Dual Reconstructive Autoencoder for Crowd Localization and Estimation in Density and FIDT Maps

FELIPE I. LAMAS, (Member, IEEE), JORGE E. PEZOA, (Member, IEEE), SEBASTIÁN E. GODOY, (Member, IEEE), GABRIEL A. SAAVEDRA, (Member, IEEE), SERGIO N. TORRES, (Member, IEEE), GONZALO A. MONTALVA, AND WEIXIAO MENG, (Senior Member, IEEE)

1Department of Electrical Engineering, Universidad de Concepción, Concepción 4030000, Chile
2Department of Civil Engineering, Universidad de Concepción, Concepción 4030000, Chile
3School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China

Corresponding author: Felipe I. Lamas (flamas@udec.cl)

This work was supported in part by the Agencia Nacional de Investigación y Desarrollo de Chile (ANID)/National Natural Science Foundation of China (NSFC) under the International Co-Operation Program under Grant PII180009, and in part by the ANID-Subdirección de Capital Humano/Magíster Nacional/2021–22210563 Fellowship.

ABSTRACT This paper proposes crowd estimation technology to help authorities make the right decisions in times of crisis. Specifically, deep learning models have faced these challenges, achieving excellent results. In particular, the trend of using single-column Fully Convolutional Networks (FCNs) has increased in recent years. A typical architecture that meets these characteristics is the autoencoder. However, this model presents an intrinsic difficulty: the search for the optimal dimensionality of the latent space. In order to alleviate such difficulty, we propose a dual architecture consisting of two cascaded autoencoders. The first autoencoder is responsible for carrying out the masked reconstruction of the original images, whereas the second obtains crowd maps from the outputs of the first one. In this way, our architecture improves the location of people and crowds in Focal Inverse Distance Transform (FIDT) maps, resulting in more accurate count estimates than estimates obtained through a single autoencoder architecture.

INDEX TERMS Deep learning, convolutional neural networks, artificial intelligence, cascaded autoencoders, crowd localization/counting, FIDT maps, density maps.

I. INTRODUCTION

The global figures for Covid-19 infections show the rapid spread of the virus in Chile [1] and the world [2]. It is known that crowded spaces are directly related to high infection rates. At the beginning of the pandemic, health authorities used various mechanisms to avoid crowds, such as the permanent closure of shopping centers, curfews, preventive and mandatory quarantines, and teleworking. However, these mechanisms notably harmed global economies and the general welfare of the population. Given the successful vaccination campaigns, it has been possible to return to the routine, maintaining social distancing and using masks. However, there are still crowds of people who do not respect the protocols for different reasons, which are generally very difficult to handle. As such, crowd detection and management are still critical.

Similarly, crowd management is highly critical in natural disasters. Earthquakes, tsunamis, forest fires, floods, and mudslides are some natural phenomena that frequently cause stampedes of uncontrolled people. Two of the countries most exposed to natural catastrophes are China and Chile [3]. Specifically, earthquakes are frequent threats in both countries, generating fatalities and considerable material losses [4].

Automatic people-counting technologies can help authorities to make vital decisions in difficult moments to reduce civilian casualties. Manual counting of people from a video...
feed of a security camera is not an option since, in general, people in these scenes change constantly. Moreover, manual counting is time-consuming; typically, the count is required in almost real-time. As such, machine-learning approaches are required to tackle this problem.

There are three main machine-learning approaches in crowd estimation: detection, regression, and density maps [5]. The detection approach was the first to appear and was mainly characterized by sliding window detectors [6], [7], [8], [9]. This approach fails when many occluded people are in the image. This problem was solved using texture and foreground feature-extraction regression methods [10], [11]. Density maps were later introduced to improve count results compared to estimate count directly from images [12]. Nowadays, Focal Inverse Distance Transform (FIDT) maps proposed by Liang et al. [13] are often used due to their better results in locating individual people within the images, unlike other density maps.

In recent years, deep-learning approaches have been proliferating for solving this problem, mainly using Convolutional Neural Networks (CNNs) to develop new models. CNNs are typically classified according to their architecture: basic, single-column, and multi-column. The basic networks were the first to appear and have fully connected layers (dense layers) at the end of their architecture to obtain the count estimate [14], [15], [16], [17]. Single-column [18], [19], [20], [21], [22], [23], [24] and multi-column [12], [13], [25], [26], [27] networks appeared later and are characterized by estimating the number of people from the estimated density map. Since dense layers are not required to obtain density maps, these last two networks are FCNs, where the difference between the two types lies in the number of columns in the architecture.

Currently, there is a trend in using single-column FCNs. In particular, a trivial architecture that meets these specifications is the encoding and decoding model, known as the autoencoder [5], [21], [28], [29], [30], [31], [32], [33]. However, an intrinsic difficulty of this model is the choice of the dimensionality of the latent coding space. In ideal terms, such a latent space should have the essential data characteristics that allow the network to accurately perform the task for which it was trained. If the dimensionality of this space is less than the optimal value, the network will have severe problems executing the task. In contrast, if the dimensionality is high, the latent space will have redundant features, resulting in poor feature extraction by the network. The difficulty in finding the optimum is based on the close relationship between the latent space and the data particularities.

To reduce the complexity of obtaining an autoencoder’s optimum latent space dimensionality for the simultaneous tasks of locating and estimating people in crowds, we propose a dual architecture composed of two cascaded autoencoders. The first autoencoder is responsible for generating reconstructive masking of the input images, resulting in images in which only the heads of the people are present. The second autoencoder takes the output images of the first one and generates the FIDT maps. Through such an approach, it is possible to improve the estimation and location of people in crowds on FIDT maps compared to single autoencoder approaches. Our dual model is suitable for counting low and medium crowd densities, with similar location performance for all types of densities. The intermediate output can be used to improve the performance of other deep learning models, such as facial expression recognition models.

In summary, the main contribution of this work in the field of crowd counting/location is the design and evaluation of a novel dual architecture based on two cascaded autoencoders, which enables:

1) An improved people counting and location, compared to a single autoencoder architecture on FIDT maps.
2) Our dual model is suitable for counting low and medium crowd densities, with uniform location performance for all types of densities.
3) The intermediate output can be used to improve the performance of other deep learning models, such as facial expression recognition models.

This paper is structured as follows: Section II explains the types of crowd maps used in the research. Section III exposes the dual-autoencoder rationale, its architecture, and its training and evaluation details. The results are shown in Section IV, whereas Section V presents the conclusion and future work.

II. CROWD MAPS

In particular, we employed two types of crowd maps: density maps and FIDT maps. Next, we detail step by step how to obtain each of them. It is worth mentioning that we used such maps as ground truths.

A. DENSITY MAPS

A density map is a crowd map that represents people’s heads by normalized Gaussian kernels [12]. The normalization aims to make the integral of each kernel equal to 1 so that we can compute the total head count as the integral of the entire density map. In particular, it is possible to obtain this type of map employing the procedure explained below.

Consider a crowd image with \( N \) people and a set of points \( A \) containing the position \( (x_i, y_i) \), with \( i = 1, 2, \ldots, N \), of each head in the image [12]. If we represent each head as a delta distribution \( \delta(x - x_i, y - y_i) \), we obtain the complete image \( C \) using (1),

\[
C(x, y) = \sum_{(x_i, y_i) \in A} \delta(x - x_i, y - y_i). \tag{1}
\]

Then, it is possible to obtain the density map \( D \) through (2),

\[
D(x, y) = C(x, y) \ast K_{\sigma}(x, y), \tag{2}
\]

where \( \ast \) represents the two dimensional convolution, \( K \) is a Gaussian kernel with \( \sigma_i = \psi d_i \), where \( \psi \) is a constant, and \( d_i = \frac{x_i^2 + y_i^2}{2} \). Y. Zhang et al. [12] have empirically found that \( \psi = 0.3 \) is the best value. The variable \( d_i \) is
the set of distances between each head \((x_i, y_i)\) and its \(k\) closest neighbors, \(d_1^i, d_2^i, \ldots, d_k^i\). An alternative approach is to consider a constant standard deviation, which allows us to generalize the kernel size for all heads regardless of their size in the images. Here, we adopt the latter approach based on the results reported by V. K. Valloli et al. [21], who, with fixed-size kernels, obtained a 25% improvement in the Mean Absolute Error (MAE) metric compared to geometry-adaptive kernels using a similar architecture.

Figure 1(a) shows an image of crowds from the ShanghaiTech dataset [12], [34], whereas Figure 1(b) exposes its density map. Such a dataset provides crowd images, along with the location points of each head. Thus, we generated the exposed density map using the previous procedure on the set of ground truth points of the respective image. For the Gaussian kernels, we use a constant standard deviation of 4 (i.e., \(\sigma = 4\)), and a window of 67 \(\times\) 67 pixels (i.e., \(\mu = 67\)). We have tried several values for these parameters and selected those that generate the best results in our models.

### B. FOCAL INVERSE DISTANCE TRANSFORM MAPS

A Focal Inverse Distance Transform (FIDT) map is a type of crowd map characterized by accurately representing the location of each person in all kinds of crowd densities. The improvement in localization is the main difference with the density maps; however, the counting procedure requires a local maximum detection strategy [13]. Next, we explain the mathematical derivation of a FIDT map.

Consider a crowd image and a set of points \(A\) with the positions \((x_i, y_i)\), with \(i = 1, 2, \ldots, N\), of the \(N\) people’s heads in the image. From this, we can obtain the Euclidean distance transform map [13] through (3),

\[
E(x, y) = \min_{(x_i, y_i) \in A} \sqrt{(x - x_i)^2 + (y - y_i)^2}. \tag{3}
\]

Then, we calculate the FIDT map as detailed in (4),

\[
F(x, y) = \frac{1}{E(x, y)^{\alpha E(x, y) + \beta} + B}. \tag{4}
\]

where we use \(B = 1\) to avoid dividing by zero and adopt \(\alpha = 0.02\) and \(\beta = 0.75\) as recommended by D. Liang et al. [13].

Figure 1(c) shows the FIDT map of the sample image. We obtained this map using the previously detailed procedure on the location points provided by the dataset for each corresponding image. Compared with Figure 1(b), we observe that the FIDT map significantly improves the location of people in the dense crowd area.

### III. DUAL RECONSTRUCTIVE AUTOENCODER

#### A. DUAL-AUTOENCODER RATIONALE

This work focused on designing a deep learning model to count and locate people within a given scene. From a general point of view, using an autoencoder would be adequate to compress the crowds’ characteristics in their latent space. Nevertheless, finding the proper dimensionality for such a coding space is the main problem. Consequently, we propose to alleviate this difficulty through an architecture of two cascaded autoencoders. The first autoencoder aims to learn the characteristics of people’s heads in crowd imagery to obtain an output image as a mosaic of circular masks of the input scene. (The center of each circle gives the location of people’s heads in the input image.)

The second autoencoder focuses primarily on generating density maps or FIDT maps. Subsequently, we obtain the estimates of the number of people from the crowd maps. The specific objective of using our dual architecture, instead of a single autoencoder, is to separate the tasks of detecting people and generating the points representative of each head. Indeed, a single autoencoder architecture must address both tasks together, drastically complicating training and generating even more difficulties in finding essential features for the latent space.

#### B. PROPOSED ARCHITECTURE

We show the architecture of the proposed neural network in Figure 2. Here we present both variants: the one that computes the FIDT maps and the one that computes the density maps, which differ only in the last block. Our model comprises two cascaded autoencoders and initially performs the reconstructive masking of the input images; for
these reasons, we name it DRA. DRA models use part of the architecture proposed by V. K. Valloli et al. [21] as a basis.

The first autoencoder takes the crowd image and converts it to an image in which only the heads of the people are present (reconstructive masking). Subsequently, the second autoencoder takes this output and generates the FIDT map or the density map, depending on the selected block at the end.

Both autoencoders have a contraction path and an expansion path. The variants of the DRA model (for both types of crowd maps) agree on the path of contraction, differing only at the end of the expansion path (dashed-blue square in Figure 2). DRA models perform feature extraction on contraction paths, which use the first five blocks of the Torchvision VGG16_BN model (without the fully connected layer). In particular, we use the pre-trained VGG16_BN on the ImageNet dataset. Next, we explain the flow of a color image in the DRA model.

C. DATA FLOW

The input RGB image is taken by the first block of the contraction path (B1_C2) of the first autoencoder, composed of two convolution layers of $3 \times 3 \times 64$ that have Batch Normalization (BN) and Rectified Linear Unit (ReLU) activation function. The feature maps are then passed through a max-pooling layer of $2 \times 2$ with stride 2, decreasing their resolution by half. The output then goes into block B2_C2, which has two convolution layers of $3 \times 3 \times 128$ with BN and ReLU. Once again, we reduce the resolution using a max-pooling layer $2 \times 2$ with stride 2 to send the feature maps to block B3_C3 composed of three convolution layers with kernels equal to those of the previous convolution layers, 256 outputs, BN, and ReLU. We apply max-pooling and three convolution layers of $3 \times 3 \times 512$, where each has batch normalization and ReLU (block B4_C3). We finish the contraction path by applying max-pooling, followed by the B5_C3 block of 3 convolutional layers with the same parameters as the previous block.

The expansion path begins applying a nearest-neighbor interpolation layer with scale factor 2, doubling the resolution of the feature maps. Then, we concatenate these maps with the outputs of block B4_C3, generating 1024 feature maps to which we apply a convolution layer of $1 \times 1 \times 256$ with BN and ReLU. Next, we send the outputs to a convolution layer with a $3 \times 3$ kernel, 256 outputs, BN, and ReLU. We then double the resolution via the nearest upsample, concatenate them with block B3_C3’s outputs, and pass them through a $1 \times 1 \times 128$ convolution layer with batch normalization and ReLU. Next, we double the resolution, concatenate with the outputs of block B2_C2, and apply a convolution of $1 \times 1 \times 64$ followed by a convolution of $3 \times 3 \times 64$, both with BN and rectified linear unit. The expansion path ends with a further doubling of the resolution, concatenation with the outputs of block B1_C2, and application of convolutional layers of $1 \times 1 \times 32 + BN + ReLU, 3 \times 3 \times 32 + BN + ReLU,$ and $3 \times 3 \times 3 + ReLU$. The passage of the RGB image through the contraction and expansion paths of the first autoencoder generates the so-called masked reconstruction of the original image. In such an output, only the heads of the people from the original image are present.

The masked reconstruction enters the second autoencoder, passing through its contraction path first. As can be seen from Figure 2, such a path is identical to that of the first autoencoder. In the case of the expansion path, it is identical to its counterpart in the first autoencoder until reaching the interpolation layer of 128 feature maps. At this point lies the difference between the variant for FIDT maps and that for density maps. In the case of FIDT maps, the feature maps go through a section identical to the respective section in the expansion path of the first autoencoder until reaching the last block, which has an extra $1 \times 1 \times 1$ convolution layer $+ ReLU$ at the end, thus generating the FIDT map. On the other hand, the variant designed for density maps takes the
128 upsampled feature maps, concatenates with those from block B2_C2, and passes them through convolutional layers of $1 \times 1 \times 64 + \text{BN} + \text{ReLU}$, $3 \times 3 \times 64 + \text{BN} + \text{ReLU}$, $3 \times 3 \times 32 + \text{BN} + \text{ReLU}$, and $1 \times 1 \times 1 + \text{ReLU}$, obtaining the density map.

As mentioned before, our model will be compared to a single autoencoder architecture that directly generates the crowd maps from the input images. For this reason, the model is called SA. The architecture of the SA model corresponds to the last autoencoder of the DRA neural network, so there is a variant for each crowd map (see Figure 3).

D. TRAINING STAGE

We use Part B of the ShanghaiTech dataset for training and evaluation. This data set has 400 training images and 316 evaluation images [12], [34]. The images in both subsets have a resolution of $768 \times 1024$. The average number and standard deviation of people in the images of the training subset are 123 and 94. Likewise, the minimum and the maximum number of people are 12 and 576, respectively. In turn, the average, standard deviation, minimum, and maximum of people in the evaluation subset images are 124, 95, 9, and 539, respectively.

In the training of the DRA model, we use a Gaussian distribution with zero mean and 0.01 standard deviation to randomly initialize the weights of all trainable layers of the expansion paths. Moreover, we used 50 epochs and a batch size of 1, generating 20,000 iterations. We also performed on-the-fly data augmentation through 14 random $400 \times 400$ crops and horizontal flips half the time. We train the first autoencoder for 8,000 iterations and the second for the following 4,000 iterations. Finally, we enable the early stopping method to monitor validation loss when training the entire model (for both autoencoders) for the remaining 8,000 iterations.

In such a training scheme, we use different loss functions depending on the type of crowd map. In particular, the FIDT map variant uses the Mean Squared Error (MSE) and Independent Structural Similarity Index Measure (I-SSIM) loss functions, whereas the density map variant only uses MSE. Specifically, in the training of the first autoencoder of the DRA model for FIDT maps, we used the MSE loss function multiplied by a constant as detailed in (5),

$$L_{1F} = \eta L_{MSE} = \eta \frac{1}{M} \sum_{i=1}^{M} (y_i - \hat{y}_i)^2,$$

where $\eta = 10^5, M$ is the number of pixels, $y_i$ is the $i$-th ground truth value, and $\hat{y}_i$ is the $i$-th estimated value.

The second autoencoder of the DRA variant for FIDT maps uses a loss given by the sum of the MSE and I-SSIM losses as shown in (6),

$$L_{2F} = L_{MSE} + L_{I-SSIM},$$

where $L_{I-SSIM}$ is given by [13]:

$$L_{I-SSIM} = \frac{1}{N} \sum_{i=1}^{N} L_{SSIM}(P_i, G_i).$$

Here, $N$ is the total number of people, and $P_i$ and $G_i$ are the prediction and ground truth for the $i$-th $30 \times 30$ independent instance region. The Structural Similarity Index Measure (SSIM) loss in (7) is given by (8),

$$L_{SSIM}(P, G) = 1 - \text{SSIM}(P, G),$$

where SSIM corresponds to the Structural Similarity Index Measure, which is calculated by (9),

$$\text{SSIM}(P, G) = \frac{(2\mu_P\mu_G + \lambda_1)(2\sigma_{PG} + \lambda_2)}{(\mu_P^2 + \mu_G^2 + \lambda_1)(\sigma_P^2 + \sigma_G^2 + \lambda_2)},$$

where $P$ and $G$ are the predicted and ground truth maps, respectively; $\mu_P$ and $\sigma_P$ (correspondingly, $\mu_G$ and $\sigma_G$) are the mean and standard deviation of the predicted map, $P$ (correspondingly of the ground-truth map, $G$). As with

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.png}
\caption{Architecture of the Single Autoencoder (SA) model.}
\end{figure}
the instance size, we adopt the values $\lambda_1 = 0.0001$ and $\lambda_2 = 0.0009$ from [13]. We carried out the training of the complete model through the joint loss function $L_{DF} = L_{DF1} + L_{DF2}$.

To train the first autoencoder of the DRA variant for density maps, we use the same loss function as for the variant for FIDT maps, that is, $L_{ID} = L_{DF1}$. We used the loss function $L_{ID2} = \tau L_{MSE}$, with $\tau = 10^4$, for the second autoencoder of the density map variant. As before, the joint loss $L_{ID}$ used to train the complete model corresponds to the sum of the individual losses, namely $L_{ID} = L_{ID1} + L_{ID2}$.

We employed Adam optimization [35] and a learning rate equal to $10^{-4}$. Likewise, for FIDT maps, we used weight decay equal to $5 \times 10^{-4}$, whereas, for density maps, we used $5 \times 10^{-3}$. All these parameters were experimentally determined to achieve the best performance. The hardware employed for training was an NVIDIA A100 Tensor Core GPU with a 40 GB HBM2 @ 1.6 TB GPU memory size, running on an accelerator-optimized (A2) Google Cloud virtual machine with 12 vCPUs and 85 GB RAM. We used Python programming language in its version 3.7.10 and PyTorch 1.9 machine learning framework.

For the first autoencoder of both variants of the DRA model, the targets were the original images multiplied by their respective binary head masks. Such masks (used only for training) were obtained by thresholding the ground truth density maps (from the database) through a threshold of $10^{-5}$. The targets of the second autoencoder correspond to crowd maps, where the type of map used depends on the selected DRA variant. The loss function used by each SA model variant corresponds directly to the one used by the second autoencoder of the respective DRA model variant. To perform fair comparisons, we train each SA variant with the same hyperparameters as the respective DRA variant.

In order to avoid overfitting, we validated the models using 158 images out of the 316 images in the evaluation set. The validation images are only used in such a procedure and in no case to evaluate the model nor in training. In training, we used early stopping with patience equal to 2,000. This method monitored the loss functions calculated on the validation subset, for which we used a batch size equal to 7.

E. EVALUATION STAGE

We perform a patch-based evaluation for all models [21]. To explain such an approach, let us consider a single model and a single evaluation image. The procedure begins with dividing the image into nine equally-sized overlapping patches $A, B, \ldots, I$ (Figure 4). Later, the model is fed with the patches, generating nine inferences. We infer the complete image from the specific contribution of each of the nine small inferences, as shown in Figure 5. Then, we integrate the entire map to carry out the counting in a predicted density map. In contrast, in a FIDT map, it is necessary to use a procedure for detecting local maxima [13].

We employed counting, localization, and reconstruction metrics in the evaluation. The first two types of metrics are responsible for quantifying the models’ performance in the tasks of counting and locating people in crowds. Specifically, localization metrics were used only for FIDT maps. Reconstruction metrics allow quantifying the DRA models’ performance in the reconstructive masking task.

Among the counting metrics, we have used the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE), which are given by (10) and (11), respectively,

$$\text{MAE} = \frac{1}{T} \sum_{i=1}^{T} |z_i - \hat{z}_i|,$$  \hspace{1cm} (10)

$$\text{RMSE} = \sqrt{\frac{1}{T} \sum_{i=1}^{T} (z_i - \hat{z}_i)^2}.$$  \hspace{1cm} (11)

In both metrics, $z_i$ and $\hat{z}_i$ are the target and estimated counts, and $T$ is the total number of evaluation images. The MAE metric measures the accuracy of the estimates, whereas RMSE measures their robustness.

We have employed the localization metrics Precision, Recall, and F1-Score, given by (12), (13), and (14),

$$\text{Precision} = \frac{TP}{TP + FP}.$$  \hspace{1cm} (12)

$$\text{Recall} = \frac{TP}{TP + FN}.$$  \hspace{1cm} (13)

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$  \hspace{1cm} (14)

where $TP$, $FP$, and $FN$ are the numbers of true positives, false positives, and false negatives, respectively. We obtained these last variables by comparing the predicted and ground truth locations using two decision thresholds: $\sigma_1 = 4$ and $\sigma_2 = 8$. The Precision metric measures the quality of successful predictions relative to the total number of times the model predicts the existence of an instance. Recall measures the number of correct predictions concerning the number of ground truth positives. The F1-Score metric measures the quality and quantity of correct predictions since it combines Precision and Recall.

The reconstruction metrics we have used are the RMSE, the SSIM, and the Feature Similarity Index Measure for color images (FSIMc) [36]. The FSIMc metric is defined by (15),

$$\text{FSIMc} = \frac{\sum_{x \in \Omega} S_L(x)[S_C(x)]^\rho PC_m(x)}{\sum_{x \in \Omega} PC_m(x)},$$  \hspace{1cm} (15)

where $\Omega$ is the set of all pixels in the image, $\rho = 0.03$ (based on [36]), $PC_m(x)$ is the maximum between the phase congruency of the prediction and the target, $S_L(x)$ is the similarity between the prediction and the target, whereas $S_C(x) = S_I(x)S_Q(x)$ is the chrominance similarity. The similarities
between the chromatic features \(S_I(x)\) and \(S_Q(x)\) are given by (16) and (17),

\[
S_I(x) = \frac{2I_1(x)I_2(x) + T_3}{I_1^2(x) + I_2^2(x) + T_3}, \quad (16)
\]

\[
S_Q(x) = \frac{2Q_1(x)Q_2(x) + T_4}{Q_1^2(x) + Q_2^2(x) + T_4}, \quad (17)
\]

where \(T_3 = T_4 = 200\) (based on [36]). \(I_1\) and \(Q_1\) are the color channels of the prediction, and \(I_2\) and \(Q_2\) are the chrominance information of the ground truth. For an RGB image, we obtained \(I\) and \(Q\) from (18),

\[
\begin{bmatrix}
Y \\
I \\
Q
\end{bmatrix} = \begin{bmatrix}
0.299 & 0.587 & 0.114 \\
0.596 & -0.274 & -0.322 \\
0.211 & -0.523 & 0.312
\end{bmatrix} \begin{bmatrix}
R \\
G \\
B
\end{bmatrix}, \quad (18)
\]

where \(Y\) corresponds to the luminance information [37].

For the reconstruction task, the RMSE metric measures the model error, whereas SSIM and FSIMc measure the similarities in structural and chromatic terms between the predictions and the ground truths of the reconstructive masks, respectively.

We evaluated the models using the 158 images of the evaluation subset set aside exclusively for testing. None of the four neural networks have seen these images in training or validation.

**IV. RESULTS**

The training of the DRA model for FIDT maps begins with Figure 6(a), which exposes the loss \(L_{1F}\) calculated over batches of the training and validation subsets. We trained only the first autoencoder from iteration 0 to 8,000, using the mentioned loss, to perform reconstructive masking. Then, the curve has a dead zone representing a pause in the training of the first autoencoder. The beginning of the dead zone activates the training of the second autoencoder employing the loss \(L_{2F}\). In Figure 6(b) we show the training curve of the second autoencoder, with the aim that it performs the generation of FIDT maps from the masked reconstructions generated by the first autoencoder. The independent training of the second autoencoder takes place from iteration 8,000 to iteration 12,000, after which we reactivated the training of the first autoencoder. Thus, we performed a joint training, where both losses operate from iteration 12,000 until the early stopping method is automatically activated. Joint training constitutes the final stage of adjustment of weights and biases to achieve a better adaptation between both parts of the network. However, as seen in Figure 6, most of the training of the autoencoders is done in the independent stages.

Similarly, the DRA model variant training for density maps starts with Figure 7(a). Identical to the training of the previous variant, we only trained the first autoencoder during the first 8,000 iterations. In such an interval, it is possible to observe the decrease of the \(L_{1D}\) loss as the iteration increases. Subsequently, we paused the training of the first autoencoder and activated the training of the second one for the successive 4,000 iterations, using the \(L_{2D}\) loss (Figure 7(b)). Finally, joint training is carried out from iteration 12,000 until the early stopping method is automatically activated. Again, the most effective training of this variant occurs during the individual training of the autoencoders since, according to the curves, the joint adjustment does not provide notable improvements in learning.

We display typical results from the DRA and SA models for FIDT maps in Figure 8. In particular, we present three images from the evaluation subset, along with the ground truth masked images, reconstructed masked images, ground truth FIDT maps, and the respective predictions of both neural networks. By comparing the predicted counts...

---

**TABLE 1.** Counting metrics for each variant of the DRA and SA models.

| Model     | MAE | RMSE |
|-----------|-----|------|
| FIDT map  |     |      |
| DRA       | 13.92 | 31.67 |
| SA        | 121.73 | 127.61 |
| Density map|     |      |
| DRA       | 19.87 | 32.59 |
| SA        | 9.50  | 15.57 |
FIGURE 6. DRA model training curves for FIDT maps.

FIGURE 7. DRA model training curves for density maps.

FIGURE 8. Sample results of DRA and SA models for FIDT maps.

TABLE 2. Localization metrics of the DRA and SA models for FIDT maps.

| Crowd Map | Model | Precision (%) | Recall (%) | F1-Score (%) |
|-----------|-------|---------------|------------|--------------|
|           |       | $\sigma_1 = 4$ | $\sigma_2 = 8$ | $\sigma_1 = 4$ | $\sigma_2 = 8$ | $\sigma_1 = 4$ | $\sigma_2 = 8$ |
| FIDT map  | DRA   | 63.11%        | 84.05%     | 37.61%       | 76.73%       | 60.24%       | 80.23%       |
|           | SA    | 27.11%        | 37.26%     | 54.56%       | 74.98%       | 36.22%       | 49.78%       |
from our model (fifth column in Figure 8) with the predicted counts of the widely-used SA model (sixth column in Figure 8), one can clearly see that the DRA model significantly improves the counting and location of people in crowds compared to the SA model. The location results of this latter model (last column) differ significantly from the ground truths (fourth column), as the predicted maps have many false positives and artifacts. This result was consistent across all the images used in the evaluation stage. In the case of the FIDT maps, the counts are directly related to the locations of the people. Despite the improvement over the SA model, the DRA model’s counting performance decreases for dense crowds positioned in the upper parts of the images (third sample image). However, despite this shortcoming, locating people remains competitive. Regarding the masked reconstructions made by the DRA model, it stands out that they are visually consistent with the ground truth masked images.

Figure 9 shows typical DRA and SA neural network results for density maps. For comparison purposes, we show the same images of the evaluation subset (see Figure 8). In addition, the count figures of the ground truth density maps differ slightly from those of the ground truth FIDT maps due to the different mechanisms for creating the crowd maps and the different schemes for obtaining the counts. The results for the networks that generate the density maps show that the counts estimated by the SA model are closer to the actual values than those generated by the DRA model when we use density maps. As expected, the masked reconstructions are similar to ground truths and those generated by the DRA variant for FIDT maps.

Comparing the results of the DRA model variants for FIDT and density maps, we observe that both have similar count estimates; however, the former significantly improves the location of people in dense crowds. In turn, the SA model for density maps provides better count estimates and fewer artifacts and false positives compared to its variant for FIDT maps.

We summarize the results of the counting metrics for all models in Table 1. The DRA model for FIDT maps has an MAE of 13.92 and an RMSE of 31.67, whereas the SA model for the same map type obtains 121.73 and 127.61. Thus, the metrics show the considerable improvement that constitutes using the devised methodology. On the other hand, the DRA model has a lower performance for counting compared to the SA model for density maps. Likewise, the DRA variant for FIDT maps has better accuracy and robustness than its variant for density maps.
The localization metrics of the DRA and SA models for FIDT maps are summarized in Table 2. The Precision, Recall, and F1-Score metrics are higher in the DRA model than in the SA model, implying a significant improvement in locating people in crowds, as shown by the sample results. For example, we can observe that the proposed DRA method is able to double the Precision.

Table 3 shows the reconstruction metrics achieved for both variants of the DRA model. As expected, both variants have highly similar RMSE, SSIM, and FSIMc values, since they have an identical learning mechanism for their first autoencoder (number of iterations, loss functions, among others). Based on these metrics, the reconstruction performance of the proposed masking is excellent.

Table 4 displays the performance of the DRA model variants and the SA model variant for density maps, along with the performance of various state-of-the-art models on the ShanghaiTech Part B dataset. It is possible to observe that our models have competitive performances. Although the SA (density maps) model performs better than the DRA (FIDT maps), it only focuses on counting, so it cannot obtain the individual location of people in dense crowds. On the other hand, unlike several state-of-the-art models, the architecture of our DRA model is simple since it corresponds to a single-column model composed of two cascaded autoencoders.

V. CONCLUSION

We conclude that our Dual Reconstructive Autoencoder (DRA) model for Focal Inverse Distance Transform (FIDT) maps improves localization and people counting compared to a single autoencoder architecture, which is widely used nowadays. These improvements were achieved due to the proposed architecture we have designed and the methodology of separating the tasks of detecting people and generating points representative of each head into independent autoencoders. Our neural network obtains crowd estimates similar to state-of-the-art models, accurate locations for all crowd density types, and excellent masked reconstructions. Despite this, our task-division approach failed to improve counting performance when density maps were used compared to the Single Autoencoder (SA) model. However, this last model only focuses on counting people, unable to obtain individual locations of people in dense crowds, whereas our model can generate both due to the dual architecture.

Among the future works, we will implement the DRA neural network for FIDT maps in the facilities of the campus of the Universidad de Concepción, Chile. We will deploy visible, near-infrared, and long-wave-infrared security cameras to characterize crowds using the proposed dual-autoencoder approach. Moreover, we will use a standalone 5G mobile communications network for centralized communication between the cameras and a deep learning server. In addition, we will generate an assembly of neural networks using the intermediate output of the DRA model to feed a facial expression recognition model. In this way, we will achieve a prototype of an intelligent, accurate, robust, fast, and effective Earthquake Early Warning System (EEWS) to help authorities make complex decisions at critical moments triggered by natural disasters.

As a discussion, we hope deep learning technologies can be adopted and implemented to contribute to society, improving each person’s general and particular well-being. Likewise, we expect that specialized organizations can regulate these technologies to prevent misuse, such as privacy violations. In this way, as has been done with other technologies, security standards and protocols could be defined at all stages of design and use, aiming at generating robust systems in code and ethics.

REFERENCES

[1] Government Chile. National Situation of COVID-19 in Chile. Accessed: Sep. 13, 2022. [Online]. Available: https://www.gob.cl/pasoapaso/cifrasoficiales/

[2] World Health Organization. WHO Coronavirus (COVID-19) Dashboard. Accessed: Sep. 13, 2022. [Online]. Available: https://covid19.who.int/

[3] WorldRiskReport. Bündnis Entwicklung Hilft and Ruhr University Bochum. Accessed: Sep. 13, 2022. [Online]. Available: https://www.ruhr-uni-bochum.de/en

[4] G. P. Hayes, E. K. Myers, J. W. Dewey, R. W. Briggs, P. S. Earle, H. M. Benz, G. M. Smoczyk, H. E. Flamine, W. D. Barnhart, R. D. Gold, and K. P. Furlong, “Tectonic summaries of magnitude 7 and greater Earthquakes from 2000 to 2015,” U.S. Geological Surv., Reston, VA, USA, Tech. Rep. 2016-1192, 2017, doi: 10.3133/ordc20161192.

[5] G. Gao, J. Gao, Q. Liu, Q. Wang, and Y. Wang, “CNN-based density estimation and crowd counting: A survey,” 2020, arXiv:2003.12783.

[6] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), vol. 1, San Diego, CA, USA, Dec. 2005, pp. 886–893, doi: 10.1109/CVPR.2005.177.

[7] W. M. Liu, C. T. Yang, and S. C. Chen, “Robust real-time face detection,” Int. J. Comput. Vis., vol. 57, no. 2, pp. 137–154, 2004.

[8] C. Wojek, S. Walk, and B. Schiele, “Multi-cue on-board pedestrian detection,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 794–801, doi: 10.1109/CVPR.2009.5206638.

[9] B. Benfold and I. Reid, “Stable multi-target tracking in real-time surveillance video,” in Proc. CVPR, Jun. 2011, pp. 3457–3464, doi: 10.1109/CVPR.2011.5995667.

[10] K. Chen, C. C. Loy, S. Gong, and T. Xiang, “Feature mining for localised crowd counting,” in Proc. Brit. Mach. Vis. Conf., vol. 1, 2012, pp. 3–20.

[11] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, “Multi-source multi-scale counting in extremely dense crowd images,” in Proc. Proc. Conf. Comput. Vis. Pattern Recognit., Jun. 2013, pp. 2547–2554, doi: 10.1109/CVPR.2013.329.

[12] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, “Single-image crowd counting via multi-column convolutional neural network,” in Proc. Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 589–597, doi: 10.1109/CVPR.2016.70.

[13] D. Liang, W. Xu, Y. Zhu, and Y. Zhou, “Focal inverse distance transform maps for crowd localization,” 2021, arXiv:2102.07925.

[14] 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. 1–10, doi: 10.1145/2733373.2806337.

[15] K. Han, W. Han, H. Yao, and L. Hou, “Image crowd counting using convolutional neural network and Markov random field,” J. Adv. Comput. Intell. Intell. Informat., vol. 21, no. 4, pp. 632–638, Jul. 2017.

[16] X. Liu, J. V. D. Weijer, and A. D. Bagdanov, “Leveraging unlabeled data for crowd counting by learning to rank,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 1–5, doi: 10.1109/CVPR.2018.00799.

[17] 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, doi: 10.1109/CVPR.2015.7298684.
[18] 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.

[19] M. Marsden, K. McGuinness, S. Little, and N. E. O’Conor, “Fully convolutional crowd counting on highly congested scenes,” in Proc. 12th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl., 2017, pp. 1–8, doi: 10.5220/000697300270033.

[20] L. Zhang, M. Shi, and Q. Chen, “Crowd counting via scale-adaptive convolutional neural network,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Lake Tahoe, NV, USA, Mar. 2018, pp. 1113–1121, doi: 10.1109/WACV.2018.00127.

[21] V. Kannadi Valloli and K. Mehta, “W-Net: Reinforced u-net for density map estimation,” 2019, arXiv:1903.11249.

[22] Z. Shi, L. Zhang, Y. Sun, and Y. Ye, “Multiscale multitask deep NetVLAD for crowd counting,” IEEE Trans. Ind. Informat., vol. 14, no. 11, pp. 4953–4962, Nov. 2018, doi: 10.1109/TII.2018.2852481.

[23] M. Zhao, J. Zhang, C. Zhang, and W. Zhang, “Leveraging heterogeneous auxiliary tasks to assist crowd counting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 12728–12737, doi: 10.1109/CVPR.2019.01302.

[24] Z. Shi, P. Mettes, and C. G. M. Snoek, “Counting with focus for free,” 2019, arXiv:1903.12206.

[25] B. Boominathan, S. S. S. Kruthivasan, and R. V. Babu, “CrowdNet: A deep convolutional network for dense crowd counting,” in Proc. 24th ACM Int. Conf. Multimedia, Oct. 2016, pp. 640–644, doi: 10.1145/2964284.2967300.

[26] V. A. Sindagi and V. M. Patel, “CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting,” in Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveillance (AVSS), Lecce, Italy, Aug. 2017, pp. 1–6, doi: 10.1109/AVSS.2017.8074991.

[27] D. B. Sam, S. Surya, and R. V. Babu, “Switching convolutional neural network for crowd counting,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 5744–5752.

[28] N. Ilyas, A. Shahzad, and K. Kim, “Convolutional-neural-network-based image crowd counting: Review, categorization, analysis, and performance evaluation,” Sensors, vol. 20, no. 1, p. 43, Dec. 2019, doi: 10.3390/s20010043.

[29] V. Nguyen and T. D. Ngo, “Single-image crowd counting: A comparative survey on deep learning-based approaches,” Int. J. Multimedia Inf. Retr., vol. 9, no. 2, pp. 63–80, Oct. 2019, doi: 10.1016/j.ijmir.2019.019.00181-y.

[30] B. Li, H. Huang, A. Zhang, P. Liu, and C. Liu, “Approaches on crowd counting and density estimation: A review,” Pattern Anal. Appl., vol. 24, no. 3, pp. 853–874, Aug. 2021, doi: 10.1007/s10044-021-00959-z.

[31] H. Bai, J. Mao, and S.-H. Gary Chan, “A survey on deep learning-based single image crowd counting: Network design, loss function and supervisory signal,” 2020, arXiv:2012.15685.

[32] M. Liu, J. Jiang, Z. Guo, Z. Wang, and Y. Liu, “Crowd counting with fully convolutional neural network,” in Proc. 25th IEEE Int. Conf. Image Process. (ICIP), Oct. 2018, pp. 953–957, doi: 10.1109/ICIP.2018.8451787.

[33] D. B. Sam, N. N. Sajjan, H. Maurya, and R. V. Babu, “Almost unsupervised learning for dense crowd counting,” in Proc. AAAI, 2019, pp. 8868–8875.

[34] Kaggle. ShanghaiTech Dataset. Accessed: Sep. 13, 2022. [Online]. Available: https://www.kaggle.com/thienn/shanghaitech

[35] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[36] L. Zhang, L. Zhang, X. Mou, and D. Zhang, “FSIM: A feature similarity index for image quality assessment,” IEEE Trans. Image Process., vol. 20, no. 8, pp. 2378–2386, Aug. 2011, doi: 10.1109/TIP.2011.2109730.

[37] C. Yang and S. H. Kwok, “Efficient gamut clipping for color image processing using LHS and YIQ,” Opt. Eng., vol. 42, n. 3, pp. 701–711, Mar. 2003, doi: 10.1117/1.1544479.

[38] L. Zeng, X. Xu, B. Cai, S. Qiu, and T. Zhang, “Multi-scale convolutional neural networks for crowd counting,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2017, pp. 465–469, doi: 10.1109/ICIP.2017.8296324.

[39] F. Lamas, K. Duguet, J. E. Pezoa, G. A. Montalva, S. N. Torres, and W. Meng, “Crowd detection and estimation for an earthquake early warning system using deep learning,” Proc. SPIE, vol. 12101, pp. 142–148, May 2022, doi: 10.1117/12.2622392.
SERGIO N. TORRES (Member, IEEE) was born in Concepción, Chile, in 1962. He received the B.S.E.E. (professional engineering) degree from the Universidad de Concepción, Concepción, in 1987, and the M.S.E.E. and Ph.D. degrees in engineering from the University of Dayton, Dayton, OH, USA, in 1997 and 2001, respectively. Since 1987, he has been a Faculty Member with the Department of Electrical Engineering, Universidad de Concepción. His research interests include digital signal and imaging processing applied to electro-optical systems, optical sensors, optical communications, and infrared imaging systems. He is a member of SPIE and OSA.

GONZALO A. MONTALVA received the B.S. degree in civil engineering and the master’s degree from the Pontifical Catholic University of Chile, in 2004. He has more than 15 years of experience in geotechnical engineering with a focus on multi-scale geotechnical characterization and modeling applied to seismic safety and the development of non-ergodic seismic demand models for subduction zones. During the professional career, he has led and/or participated in 11 research projects and disseminated his work in more than 34 scientific articles and/or books. He is currently with the Department of Civil Engineering, Faculty of Engineering, Universidad de Concepción. He was an Associate Researcher of the millennium nucleus seismic cycle along subduction zones, among other research projects. His work focuses on the effect of surface geology on the seismic response of engineering works. He is the author of predictive models of seismic intensities for Chile capable of being integrated into state-of-the-art probabilistic seismic hazard analysis (PSHA), one of them widely used in South America and other subduction zones. His research interests include seismic hazard analysis, basins seismic response, new methodologies for liquefaction analysis in subduction zones, rapid post-natural disaster characterization, and the broad area of geotechnical earthquake engineering. He was the past President of the Chilean Geotechnical Society.

WEIXIAO MENG (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees from the Harbin Institute of Technology (HIT), Harbin, China, in 1990, 1995, and 2000, respectively. From 1998 to 1999, he worked at NTT DoCoMo on adaptive array antenna and dynamic resource allocation for beyond 3G as a Senior Visiting Researcher. He is currently a Full Professor and the Vice Dean of the School of Electronics and Information Engineering, HIT. He has published four books and over 300 papers on journals and the international conferences. His research interests include broadband wireless communications, space-air-ground integrated networks, and wireless localization technologies. He is a fellow of the China Institute of Electronics, a member of the IEEE ComSoc and the China Institute of Communication. In 2005, he was honored provincial excellent returnee and selected into New Century Excellent Talents (NCET) plan by the Ministry of Education (MOE), China, in 2008, and the Distinguished Academic Leader of Harbin. Under his leading, Harbin Chapter won the IEEE ComSoc Chapter of the Year Award, the Asia-Pacific Region Chapter Achievement Award, and the Personal Member and Global Activities Contribution Award, in 2018. In 2021, he won the Best Paper Award of the IEEE SYSTEMS JOURNAL. He acted as the leading TPC Co-Chair of ChinaCom2011 and ChinaCom2016, the leading Services and Applications Track Co-Chair of IEEE WCNC2013, the Awards Co-Chair of IEEE ICC2015, the Wireless Networking Symposium Co-Chair of IEEE GLOBECOM2015, the AHSN Symposium Co-Chair of IEEE GLOBECOM2018, the leading Workshop Co-Chair of IEEE ICC2019 and IEEE ICNC2020, and the AHSN Symposium Co-Chair of IEEE ICC2020. He is the Chair of the IEEE Communications Society Harbin Chapter. He was an Editorial Board Member of Wiley’s WCPE journal from 2010 to 2017, an Area Editor of PHYCOM journal from 2014 to 2016, an Editorial Board Member of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS from 2014 to 2017, and has been an Editorial Board Member of the IEEE WIRELESS COMMUNICATIONS since 2015.