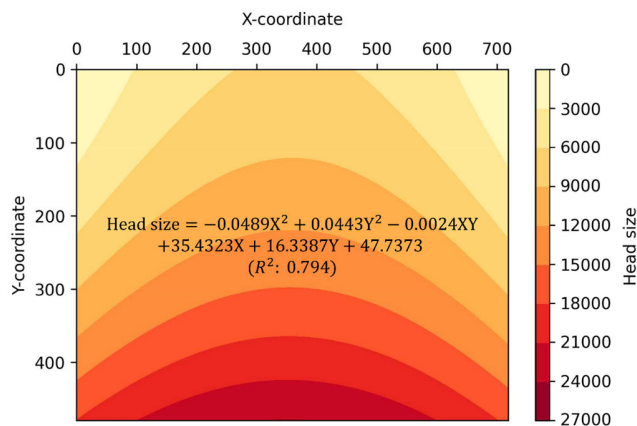


FIGURE 2. Perspective map derived from a linear regression analysis.

negative sign signifies the switch of the log-likelihood to the loss function.

$$\mathcal{L}_{pair} = -\frac{1}{N_{\mathcal{P}}} \sum_{i \in \mathcal{P}} \ln \frac{1}{1 + e^{\hat{y}_{i_2} - \hat{y}_{i_1}}} \quad (5)$$

The likelihood for observing the rank among an image triplet can be expressed by an ordered joint probability. This probability can be decomposed into the product of two conditional probabilities as shown in Eq. (6). For this formula, the multinomial logit choice model is applied to each conditional choice probability, with the passenger count output regarded as the systematic part. The first conditional term represents the probability that the most crowded image is chosen, and the second term denotes the conditional probability that the second most crowded image is chosen out of the remaining images. For the formulation, images in a triplet should be arranged in the decreasing order of crowdedness level.

$$\begin{aligned} P(i_1, i_2, i_3) &= P(i_1|i_1, i_2, i_3) P(i_2|i_2, i_3) \\ &= \frac{e^{\hat{y}_{i_1}}}{e^{\hat{y}_{i_1}} + e^{\hat{y}_{i_2}} + e^{\hat{y}_{i_3}}} \times \frac{e^{\hat{y}_{i_2}}}{e^{\hat{y}_{i_2}} + e^{\hat{y}_{i_3}}} \end{aligned} \quad (6)$$

Finally, Eq. (7) describes the third loss function, representing the negative log-likelihood that estimated passenger counts ($\hat{y}_{i_1}, \hat{y}_{i_2}, \hat{y}_{i_3}$) adhere to the ranks within image triplets. For facilitate the computation, three images in each triplet are arranged in decreasing order of crowdedness. \mathcal{T} denotes the set of image triplets for training, and $N_{\mathcal{T}}$ represents the set size.

$$\begin{aligned} \mathcal{L}_{triplet} &= -\frac{1}{N_{\mathcal{T}}} \sum_{i \in \mathcal{T}} \left[e^{\hat{y}_{i_1}} + e^{\hat{y}_{i_2}} - \ln \left(e^{\hat{y}_{i_1}} + e^{\hat{y}_{i_2}} + e^{\hat{y}_{i_3}} \right) \right. \\ & \quad \left. - \ln \left(e^{\hat{y}_{i_2}} + e^{\hat{y}_{i_3}} \right) \right] \end{aligned} \quad (7)$$

Two distinct approaches were devised to incorporate paired and tripled image data during the training of the simple regression-based passenger counting model. The first approach integrates the three loss functions using weights and minimizes the combined loss function holistically. Eq.

(8) depicts the combined loss function, where λ_c 's represent weights for individual losses.

$$\mathcal{L}_{com_sim} = \mathcal{L}_{MSE} + \lambda_{pair}\mathcal{L}_{pair} + \lambda_{triplet}\mathcal{L}_{triplet} \quad (8)$$

The second approach employs a stepwise training process. The initial step pretrains the model using paired and tripled images with ranks. Subsequently, the pretrained model is fine-tuned using images labeled fully with passenger counts. In other words, the model parameters are pretrained such that \mathcal{L}_{pair} and $\mathcal{L}_{triplet}$ are minimized, and \mathcal{L}_{MSE} is minimized in the subsequent step after initializing parameters with those pretrained in the previous step.

D. CALIBRATING THE AUGMENTED REGRESSION-BASED MODEL

The augmented regression-based model requires a perspective map to construct the ground truth density map for each training image. A perspective map signifies the relative scale of each pixel within an input image, thereby reflecting the varying number of pixels occupied by a human head for different positions in the image.

More concretely, the perspective map is necessary as a reference to reflect the different scale of human heads according to different position in an image when creating the ground-truth density map for each image. Once the perspective map is prepared, the density of human heads at a position in a density map can be determined in terms of the scale (=bandwidth) at the same position in the perspective map. That is, a smaller bandwidth is used to compute the density for positions far from the camera, whereas a larger bandwidth is used to reflect the density for positions near the camera. It should be noted that the density map adjusts the object concentration within an image rather than for the actual space in a transit car. The density map points to the concentration of human heads for unit area of image.

To produce the perspective map in advance, we implemented a polynomial regression analysis to identify the scales for each position within an input image. The size of a human head, quantified in pixels, was designated as the dependent variable, with pixel coordinates established as independent variables. Each pixel value within the perspective map is set as the estimated head size. The square root of this value determines the bandwidth of a Gaussian kernel, which is used to generate the ground truth density map for each training image.

Fig. 2 presents the estimated perspective map drawn from the result of regression analysis. Fig.3 provides examples of ground truth density maps for various input images. The passenger density for distant pixels is calculated using smaller bandwidths, and for closer pixels with larger bandwidths, enabling an adjustment for the scale difference due to perspective. We selected a proportional factor ($\tau = 1.02$) to modify the pixel-wise bandwidth so that the total sum of the pixels in a density map aligns with the observed number of passengers. Eq. (9) represents the relationship between a pixel value (P_{ij}) in the perspective map and the corresponding

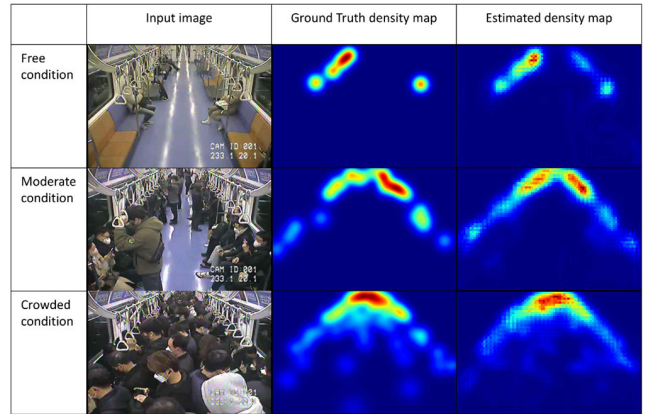


FIGURE 3. Examples of raw input images and their ground truth density map for training images.

bandwidth (B_{ij}) used to create a density map. Fig. 4 illustrates the comparison across all training images between the observed passenger count and the total sum of pixel values of a density map. After adjustments with τ , both measures demonstrate a strong agreement.

$$B_{ij} = \tau P_{ij} \quad (9)$$

During the construction of the ground truth density maps prior to training the augmented model, the most labor-intensive task involves preparing data for the passenger head size to obtain a perspective map. We manually identified the size of 12,500 passenger heads across all training images. We investigated the minimum number of passenger heads required to be annotated in order to achieve an R^2 index comparable to using the entire dataset. Fig. 5 indicates that annotating only 500 passenger heads yields a similar outcome to annotating the entire dataset. For a fair comparison, the R^2 index was calculated for the complete dataset when each regression analysis was conducted with a smaller sample size. This suggests that a small amount of manual labor is required to label passenger heads in order to obtain a reliable perspective map. After determining the perspective map, the ground-truth density map can be easily made based on it for each training image.

Eq. (10) designates the loss function associated with the discrepancy between estimated density maps and the corresponding ground truth density maps. In this equation, x_i represents pixel values in the ground truth density map for the i^{th} training image, and \hat{x}_i denotes pixel values in the estimated density map returned by the augmented regression-based model.

$$\mathcal{L}_D = \frac{1}{N_{\mathfrak{A}}} \sum_{i \in \mathfrak{A}} (x_i - \hat{x}_i)^2 \quad (10)$$

The calibration of the augmented regression-based model was exclusively performed in a sequential manner. The simultaneous calibration approach is not applicable to the model due to the absence of a density map for ranking data. The sequential procedure commences with the pretraining of the

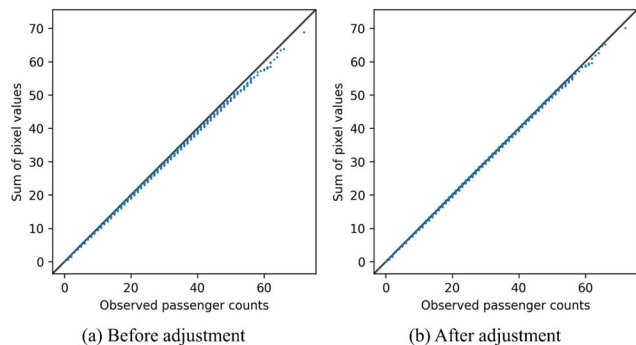


FIGURE 4. Relationship between the passenger count and the sum of pixel values in a density map.

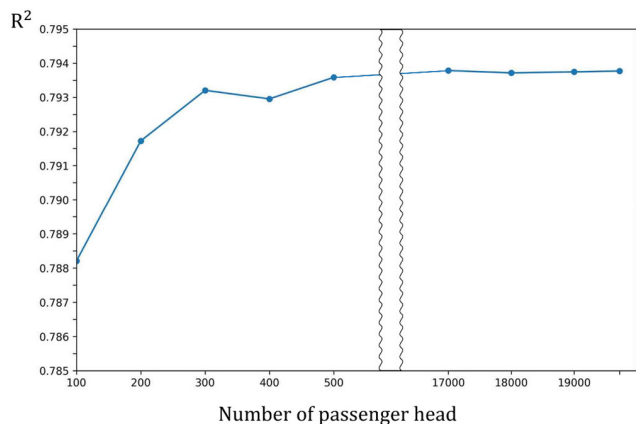


FIGURE 5. Finding minimum number of annotations to have the same performance as full data regression provides.

count regression head (the upper pipeline in Fig. 1) using ranked image data. After concluding the pretraining process, we fine-tuned the model with two pipelines based on ground truth density maps and passenger counts, thereby minimizing the combined loss function.

$$\mathcal{L}_{aug} = \mathcal{L}_{MSE} + \lambda_D \mathcal{L}_D \quad (11)$$

Eq. (11) illustrates the combined loss function for the augmented regression-based model. This function is composed of both the loss [Eq. (4)] for aligning with observed passenger counts, and the loss [Eq. (10)] for aligning with the ground truth density maps. Here, λ_D represents the relative weight of the latter loss.

IV. DATA ACQUISITION

Image data was collected from the Shinbundang Metro line, which connects the southern region of the Seoul Metropolitan Area (SMA) to the Gangnam city center. Data collection took place on a specific date (March 16, 2023). A single train consisting of six cars was chosen to collect video images during the 19-hour operational period. Each car was equipped with two overhead video cameras. As passenger load is likely to remain consistent while the train moves between two consecutive stations, a single image was randomly selected each

TABLE 1. average times to label an image for different labeling methods.

| Labeling method | Average time taken to label an image |
|--------------------------------|--------------------------------------|
| Passenger counts with position | 36 seconds per image |
| Ranks in paired images | 1.64 seconds per image pair |
| Ranks in tripled images | 2.88 seconds per image triplet |
| Bounding boxes | 120 seconds per image |

time the train passed a segment. After filtering out unsuitable images, a total of 3,000 images were retained. Of these, 2,400 images were designated for training and 600 images were allocated for testing.

From the 2,400 training images, 240 images were utilized to create 21,000 ranked image pairs, and 60 images were used to generate 21,000 ranked triplets. The remaining 2,100 images were reserved for count regression. For comprehensive model evaluation, all 3,000 images were fully annotated with bounding boxes for passenger heads. The total number of passengers within these images amounted to 76,676.

We conducted experiments to measure the average times taken to label an image using different methods. Table 1 presents empirical results obtained from three human labelers. Drawing bounding boxes for each human head consumed the most time, followed by counting passengers using head positions. Labeling images with ranks required significantly less time, thus rationalizing the adoption of the proposed calibration approach.

V. TEST RESULTS AND COMPARISON

The proposed regression-based models were trained and tested on separate image sets. We will focus on presenting and comparing test results across different training methods. Furthermore, two different backbones of the model were tested (VGG16 and ResNet50). The model performance was evaluated using two metrics: Mean Absolute Error (MAE) and Mean Squared Error (MSE). No case was observed where these two indices contradicted each other. MAE was particularly convenient for comparison because its scale aligns with passenger counts. As a reference, the test results of the simple and augmented models are presented without the use of paired or tripled data for training.

For the simple regression-based model, the sequential approaches proved superior to the simultaneous approaches for both backbones. When VGG16 was adopted for backbone, Among the various sequential approaches, pretraining the model on both paired and tripled images recorded the best performance, reducing MAE by 23.7% and MSE by 33.5% respectively, compared to the baseline approach. When ResNet50 was used for backbone, MAE was reduced by 20.1% and MSE was reduced by 32.6%. When comparing the performance between two different backbones, VGG16 outperformed ResNet50 for both the baseline case and the case where using ranked data sequentially.

TABLE 2. Comparing the performance of regression-based passenger counting models across different calibration schemes. (a) Comparison results from the model using VGG16 as backbone.

(a) Comparison results from the model using VGG16 as backbone

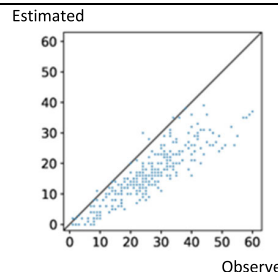
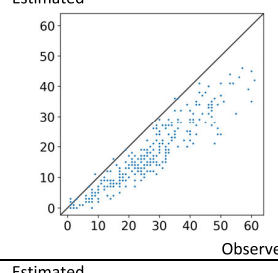
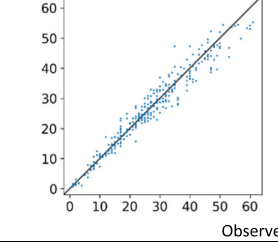
| Model | Calibration method | Training Process | VGG16 | |
|---|---|--|--------------|--------------|
| | | | MAE | MSE |
| Simple regression on-based model | Baseline | Passenger counts | 3.38 | 19.52 |
| | Simultaneous calibration ($\lambda_{pair}= \lambda_{triple}= 10$) | Paired data + Passenger counts | 2.84 | 15.16 |
| | | Tripled data + Passenger counts | 2.88 | 14.08 |
| | | Paired data + Tripled data + Passenger counts | 2.71 | 14.12 |
| | Sequential calibration | Paired data \rightarrow Passenger counts | 2.83 | 15.56 |
| | | Tripled data \rightarrow Passenger counts | 2.81 | 14.01 |
| | | Paired data \rightarrow Tripled data \rightarrow Passenger counts | 2.63 | 13.23 |
| Tripled data \rightarrow Paired data \rightarrow Passenger counts | | 2.66 | 12.98 | |
| (Tripled data + Paired data) \rightarrow Passenger counts | | 2.58 | 12.98 | |
| Augmented regression on-based model | Baseline ($\lambda_D = 10^5$) | Passenger counts + Density maps | 2.59 | 13.00 |
| | Sequential calibration ($\lambda_D = 10^2$) | Paired data \rightarrow (Passenger counts + Density maps) | 2.39 | 11.93 |
| | | Tripled data \rightarrow (Passenger counts + Density maps) | 2.38 | 10.73 |
| | | Paired data \rightarrow Tripled data \rightarrow (Passenger counts + Density maps) | 2.22 | 9.04 |
| | | Tripled data \rightarrow Paired data \rightarrow (Passenger counts + Density maps) | 2.34 | 11.25 |
| | | (Tripled data + Paired data) \rightarrow (Passenger counts + Density maps) | 2.35 | 10.54 |

(b) Comparison results from the model using ResNet50 as backbone

| Model | Calibration method | Training Process | ResNet50 | |
|---|---|--|--------------|--------------|
| | | | MAE | MSE |
| Simple regression on-based model | Baseline | Passenger counts | 3.02 | 16.91 |
| | Simultaneous calibration ($\lambda_{pair}= \lambda_{triple}= 10$) | Paired data + Passenger counts | 2.74 | 14.26 |
| | | Tripled data + Passenger counts | 2.73 | 13.68 |
| | | Paired data + Tripled data + Passenger counts | 2.61 | 12.77 |
| | Sequential calibration | Paired data \rightarrow Passenger counts | 2.62 | 13.00 |
| | | Tripled data \rightarrow Passenger counts | 2.68 | 13.42 |
| | | Paired data \rightarrow Tripled data \rightarrow Passenger counts | 2.44 | 11.90 |
| Tripled data \rightarrow Paired data \rightarrow Passenger counts | | 2.44 | 11.26 | |
| (Tripled data + Paired data) \rightarrow Passenger counts | | 2.41 | 11.40 | |
| Augmented regression on-based model | Baseline ($\lambda_D = 10^5$) | Passenger counts + Density maps | 2.60 | 12.15 |
| | Sequential calibration ($\lambda_D = 10^2$) | Paired data \rightarrow (Passenger counts + Density maps) | 2.48 | 10.53 |
| | | Tripled data \rightarrow (Passenger counts + Density maps) | 2.46 | 11.10 |
| | | Paired data \rightarrow Tripled data \rightarrow (Passenger counts + Density maps) | 2.35 | 10.60 |
| | | Tripled data \rightarrow Paired data \rightarrow (Passenger counts + Density maps) | 2.43 | 11.15 |
| | | (Tripled data + Paired data) \rightarrow (Passenger counts + Density maps) | 2.43 | 11.09 |

Regarding the augmented regression-based model, both the density map and passenger count were simultaneously matched with the ground truth during training. Comparing

TABLE 3. comparison in counting performance between the detection- and regression-based models.

| Model | Estimated vs. Observed Counts | MAE | MSE |
|--------------------------------------|---|------|--------|
| Detection-based model (EfficientDet) |  | 9.43 | 121.43 |
| Detection-based model (Yolo v5) |  | 9.11 | 109.29 |
| Augmented regression-based model |  | 2.22 | 9.04 |

the two baseline cases, the augmented model surpassed the simple model for both backbones, with a considerable performance gain attributed to the inclusion of density maps in training. The inclusion of density maps in training reduced both indices by 23.4% and 33.4%, respectively, when comparing baseline cases.

Furthermore, sequential calibration of the model with both paired and tripled data significantly enhanced the performance of the augmented model. Compared to the baseline case of the augmented regression-based model, both indices were reduced by 14.3% and 30.5% for VGG16, and 9.6% and 12.8% for ResNet50, respectively.

The simple regression-based method best performed when sequentially using tripled and paired datasets, whereas the augmented regression-based method recorded the best performance when trained on the integrated dataset. The potential reason for the discrepancy might stem from using the density map fitting in the augmented method. The results of testing both models across different calibration approaches are summarized in Table 2.

Table 3 elucidates why the detection-based method is ill-suited for crowd counting. Images from a metro train’s video footage feature many small objects and occlusions, leading the model to consistently underestimate the passenger count.

The detection-based model is unable to recognize small objects from remote perspectives. Consequently, the MAE from an YOLO5, one of the most prevalent detectors in the field, is four times larger than the best result from the proposed regression-based model trained additionally on ranked images. When comparing the performance of the two “state-of-the-art” object detectors, YOLO5 slightly outperformed EfficientDet.

VI. CONCLUSION

Determining passenger load in transit vehicles using computer vision technologies presents significant challenges, primarily due to the burdensome labeling tasks. Annotating passenger heads in images, either by positioning or delineating, necessitates considerable human effort before training a model for crowd counting. This study proposes an efficient training method aimed at reducing the required human effort for labeling images without compromising the performance of the model.

Ranked images significantly improved the training performance as ranking images requires considerably less human effort than annotating images by positioning passengers or drawing bounding boxes. The simple regression model was trained using both sequential and simultaneous approaches, with the sequential training method proving superior.

When augmenting the calibration of a passenger counting model with density maps, only the sequential approach was tested, since no ground truth density map was available for ranked images. As a result, the augmented model outperformed the simple regression-based model with or without training on ranked images.

The proposed approach is currently limited to a specific metro line in the Seoul metropolitan area. Future work should focus on developing a generic training method that can be applied to any transit system without the need of an additional labeling task. The image resolution used in this study is relatively low compared to those in previous studies, which might explain why the resultant metrics are worse than expected. However, the model performance is considerably more reliable than the current method employed by the Shinbundang line, which measures passenger load based on changes in train mass.

REFERENCES

- [1] C. Oberli, M. Torres-Torriti, and D. Landau, “Performance evaluation of UHF RFID technologies for real-time passenger recognition in intelligent public transportation systems,” *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 748–753, Sep. 2010.
- [2] M. L. Ferreira, J. A. de Gouveia, E. Facchini, M. Pokorny, and E. Dias, “Real time monitoring of public transit passenger flows through radio frequency identification-RFID technology embedded in fare smart cards,” *Latest Trends Syst.*, vol. 2, pp. 599–605, Sep. 2012.
- [3] R. Kovacs, L. Nadai, and G. Horvath, “Concept validation of an automatic passenger counting system for trams,” in *Proc. 5th Int. Symp. Appl. Comput. Intell. Informat.*, May 2009, pp. 211–216.
- [4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
- [5] M. Tan, R. Pang, and Q. V. Le, “EfficientDet: Scalable and efficient object detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10778–10787.
- [6] A. B. Chan, Z.-S. John Liang, and N. Vasconcelos, “Privacy preserving crowd monitoring: Counting people without people models or tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–7.
- [7] C. Wang, H. Zhang, L. Yang, S. Liu, and X. Cao, “Deep people counting in extremely dense crowds,” in *Proc. 23rd ACM Int. Conf. Multimedia*, Oct. 2015, pp. 1299–1302.
- [8] C. Zhang, H. Li, X. Wang, and X. Yang, “Cross-scene crowd counting via deep convolutional neural networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 833–841.
- [9] 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.
- [10] Q. Wang, J. Wan, and Y. Yuan, “Deep metric learning for crowdedness regression,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2633–2643, Oct. 2018.
- [11] C. Shang, H. Ai, and B. Bai, “End-to-end crowd counting via joint learning local and global count,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 1215–1219.
- [12] B. Sheng, C. Shen, G. Lin, J. Li, W. Yang, and C. Sun, “Crowd counting via weighted VLAD on a dense attribute feature map,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1788–1797, Aug. 2018.
- [13] X. Cao, Z. Wang, Y. Zhao, and F. Su, “Scale aggregation network for accurate and efficient crowd counting,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 734–750.
- [14] Y. Li, X. Zhang, and D. Chen, “CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1091–1100.
- [15] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, “Single-image crowd counting via multi-column convolutional neural network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 589–597.
- [16] V. A. Sindagi and V. M. Patel, “Generating high-quality crowd density maps using contextual pyramid CNNs,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1879–1888.
- [17] G. Olmschenk, H. Tang, and Z. Zhu, “Crowd counting with minimal data using generative adversarial networks for multiple target regression,” in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1151–1159.
- [18] G. Olmschenk, Z. Zhu, and H. Tang, “Generalizing semi-supervised generative adversarial networks to regression using feature contrasting,” *Comput. Vis. Image Understand.*, vol. 186, pp. 1–12, Sep. 2019.
- [19] L. Wang, Y. Li, and X. Xue, “CODA: Counting objects via scale-aware adversarial density adaption,” 2019, *arXiv:1903.10442*.
- [20] L. Dong, H. Zhang, D. Zhou, J. Shi, and J. Ma, “CCTwins: A weakly-supervised transformer-based crowd counting method with adaptive scene consistency attention,” *IEEE Trans. Consum. Electron.*, early access, May 2023, doi: 10.1109/TCE.2023.3274651.
- [21] A. Hafeezallah, A. Al-Dhamari, and S. A. R. Abu-Bakar, “Multi-scale network with integrated attention unit for crowd counting,” *Comput. Mater. Continua*, vol. 73, no. 2, pp. 3879–3903, 2022.
- [22] B. Li, Y. Zhang, H. Xu, and B. Yin, “CCST: Crowd counting with Swin transformer,” *Vis. Comput.*, vol. 39, no. 7, pp. 2671–2682, Jul. 2023.
- [23] X. Qian, C. Li, W. Wang, X. Yao, and G. Cheng, “Semantic segmentation guided pseudo label mining and instance re-detection for weakly supervised object detection in remote sensing images,” *Int. J. Appl. Earth Observ. Geoinf.*, vol. 119, May 2023, Art. no. 103301.
- [24] X. Qian, Y. Huo, G. Cheng, C. Gao, X. Yao, and W. Wang, “Mining high-quality pseudoinstance soft labels for weakly supervised object detection in remote sensing images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–15, 2023.
- [25] L. Li, X. Yao, X. Wang, D. Hong, G. Cheng, and J. Han, “Robust few-shot aerial image object detection via unbiased proposals filtration,” *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–11, 2023.
- [26] X. Qian, B. Wu, G. Cheng, X. Yao, W. Wang, and J. Han, “Building a bridge of bounding box regression between oriented and horizontal object detection in remote sensing images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–9, 2023.
- [27] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, Munich, Germany, Cham, Switzerland: Springer, 2015, pp. 234–241.

- [28] R. Sutopo, J. M.-Y. Lim, V. M. Baskaran, K. Wong, M. Tistarelli, and H. F. Liau, "Appearance-based passenger counting in cluttered scenes with lateral movement compensation," *Neural Comput. Appl.*, vol. 33, no. 16, pp. 9891–9912, Aug. 2021.
- [29] J. D. Ortuzo and L. G. Willumsen, *Modelling Transport*, 2nd ed. New York, NY, USA: Wiley, 1995.
- [30] R. D. Luce and P. Suppes, "Preference, utility and subjective probability," in *Handbook of Mathematical Psychology*, R. D. Luce, R. R. Bush, and E. Galanter, Eds. New York, NY, USA: Wiley, 1965.
- [31] R. G. Chapman and R. Staelin, "Exploiting rank ordered choice set data within the stochastic utility model," *J. Marketing Res.*, vol. 19, no. 3, p. 288, Aug. 1982.



HOJUN LEE is currently pursuing the master's degree with the Department of Smart City, Chung-Ang University. His research interests include machine learning and visual recognition.



KYEONGJUN LEE is currently pursuing the master's degree with the Department of Urban Engineering, Chung-Ang University. His research interests include machine learning and visual recognition.



JIWON KANG is currently pursuing the master's degree with the Department of Urban Engineering, Chung-Ang University. Her research interests include machine learning and visual recognition.



KEEMIN SOHN was born in Seoul, South Korea, in 1968. He received the Ph.D. degree in transportation planning from the Department of Civil Engineering, Seoul National University (SNU), in 2003.

He is currently a Professor with the Department of Urban Engineering and the Department of Smart City, Chung-Ang University. His research interest includes the applications of artificial intelligence to transportation engineering and planning.

...