IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 11 October 2024, accepted 26 October 2024, date of publication 5 November 2024, date of current version 15 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3491830

RESEARCH ARTICLE

Improving Visual Pedestrian Attributes Discernment With Textual Reconstruction

YEJUN LEE[1], **JINAH KIM**[2], **JUNGCHAN CHO**[1], **(Member, IEEE), AND JHONGHYUN AN**[1]
[1]School of Computing, Gachon University, Sujeong-gu, Seongnam-si, Gyeonggi-do 13120, Republic of Korea
[2]School of Electrical Engineering, Korea University, Seongbuk-gu, Seoul 02841, Republic of Korea

Corresponding authors: Jungchan Cho (thinkai@gachon.ac.kr) and Jhonghyun An (jhonghyun@gachon.ac.kr)

This work was supported in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP), Korean Government [Ministry of Science and ICT (MSIT)], Development of High Performance Visual BigData Discovery Platform for Large-Scale Realtime Data Analysis, under Grant 2014-3-00123; in part by the Institute of Information Communications Technology Planning Evaluation (IITP), Korean Government (MSIT), Development of an Automotive Software Platform for Software-Defined-Vehicle (SDV) Integrated with an AI Framework Required for Intelligent Vehicles under Grant RS-2024-00397615; and in part by the Gachon University, South Korea under Grant GCU-202406160001.

ABSTRACT Recently, multi-modal research combining visual and textual information has emerged in Pedestrian Attribute Recognition (PAR). In this field, textual information has primarily been addressed through text modeling using tokenizers and textual encoders. However, separately learned visual and text encoders often find correlations between visual and textual features that may be insufficient at the human cognitive level. To address this issue, we drew inspiration from the way people describe pedestrian attributes and developed a method that mimics this cognitive process. This approach enhances visual encoders' ability to discriminate by generating sentences from images, masking important words, and then reconstructing them. Our method, which improves visual pedestrian attributes using textual information, demonstrates significant performance enhancements on the RAP and PA100k datasets, as well as on zero-shot datasets like RAP2zs and PETAzs, which do not overlap with the training and test sets. These improvements yield more meaningful results.

INDEX TERMS Visual–textual, reconstruction, human understanding, pedestrian attribute recognition.

I. INTRODUCTION

Pedestrian Attribute Recognition (PAR) is a crucial task in computer vision that focuses on the automatic identification of various individual attributes from images or videos. This technology is particularly valuable in surveillance scenarios, such as those involving CCTV cameras, where it can be used to identify or track individuals. PAR plays a significant role in diverse applications, including locating missing persons, assisting in rescue operations for accidents or falls, and enhancing security. As surveillance operations typically require substantial human effort, advancements in PAR technology are highly beneficial for improving the efficiency and accuracy of these tasks. Recent developments in deep neural networks across various fields have led to significant improvements in human understanding tasks, including Human Action Recognition [1], [2], [3], Pedestrian Detection and Tracking [4], [5], [6], and Facial Attributes Recognition [7], [8]. These advancements have also influenced several studies in PAR, which have primarily addressed the problem using only visual information obtained from camera modules, which are categorized into two main types: multi-label classifier-based methods and sequential module-based methods. Multi-label classifier-based methods define the problem as a multi-label classification task, aiming to train a multi-label classifier. To achieve this, these methods leverage pose estimation [9], [10], region proposals [11], [12], [13], [14], and visual attention mechanisms [15], [16], [17], [18]. These techniques are used to exploit the correspondence between pedestrian attributes and specific regions within an image. Sequential module-based methods approach PAR as a sequential prediction process, employing recurrent neural network architectures to explore attribute correlations. The

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

recurrent encoder-decoder framework has been integrated into PAR to reveal relationships among attributes, as well as hidden connections between attributes [19], [20], [21] and localized regions [22], [23]. However, sequential modules are significantly affected by predefined sequence orders and are challenging to optimize as the length of the attribute sequence increases. Additionally, these visual-only PAR methods treat pedestrian attributes as numerical or sequence labels, excluding important textual information about the attributes.

Conversely, visual-textual cross-modal approaches have also been applied to PAR recently [24]. By using pre-trained language models to embed pedestrian attributes and perform cross-modal fusion with visual features, these methods capture the correspondence between textual and visual information. However, this approach primarily focuses on extracting fusion information-region correlations and struggles to comprehend the overall context of an image and truly understand a 'pedestrian.' To deeply understand pedestrian attributes, it is essential not only to exploit each attribute and determine their correlations but also to grasp the overall context of the image. For example, consider how humans view images: they tend to perceive a photo as a whole context rather than describing it in parts. This implies that a single image can be described as a meaningful sentence. Such an approach allows for estimating correlations between attributes from a more human-like perspective, rather than analyzing each region and attribute separately.

In this study, we propose a context reconstruction module that helps visual extractors make more advanced estimates. The proposed module consists of two parts: a proposal-generation module and a text-reconstruction module. As shown in Fig. 1, our method employs the proposal-generation module to identify regions likely to contain important human attributes. These proposed regions serve as the basis for our text-reconstruction module, which simulates how humans describe photographs by applying weighted information to the identified areas. These modules enhance the accuracy of pedestrian attribute recognition, allowing visual encoders to understand human attributes in a more contextual manner. Additionally, the modules do not affect inference time and are not included in the final model parameters.

The main contributions of this study are summarized as follows.

1) We propose a context reconstruction module that enables visual encoders to achieve a more comprehensive understanding of image context through text reconstruction, resulting in improved pedestrian attribute discrimination compared to the baseline by filling in masked words.
2) We introduce a masking method where different weights are applied to each word in a sentence, with higher importance given to key attributes and less emphasis on verbs or investigative terms. This approach leads to a more selective and contextually meaningful form of masking, providing a reasonable method for word masking.
3) This module enhances attribute discrimination by using a visual feature enhancement method in conjunction with the context reconstruction module. It positively impacts classification performance when fused with text features processed through the Text-Image Self-attention module.
4) Since this module is used in a plug-in manner, it has the advantage of not increasing the model's parameters or inference time after training or learning is completed.

II. RELATED WORK

A. PEDESTRAIN ATTRIBUTE RECOGNITION(PAR)

Recently, the field of Pedestrian Attribute Recognition (PAR) has been divided into two main categories: visual-only PAR and visual-textual cross-modal PAR.

1) VISUAL-ONLY PAR

Classifier-Based Approaches: These approaches treat pedestrian attribute recognition as a multi-label classification task. DeepMAR [25] introduced a deep neural network tailored for this task and a weighted sigmoid cross-entropy loss to handle data imbalance effectively. Liu et al. [16] developed HydraPlus-Net by leveraging multidirectional attention modules to process multiscale features across various levels. Li et al. [9] used a spatial transformer network to convert estimated keypoints into clearly defined regions, integrating these regions with part-based features to enhance attribute recognition. Liu et al. [11] utilized a framework called the Localization-Guided Network (LGNet) to localize regions associated with different attributes. The View-Sensitive Pedestrian Attribute (VeSPA) [10] model features three view-specific units, each tailored to a specific view. This model helps generate view-dependent results and achieve final predictions through cross-view fusion, which can implicitly localize attributes. These techniques employ pose estimation and spatial transformation to establish links between attributes and regions of interest. However, classifier-based approaches often handle attributes uniformly and may miss deeper hidden correlations.

Sequential Approaches: This method uses recurrent neural networks to explore the context and relationships between attributes with the aim of improving performance. Wang et al. [19] introduced Joint Recurrent Learning (JRL), a model that segments images into horizontal strips for richer context encoding and provides an attention mechanism to enhance the representation capacity of context vectors. Zhao et al. [26] proposed Grouping Recurrent Learning (GRL), which organizes attributes by body region and leverages both intragroup semantic mutual exclusion and intergroup correlations using LSTM units. Recurrent convolutional and attention modules [20] focus on investigating contextual correlations with ConvLSTM units and highlight key regions.

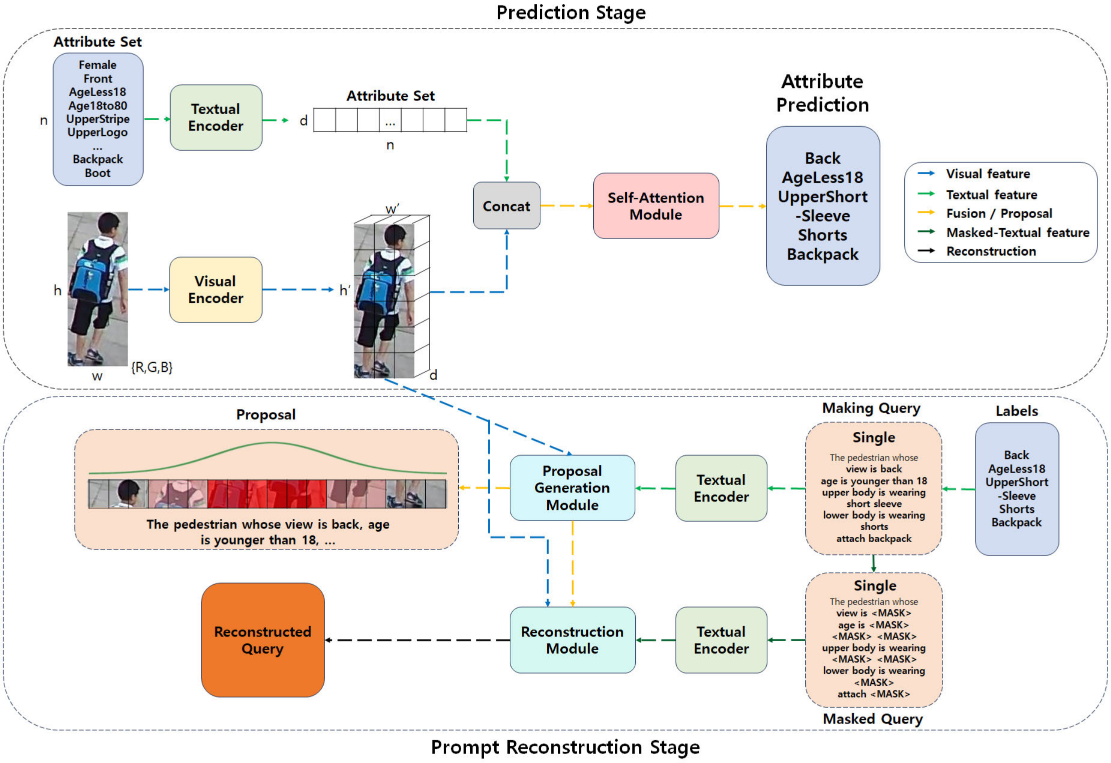


FIGURE 1. Overview of proposed method.

Visual-Semantic Graph Reasoning (VSGR) [27] creates a semantic graph that defines attribute relationships through directed edges. These sequential methods emphasize the importance of attribute correlations in recognition and explore these links using either recurrent or graph-based models. However, their semantic correlations are learned from scratch and depend on predefined prediction orders.

However, both methods rely solely on information from images and do not utilize the rich expressions or relationships inherent in natural language. A model that can accurately describe pedestrian attributes using natural language would provide a clearer understanding of these attributes. This is similar to how humans gain a better understanding of photographs when they describe them in words. This suggests that integrating natural language into PAR could enhance a model's ability to understand and represent pedestrian attributes.

2) FOUNDATION AND VISUAL-TEXTUAL CROSS-MODAL PAR

The visual-textual Baseline (VTB) [24] models attribute annotations as the textual modality, leveraging pre-trained textual encoders to explore prior textual correlations among attributes. Wang et al. [28] propose the Projector-Assisted Hierarchical Pretraining method (PATH) to enhance the learning of both coarse- and fine-grained human attributes at multiple levels of detail. Building on this, HumanBench has developed a versatile visual-only foundation pretraining model tailored for human-centric vision tasks, such as surveillance, autonomous driving, and the metaverse. Tang et al. [29] introduced Hulk, a foundational multi-modal model designed to efficiently address a broad range of human-centric tasks, including 2D and 3D vision, skeleton-based action recognition, and vision-language interactions, without the need for task-specific fine-tuning. By incorporating two universal heads—one for discrete data such as language and the other for continuous data such as spatial coordinates—Hulk is adept at translating between various modalities, enhancing its adaptability across diverse applications.

VTB [24] did not effectively utilize language information and the Hulk model experienced performance degradation with a reduction in the training of a single task. This suggests that good performance relies on the use of various tasks and datasets, necessitating a large amount of data. Such extensive data requirements can be impractical for tasks with limited datasets or experimental setups that require low computational resources. Therefore, we propose a vision-language cross-modal learning method that can be implemented with only a single dataset and leverage language information in depth.

III. PROPOSED METHOD

In this study, we propose a two-stage approach, as shown in Fig. 1. The first stage involves predicting pedestrian attributes by integrating textual and visual information, similar to existing pedestrian attribute recognition methods. The second stage generates visual regions within the pedestrian images, masks them according to natural language prompts, and reconstructs them. This reconstruction enhances the visual features necessary for accurate attribute prediction. To guide the learning process, we employed loss functions for both M pedestrian attributes and O-length natural language prompts.

A. ATTRIBUTE RECOGNITION MODULE

1) VISUAL ENCODER

The image encoder is tasked with compressing an image of a pedestrian into a visual embedding, which is a crucial step for the combination easily with pedestrian attribute predictions and text embeddings. $I = \{i_1, i_2, i_2, \ldots, i_N\}$, signifies the whole image, whereas N denotes the count of images within the full training dataset. $\mathrm{VE}\prime = \{\mathrm{L}_1, \mathrm{L}_2, \mathrm{L}_2, \ldots, \mathrm{L}_{K-1}\}$, refers to the result obtained after omitting the final layer from the Visual Encoder.

$$F_v = \mathbf{VE}'(\mathrm{concat}(e_p, \mathrm{Patch}(I))) \tag{1}$$

$F_v \in \mathbb{R}^{S\times C_v}$ is the visual feature calculated by a vision transformer, where S represents the number of patches extracted from the image, and C_v denotes the size of the image channel. The visual features F_v are obtained by passing the image through the image encoder. Patch$(\cdot)$ is a 2-dimensional convolution patch embedding operation. $\mathbf{VE}(\cdot)$ refers to the transformer encoder of the vision transformer, and e_p is a learnable patch token. concat$(\cdot)$ is the operation that merges two patches into one.

2) TEXTUAL ENCODER

Natural language prompts were preprocessed into textual features to fuse natural-language-based pedestrian attributes with visual features from a vision transformer, the natural language prompts are preprocessed into textual features.

$$F_t = \mathrm{TE}_{\mathrm{mean}}(\mathrm{annotate}(\mathbf{Attr})) \tag{2}$$

Here, $\mathbf{Attr} = \{a_1, a_2, \ldots, a_M\}$ represents the set of M attributes and a_i denotes the ith attribute. The function annotate$(\cdot)$ processes and tokenizes raw attributes (e.g., 'ub-TShirt' to 'upper t-shirt') using the Mpnet tokenizer [30]. The text encoder $\mathrm{TE}_{\mathrm{mean}}(\cdot)$ based on Mpnet averages the word embeddings of each attribute sentence to produce a single feature vector. The output $F_t \in \mathbb{R}^{M\times C_t}$ represents textual features, where C_t is the channel size of the text encoder.

3) VISUAL-TEXTUAL SELF-ATTENTION MODULE

The visual $\hat{F}_v$ and textual features $\hat{F}_t$ computed by the vision transformer and Mpnet text encoder differ in their dimensions and distributions, respectively. Their dimensions were integrated using linear layers to fuse them.

$$\hat{F}_v = \mathrm{L}_v(F_v + e_v^{\mathrm{embed}}), \quad \hat{F}_t = \mathrm{L}_t(F_t + e_t^{\mathrm{embed}}) \tag{3}$$

Here, $\hat{F}_v \in \mathbb{R}^{S\times D}$ and $\hat{F}_t \in \mathbb{R}^{M\times D}$ represent the dimensionally integrated features, where $\mathrm{L}_v(\cdot)$ and $\mathrm{L}_t(\cdot)$ are linear layers that standardize the dimensions to D, and e_v^{embed} and e_t^{embed} are learnable embeddings for visual and textual data, respectively. The fused feature map Z is obtained by concatenating $\hat{F}_v$ and $\hat{F}_t$:

$$Z = \mathrm{concat}(\hat{F}_t, \hat{F}_v), \quad Z \in \mathbb{R}^{(M+S)\times D} \tag{4}$$

This combined feature map Z is processed using multi-head self-attention and multi-layer perceptrons.

$$\hat{Z} = \mathrm{L}_K(Z) = \mathrm{MLP}_K(\mathrm{MSA}_K(Z)) \tag{5}$$

where $\hat{Z} \in \mathbb{R}^{(M+S)\times D}$ denotes the attention-enhanced feature map. For attribute prediction, a text-attention feature map $\hat{Z}$ is used, and a linear layer L_c is applied to predict the presence of each attribute.

$$p_i^c = \mathrm{FFN}_i(\hat{z}_i) = \mathrm{BN}(\mathrm{L}_c^i(\hat{z}_i)), \quad i = 1, 2, \ldots, M \tag{6}$$

The final prediction scores $P^c \in \mathbb{R}^M$ are used to determine the presence of each attribute.

B. PROMPT RECONSTRUCTION MODULE

$$\hat{T} = \mathrm{TE}_{\mathrm{each}}(T) \tag{7}$$

$T = \{t_1, t_2, t_3, \ldots, t_O\}$, $T \in \mathbb{R}^O$ is a prompts composed of O words that describe the image, similar to the examples of text prompts in Fig. 2, which has been processed through the tokenizer of Mpnet [30]. and $\hat{T} = \left\{\hat{t}_1, \hat{t}_2, \hat{t}_3, \ldots, \hat{t}_N\right\}$, $\hat{T} \in \mathbb{R}^{O\times C_t}$ is a series of textual embedding features for each tokenized word. $\mathrm{TE}_{\mathrm{each}}(\cdot)$ represents the textual encoder, which utilizes MPNet to generate embedding features for each word in the prompts corresponding to attributes as individual features. This method is employed because each word in the sentence plays a significant role in describing the image, making it important to consider all words individually.

$$H_v = \mathrm{D}_v(\hat{F}_v, \mathrm{E}_t(\hat{T})) \tag{8}$$

$H_v \in \mathbb{R}^{S\times D}$ is an image-focused feature map. $\mathrm{D}_v(A, B)$ is a transformer decoder for the visual feature map, where A is used as the query, and B is used as both the key and the value. $\mathrm{E}_t(\cdot)$ is a transformer encoder for text prompt features. $\hat{F}_v$ represents the visual features obtained in Section A. Detailed configurations of the transformer encoder and transformer decoder will be described in the Experiments section. D is set to the same dimension as the hidden size of both the encoder and decoder, as specified in Section A.

$$Q = \mathrm{L}_{\mathrm{gauss}}(H_v) \tag{9}$$

The gaussian proposal parameters $Q \in \mathbb{R}^{2\times K}$ can be represented as $Q = \{(c_1, w_1), (c_2, w_2), (c_3, w_3), \ldots, (c_k, w_k)\}$. K is the total number of gaussian proposals to be used, and w_k, c_k denote the width and center of each gaussian, respectively. $\mathrm{L}_{\mathrm{gauss}}(\cdot)$ is the linear layer used to generate gaussian proposal parameters, and the number of gaussian

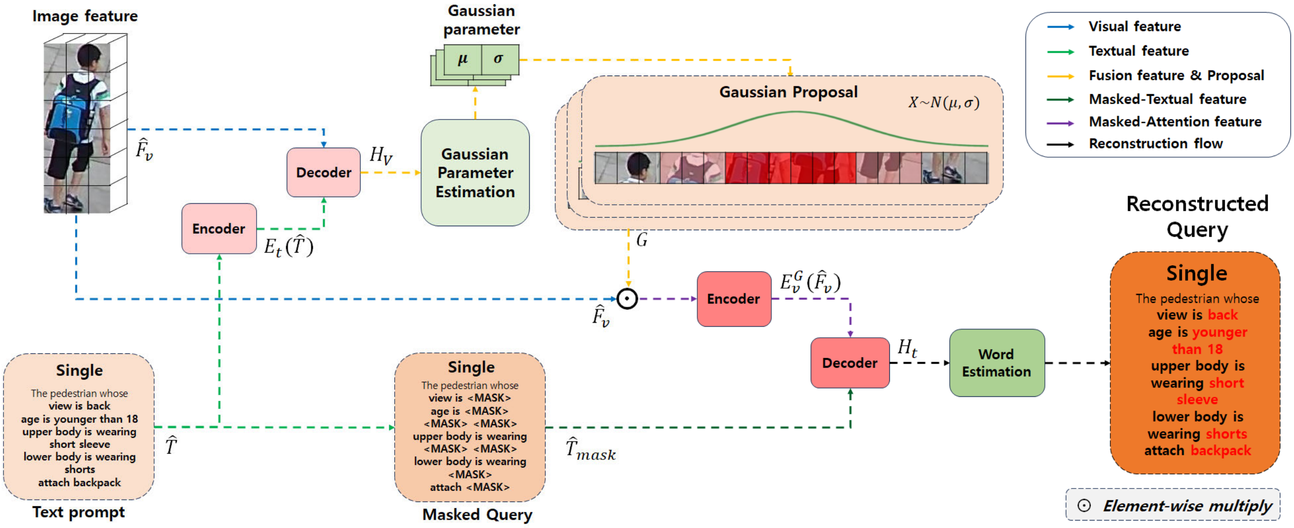


Image feature
$\hat{F}_v$
Decoder
H_V
Encoder
$E_t(\hat{T})$
Gaussian parameter
μ
σ
Gaussian Parameter Estimation
Gaussian Proposal
$X \sim N(\mu, \sigma)$
G
Visual feature
Textual feature
Fusion feature & Proposal
Masked-Textual feature
Masked-Attention feature
Reconstruction flow
$\hat{F}_v$
Encoder
$E_v^G(\hat{F}_v)$
Decoder
H_t
Word Estimation
Reconstructed Query
Single
The pedestrian whose view is back age is younger than 18 upper body is wearing short sleeve lower body is wearing shorts attach backpack
Single
The pedestrian whose view is back age is younger than 18 upper body is wearing short sleeve lower body is wearing shorts attach backpack
$\hat{T}$
Text prompt
Single
The pedestrian whose view is <MASK> age is <MASK> <MASK> <MASK> upper body is wearing <MASK> <MASK> lower body is wearing <MASK> attach <MASK>
Masked Query
$\hat{T}_{mask}$
⊙ Element-wise multiply

FIGURE 2. Overview of prompt reconstruction stage.

proposals generated varies depending on the hyperparameter K.

$$G_{ki} = \frac{1}{\sqrt{2\pi}(w_k/\sigma_k)} e^{(-\frac{(i/S-c_k)^2}{2(w_k/\sigma_k)^2})}, \qquad k = 1, 2, 3, \ldots, K; \quad i = 1, 2, 3, \ldots, S \tag{10}$$

The gaussian proposal G is used in the masked prompt reconstruction module. σ_k is a hyper-parameter that adjusts the width of the gaussian weights. The gaussian proposal $G \in \mathbb{R}^{K \times S}$ is a mask that applies weights to specific parts of the image patch, as depicted in Fig. 2.

1) MASKED PROMPT RECONSTRUCTION MODULE

To enhance the image encoder, a strategy was employed that utilizes region proposals generated from the proposed generation module and masked text sentence information. This involves randomly masking the original text sentences and then reconstructing them to align with the generated region proposals and masked text information.

$$\hat{T}_{\text{mask}} = \text{Mask}(\hat{T}, \alpha), \quad 0 \le \alpha \le 1 \tag{11}$$

Mask(A, B)is a function that randomly masks words in A at a ratio determined by B using weighted probabilities. Table 1 shows the weights for softmax-based masking function used in this experiment, where α was set to 0.5.

$$H_t^k = \text{D}_t^{G_k}(\hat{T}_{\text{mask}}, \text{E}_v^{G_k}(\hat{F}_v)) \tag{12}$$

$H_t^k \in \mathbb{R}^{O \times D}$ represents the masked text prompt attention feature map. $\text{D}_t(A, B)$ is a transformer decoder for the features of masked text prompts, where A is used as the query, and B is used as both the key and value. $\text{E}_v(\cdot)$ is the transformer encoder for the visual feature, and both D_t and E_v perform multi-head attention using the gaussian proposal G_k, generated in the proposal generation, as attention weights. Through this process, the visual encoder and the gaussian proposal generation module are enhanced to identify important parts within the visual feature for reconstructing the masked text prompt as shown as Fig. 2.

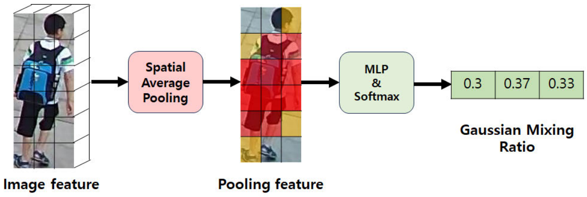


FIGURE 3. Estimation of gaussian mixing ratio through spatial average pooling.

$$R = \text{Softmax}(\text{MLP}_{\text{ratio}}(\text{SAP}(\hat{F}_v))) \tag{13}$$

As illustrated in Fig. 3, the ratio for gaussian proposal mixing, $R \in \mathbb{R}^K$, can be represented as $R = \{r_1, r_2, r_3, \ldots, r_K\}$. This is used to appropriately mix weights for K different gaussian proposals. SAP($\cdot$) refers to spatial average pooling, which pools S image patches containing spatial information. $\text{MLP}_{\text{ratio}}(\cdot)$ generates k mixing weights from the compressed visual feature and adjusts them through Softmax to ensure their sum equals 1.

$$H_t = \sum_{k=1}^{K} r_k {H_t}^k \tag{14}$$

The reconstructed mixing gaussian text prompt feature map $H_t \in \mathbb{R}^{O \times D}$ combines the results of the transformer computed by each gaussian proposal using the mixing weight R, thereby allowing the gaussian proposals to be mixed through the mixing ratio, enhancing the ability to better reconstruct masked text prompts by using visual region proposals of various sizes.

$$P^s = \mathrm{Softmax}(\mathrm{L}_r(H_t)) \tag{15}$$

The score for predicting each word of the prompt, $P^s \in \mathbb{R}^{O\times d}$, can be represented as follows, $P^s = \{p_1^s, p_2^s, p_3^s, \ldots, p_O^s\}$. Through the mixed gaussian proposal feature map H_t, and the linear layer, L_r, a reconstructed text prompt of total length O is generated. d is the total number of words in Mpnet's tokenizer, and softmax is computed over the d-dimension. Through this process, prompt reconstruction using Gaussian proposals is performed, enabling the visual encoder in the pedestrian attribute recognition module to estimate correlations between each attribute from a more human-like perspective, rather than estimating by area and attribute separately.

TABLE 1. Standard masking weight table.

Type	Word	Masking Weight
Period, joint	dot(.), comma(,)	-2
Position fixed words	A, Pedestrian, whose, is	1
Properties	age, lower, upper, body, attach, action, ...	2
Keywords	wearing, T-shirt, sweater, cotton, skirt, dress, jeans, ...	4

C. LOSS FUNCTION AND OPTIMIZATION

The entire dataset can be represented as $D_{\mathrm{train}} = (I_i, Y_i)_{i=1}^N$. For attribute prediction, a weighted binary cross-entropy loss function was employed. This loss function is commonly used in traditional attribute recognition to mitigate the distribution imbalance among pedestrian attributes [25]

$$L^c = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} w_j(\log(p_{ij}^c))) + (1-y_{ij})\log(1-p_{ij}^c)),$$
$$w_j = \begin{cases} e^{1-r_j}, & y_{ij} = 1 \\ e^{r_j}, & y_{ij} = 0 \end{cases} \tag{16}$$

Here, w_j represents the imbalance weight introduced to address the data-imbalance issue. r_j is the positive sample ratio of attribute Attr_j.

$$L^s = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{d}\left(\log\left(p_{ij}^s\right)\right) + \left(1 - \mathrm{onehot}\left(t_{ij}\right)\log\left(1-p_{ij}^s\right)\right) \tag{17}$$

The loss function for the text prompt reconstruction results generated in the prompt reconstruction module is computed using a cross-entropy loss function. This function is applied to the masked text prompt reconstruction results to enhance the learning of the image encoder. Cross-entropy loss is commonly used in the text prediction domain. Here, onehot$(\cdot)$ refers to the process of converting text prompt predictions into index numbers and then to a one-hot encoding format.

$$\mathrm{Loss} = L^c + \beta(L^s) \tag{18}$$

The overall loss function is the sum of the loss from the pedestrian attribute prediction module and the loss from the text prompt reconstruction module. This sum is then scaled by a predefined hyperparameter β, which was set to 1 for this experiment.

IV. EXPERIMENT

A. EXPERIMENT DETAILS

Patch sizes used in pedestrian attribute prediction module P_h and P_w, are set to 16. The number of layers for the image encoder is set to correspond to the Vision Transformer, with 8 blocks used for ViT-small and 12 blocks for ViT-base. The size of the image feature map, S, and the size of the text feature map, M, are both 768, obtained using pretrained models Vision Transformer [31] and Mpnet [30], respectively. The dimension size used for merging two different modalities, D, is set to 768. The parameter O, which determines the maximum length of text sentences, varies depending on the number of pedestrian attributes in the dataset. Each word in the sentence is not masked with the same probability but is determined by weights and softmax, as shown in Table 2. The proposed method was trained and evaluated using an NVIDIA Titan RTX GPU. The model was trained for 20 epochs on the RAP [32], RAP2 [33], PA100k [16], PETA [34], RAP2-zs [35], and PETA-zs [35] datasets. The initial learning rate was set to 2e-3 and adjusted using a cosine learning rate scheduler [36]. An SGD optimizer was employed, with a weight decay of 1e-4 and a momentum of 0.9. The batch size was set to 64, and the input images were resized to 256 × 128. Only the results from the pedestrian attribute prediction module were used for evaluation. During inference, the proposed model and VTB [24] had the same total number of parameters. For ViT-small, it is 49.35M, and for ViT-base, it is 87.15M. Similarly, the inference speed is the same, with 7.56G for ViT-small and 12.71G for ViT-base.

B. TECHNICAL DETAILS OF PROPOSED MODULES

The generation and reconstruction modules proposed in this study are based on the transformer architecture introduced by Vaswani et al. [37]. Two distinct transformers are employed for each module to effectively process visual and textual features. In the proposal generation module, a visual transformer handles the visual features, while a textual transformer processes the textual features. Both features are encoded using self-attention mechanisms within their respective transformers. These encoded features are then decoded using the decoder of the visual transformer. The resulting fused feature is used to derive gaussian parameters, which are employed to create a gaussian mask highlighting the important attributes of the pedestrian. This gaussian mask is referred to as a gaussian proposal. In the reconstruction module, both the visual and textual features, along with the previously generated Gaussian mask, are encoded through self-attention mechanisms within their respective transformers. Decoding is then performed by referencing only the restricted areas using the textual transformer's decoder and the gaussian mask. The masked fusion features generated through this decoding process are combined according to the ratio provided by the spatial average pooling Module. Finally, the combined features are used to predict words using the MLP. This visual feature enhancement method improves attribute discrimination by leveraging the proposed modules to enhance the visual backbone. This whole process is illustrated in Fig. 2.

C. DATASET AND EVALUATION METRIC

To evaluate the proposed model, we utilized six pedestrian attribute recognition datasets: RAP [32], PA100k [16], PETA [34], PETA-zs [35], RAP2-zs [35], and RAP2 [33]. The RAP dataset [32] consists of 41,585 images, with 33,268 used for training and 8,317 for testing. These images are annotated with 69 binary attributes and three multiclass attributes. However, only 51 attributes were used for training and evaluation in this study, as specified by the official protocol. The PA100k dataset [16] includes 100,000 images obtained from outdoor surveillance cameras. It is divided into 80,000 training images, 10,000 validation images, and 10,000 testing images, and is annotated with 26 pedestrian attributes. The PETA dataset [34] contains 19,000 images: 9,500 for training, 1,900 for validation, and 7,600 for testing. This dataset is annotated with 61 binary attributes and four multiclass attributes. For this study, only 35 attributes with a positive ratio greater than 5%, which are common in pedestrian attribute recognition, were used for training and evaluation. In addition to the commonly used datasets in pedestrian attribute recognition studies, this study employed three additional benchmark datasets: PETA-zs [35], RAP2-zs [35], and RAP2 [33]. RAP2 is an extended version of the RAP dataset [33] and consists of 84,928 images, divided into 67,943 training images and 16,985 testing images. Although there are 72 pedestrian attributes, only 54 were used according to the official protocol. Jia et al. [35] noted that the evaluation performance in the RAP2 and PETA datasets may be distorted because the same individuals appear in both the training and testing image splits. To address this issue, they proposed zero-shot datasets, PETA-zs and RAP2-zs, where the same individuals do not appear in both the training and testing images. These datasets are derived from the PETA dataset [34] and the RAP2 dataset [33].

For the evaluation, this study utilized one label-based metric and four instance-based metrics, totaling five evaluation metrics commonly used in pedestrian attribute recognition. The label-based metric employed was mean accuracy, which calculates the average accuracy for both positive and negative samples of each attribute. The instance-based metrics used were accuracy, precision, recall, and F1-score.

D. SINGLE-SIGMA

Table 2 categorizes the methods of pedestrian attribute recognition into four types. The first type consists of classifier-based methods including DeepMAR [25], HPNet [16], LGNet [11], VeSPA [10], PGDM [9], HDMTL [38], SSR [39], ALM [40], Rethinking [41], MT-CAS [42], PD-Net [43], JLAC [44], SSC [35], and DAFL [45]. The second type includes sequence-based methods, such as JRL [19], GRL [26], and RC/RA [20]. The third type is an image-text fusion-based method, represented by VTB [24]. The fourth type is the method proposed in this study, which involves image-text fusion and text–sentence reconstruction. Here, VTB* are the performance recorded under the same experimental conditions as 'VTR' using ViT-Base/16 and ViT-Small/16.

Table 3 and Table 4 present the results of experiments conducted with the image-text fusion-based method VTB [24] and the method proposed in this study, referred to as VTR. These experiments varied the size of the image encoder across different datasets. In this research, the hyperparameter K for gaussian proposal generation was fixed at 1, while the value of sigma varied during the experiments. For PETA**, where VTB did not provide results, the performance was replicated under the same conditions as those used in the VTR experiments. Table 2 shows that both classifier-based and sequence-based methods perform worse than the image-text fusion-based method VTB. Additionally, compared with the three methods mentioned above, the method involving image-text fusion and text sentence reconstruction demonstrated the best performance on the RAP dataset in terms of accuracy and F1-score, and it achieved the highest performance across all five metrics on the PA100k dataset. For the RAP dataset, the highest F1-score indicates that the method effectively addresses the data imbalance problem without biasing towards either precision or recall. The use of gaussian weights, generated with appropriate sigma values to restore masked text sentences, positively influenced the image encoder, thereby enhancing the performance of the pedestrian attribute prediction module. However, a drawback of this approach is the need for continuous experimentation with varying hyperparameters until satisfactory results are achieved.

E. MULTI-SIGMA MIXING

To address these limitations, a method was devised that uses the average of various gaussian proposals created with multiple sigma values. This approach is referred to as the multiple-only(mean) method in Table 5 and Table 6. The best-performing experiments from Table 3 and Table 4 are labeled as "best."

As shown in Fig. 4, the multiple sigma(mean) and multiple+mixing sigma(spatial average pooling) methods required only a single training session, whereas the best method required multiple training sessions. As indicated in Table 5 and Table 6, the best method outperformed the

TABLE 2. Performance with other models.

Method	Visual Encoder	RAP					PA100k				
		mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1
DeepMAR [25]	CaffeNet	73.79	62.02	74.92	76.21	75.56	72.70	70.39	82.24	80.42	81.32
HPNet [16]	Inception	76.12	65.39	77.33	78.79	78.05	74.21	72.19	82.97	82.09	82.53
LGNet [11]	Inception-V2	78.68	68.00	80.36	79.82	80.09	76.96	75.55	86.99	83.17	85.04
VeSPA [10]	Inception	77.70	67.35	79.51	79.67	79.59	-	-	-	-	-
PGDM [9]	CaffeNet	74.31	64.57	78.86	75.90	77.35	74.95	73.08	84.36	82.24	85.04
HDMTL [38]	ResNet50	74.43	67.09	83.13	76.00	79.41	-	-	-	-	-
SSR [39]	ResNet50	79.92	67.45	80.46	80.23	80.34	-	-	-	-	-
ALM [40]	BN-Inception	81.87	68.17	74.71	86.48	80.16	80.68	77.08	84.24	88.84	86.46
Rethinking [41]	ResNet50	78.48	67.17	82.84	76.28	78.94	79.38	78.56	89.41	84.78	86.55
MT-CAS [42]	ResNet34	-	-	-	-	-	77.20	78.09	88.46	84.86	86.62
PD-Net [43]	Inception-V3	-	-	-	-	-	80.40	78.80	87.50	86.91	87.20
JLAC [44]	ResNet50	83.69	69.15	79.31	82.40	80.82	82.31	79.47	87.45	87.77	87.61
SSCsoft [35]	ResNet50	82.83	68.16	74.74	**87.54**	80.27	81.70	78.85	85.80	88.92	86.89
DAFL [45]	ResNet50	**83.72**	68.18	77.41	83.39	80.29	83.54	80.13	87.01	89.19	88.09
JRL [19]	AlexNet	74.74	-	75.08	74.96	74.62	-	-	-	-	-
GRL [26]	Inception-V3	81.20	-	77.70	80.90	79.29	-	-	-	-	-
RC [20]	Inception-V3	78.47	-	82.67	76.65	79.54	-	-	-	-	-
RA [20]	Inception-V3	81.16	-	**79.45**	79.23	79.34	-	-	-	-	-
VTB* [24]	ViT-S/16	79.46	66.69	76.98	81.51	78.76	79.28	77.24	86.34	85.97	85.71
VTB [24]	ViT-B/16	82.67	69.44	78.28	84.39	80.84	83.72	80.89	87.88	89.30	88.21
VTR(ours)	ViT-S/16	79.86	66.99	76.91	82.08	79.00	79.33	77.75	86.59	86.42	86.07
VTR(ours)	ViT-B/16	83.04	**69.46**	78.05	84.72	**80.87**	**83.94**	**81.00**	**88.00**	**89.35**	**88.30**

TABLE 3. Performance comparison between the baseline and the single-sigma method.

Method	Visual Encoder	Sigma	RAP					PA100k					PETA**				
			mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1
VTB [24]	ViT-B/16	x	82.67	69.44	**78.28**	84.39	80.84	83.72	80.89	87.88	89.30	88.21	83.26	76.11	83.23	86.57	84.53
VTR(ours)	ViT-B/16	7	50.00	42.23	69.75	50.88	58.35	83.35	80.82	87.58	89.59	88.19	85.65	78.75	85.07	88.36	86.39
		9	83.04	**69.46**	78.05	**84.72**	**80.87**	83.53	80.84	87.72	89.40	88.18	**86.13**	**78.83**	**85.17**	**88.39**	**86.45**
		11	**83.05**	69.27	77.81	**84.72**	80.73	**83.94**	**81.00**	**88.00**	**89.35**	**88.30**	85.41	78.44	84.86	88.15	86.17
VTB* [24]	ViT-S/16	x	79.73	66.85	76.75	82.07	78.89	78.30	76.47	85.56	85.68	85.18	81.45	74.79	82.42	84.84	83.28
VTR(ours)	ViT-S/16	7	78.54	65.98	75.58	**82.18**	78.29	**79.33**	**77.75**	**86.59**	**86.42**	**86.07**	82.11	75.20	82.41	85.55	83.61
		9	**79.86**	**66.99**	**76.91**	82.08	**79.00**	78.12	76.75	86.24	85.42	85.37	77.43	70.62	79.04	82.06	80.12
		11	79.34	66.73	76.52	82.16	78.81	79.20	77.29	86.06	86.28	85.75	**82.77**	**75.72**	**82.76**	**86.04**	**84.04**

TABLE 4. Performance comparison between the baseline and the single-sigma method.

Method	Visual Encoder	Sigma	PETAzs					RAPzs					RAPv2				
			mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1
VTB* [24]	ViT-B/16	x	75.85	61.60	73.32	76.07	74.25	**76.87**	66.87	76.24	82.77	78.97	81.02	67.84	**76.50**	83.98	79.68
VTR(ours)	ViT-B/16	7	75.08	61.51	73.28	76.11	74.23	76.39	66.83	**76.37**	82.52	78.93	**81.07**	**67.89**	76.34	**84.27**	**79.72**
		9	75.75	61.37	73.22	75.57	73.94	76.10	66.72	76.26	82.49	78.86	79.54	66.75	75.64	83.35	78.90
		11	**76.40**	**62.74**	**74.21**	**76.96**	**75.14**	76.76	**66.98**	76.17	**83.02**	**79.05**	80.88	67.80	76.42	84.00	79.66
VTB* [24]	ViT-S/16	x	66.79	53.49	67.60	68.54	67.50	69.81	61.81	73.22	**78.28**	75.16	78.83	65.95	75.57	81.91	78.21
VTR(ours)	ViT-S/16	7	**68.88**	**55.24**	**68.74**	70.05	68.88	**70.93**	**62.19**	**74.17**	77.69	**75.39**	78.11	65.62	75.27	81.82	77.99
		9	68.83	54.75	68.69	69.44	68.53	69.11	61.00	73.49	76.48	74.45	**79.26**	66.19	75.59	**82.35**	**78.42**
		11	68.56	**55.24**	68.65	**70.23**	**68.93**	50.00	41.07	69.16	49.43	57.22	79.16	**66.20**	**75.76**	82.11	78.41

TABLE 5. Performance comparison between the baseline, single-sigma, multi-sigma, and mixing ratio multi-sigma methods.

Method	Visual Encoder	Sigma		RAP					PA100k					PETA**				
		Multiple	Mixing	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1
VTB	ViT-B/16	-	-	82.67	69.44	**78.28**	84.39	80.84	83.72	80.89	87.88	89.30	88.21	83.26	76.11	83.23	86.57	84.53
VTR(ours)	ViT-B/16			**83.04**	**69.46**	78.05	84.72	**80.87**	**83.94**	81.00	88.00	89.35	88.30	**86.13**	**78.83**	**85.17**	**88.39**	**86.45**
		✓		82.96	68.90	77.29	84.81	80.47	83.81	**81.33**	**88.32**	**89.85**	**88.51**	85.49	78.60	85.05	88.18	86.29
		✓	✓	82.99	69.18	77.60	**84.91**	80.69	83.17	81.21	88.04	89.67	88.47	85.82	78.74	**85.17**	88.27	86.39
VTB	ViT-S/16	-	-	79.73	66.85	76.75	82.07	78.89	78.30	76.47	85.56	85.68	85.18	81.45	74.79	82.42	84.84	83.28
VTR(ours)	ViT-S/16			**79.86**	**66.99**	**76.91**	82.08	**79.00**	79.33	77.75	86.59	86.42	86.07	82.77	75.72	82.76	86.04	84.04
		✓		78.75	66.54	76.57	81.90	78.69	79.21	**78.08**	**86.78**	**86.69**	**86.30**	79.04	72.22	80.47	83.06	81.38
		✓	✓	78.89	66.52	76.37	**82.09**	78.68	**79.36**	77.64	86.62	86.34	86.03	**82.83**	**75.86**	**83.05**	**85.77**	**84.06**

existing VTB [24] method in all 12 experiments, while the multiple sigma(mean) method demonstrated improvements in 7 out of 12 experiments with only one trial. This suggests that the mean method was effective in the majority of exper-

TABLE 6. Performance comparison between the baseline, single-Sigma, multi-Sigma, and mixing ratio multi-sigma methods.

Method	Visual Encoder	Sigma		PETAzs					RAPzs					RAPzs				
		Multiple	Mixing	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1
VTB	ViT-B/16	-	-	75.85	61.60	73.32	76.07	74.25	76.87	66.87	**76.24**	82.77	78.97	81.02	67.84	76.50	83.98	79.68
VTR(ours)	ViT-B/16			**76.40**	**62.74**	**74.21**	**76.96**	**75.14**	76.76	**66.98**	76.17	**83.02**	**79.05**	81.07	67.89	76.34	**84.27**	**79.72**
		✓		75.89	61.84	73.50	76.29	74.44	73.00	64.55	74.93	80.72	77.27	81.11	67.76	76.47	83.87	79.62
		✓	✓	75.61	61.75	73.76	76.08	74.46	**76.88**	66.64	76.14	82.52	78.81	**81.13**	**67.90**	**76.53**	84.01	**79.72**
VTB	ViT-S/16	-	-	66.79	53.49	67.60	68.54	67.50	69.81	61.81	73.22	**78.28**	75.16	78.83	65.95	75.27	81.91	78.21
VTR(ours)	ViT-S/16			68.56	55.24	68.65	70.23	68.93	70.93	62.19	**74.17**	77.69	75.39	**79.26**	66.19	75.59	**82.35**	78.42
		✓		68.97	55.46	69.17	70.02	69.10	71.07	**62.20**	73.86	78.04	**75.41**	78.56	**66.29**	**76.00**	81.99	**78.48**
		✓	✓	**69.51**	**55.97**	**69.70**	**70.59**	**69.65**	**71.71**	61.95	73.77	77.66	75.17	78.76	66.11	75.65	82.11	78.35

TABLE 7. Performance comparison between the baseline, single-sigma, multi-sigma, and mixing ratio multi-sigma methods.

Method	Visual Encoder	Sigma		Total Experiments(mean)				
		Multiple	Mixing	mA	Acc	Prec	Recall	F1
Baseline	ViT-S/16	-	-	75.82	66.56	76.80	80.22	78.03
Fixed Sigma(Best)	ViT-S/16			76.79(+0.97)	**67.35(+0.79)**	77.44(+0.64)	**80.80(+0.58)**	78.64(+0.61)
Multiple Sigma(Mean)		✓		75.93(+0.11)	66.80(+0.24)	77.14(+0.34)	80.28(+0.06)	78.23(+0.20)
Multiple+Mixing Sigma(SAP)		✓	✓	**76.84(+1.02)**	67.34(+0.78)	**77.53(+0.73)**	80.76(+0.54)	**78.66(+0.63)**

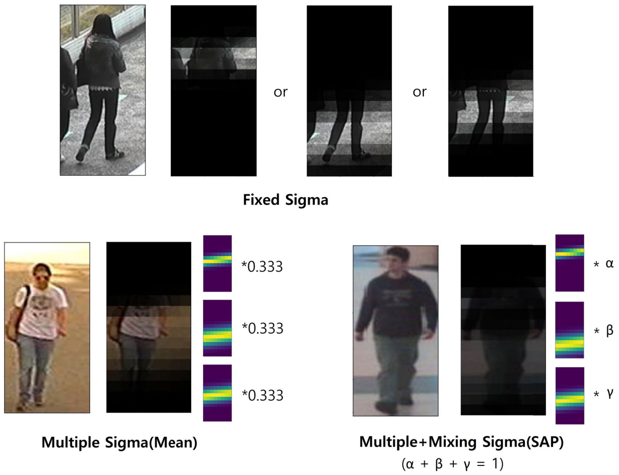


FIGURE 4. Comparing the fixed sigma, multiple sigma(Mean), and multiple+mixing sigma(SAP) methods.

iments without needing multiple training sessions. Using the mean method for gaussian mixing across all datasets may not adequately address data imbalances. Therefore, a spatial average pooling method was designed to enable the neural network to dynamically adjust the mixing ratio of gaussians as it learns. As observed in Fig. 4, the mean method mixes all the gaussian proposals at equal ratios, whereas the spatial average pooling method adjusts the mixing ratios dynamically through learning. This approach facilitates better reconstruction of masked text sentences, thereby providing more effective training information to the image encoder.

As the result as shown in Table 7, while the fixed sigma method achieves strong results. but in fixed sigma method, the model must be trained multiple times depending on the number of sigma values. In contrast, both the mean and SAP methods complete the process in just one train. The Mean method averages all sigma, but the SAP method uses a flexible mixing ratio via MLP and softmax, leading to more robust performance. This flexibility allows the SAP method to achieve comparable or even better results, particularly in all metric scores, demonstrating it as a more efficient and effective strategy.

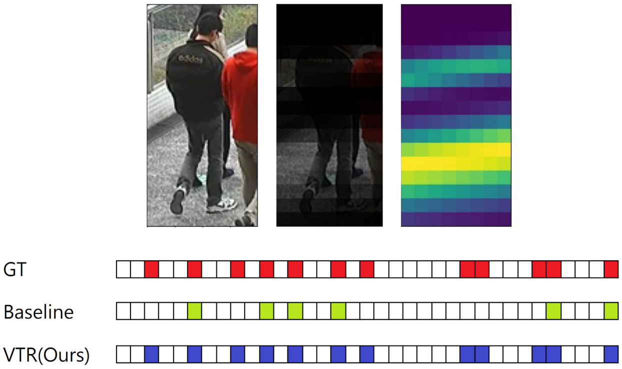


FIGURE 5. Comparison between baseline and ours(VTR).

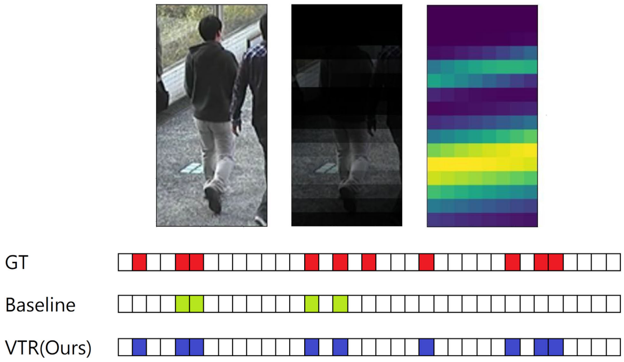


FIGURE 6. Comparison between baseline and ours(VTR).

Fig. 5 and Fig. 6 compare the results of previous studies with the proposed method in this research. The proposed generation module creates regions in the images, as shown on the far-right side of the figures, that are considered important

TABLE 8. Comparison of attribute-wise accuracy on PETA dataset.

Class	ageLess30	Male	lowerTrousers	Nothing	footShoes	footSneaker	carryNothing	carryMessengerBag	upperLogo	upperTshirt	upperShortSleeve	carryBackpack
Baseline	44.3	50.9	44.78	71.78	27.82	16.48	22.1	23.76	1.92	6.02	12.03	15.89
VTR(ours)	46.14	52.26	46.07	73.07	28.84	17.5	23.07	24.59	2.72	6.76	12.76	16.59
Improved Accuracy	1.84	1.36	1.29	1.29	1.02	1.02	0.97	0.83	0.8	0.74	0.73	0.7

TABLE 9. Comparison of attribute-wise accuracy on PETA dataset.

Class	carryPlasticBags	Sunglasses	upperThinStripes	Muffler	upperCasual	upperJacket	upperPlaid	lowerCasual	lowerShorts	ageLess60	lowerShortSkirt	lowerFormal
Baseline	5.72	0.9	0.31	7.34	83.26	3.86	1.9	84.11	2.28	7.8	3.5	10.8
VTR(ours)	6.35	1.4	0.75	7.69	83.52	4.07	2.11	84.32	2.48	7.98	3.68	10.98
Improved Accuracy	0.63	0.5	0.44	0.35	0.26	0.21	0.21	0.21	0.2	0.18	0.18	0.18

TABLE 10. Comparison of attribute-wise accuracy on PETA sataset.

Class	hairLong	ageLarge60	upperVNeck	footLeather	carryOther	upperFormal	footSandals	Hat	lowerJeans	upperOther	ageLess45
Baseline	21.46	5.47	0	25.35	13.9	10.73	1.06	8.71	26.05	38.35	26.82
VTR(ours)	21.6	5.61	0.14	25.48	13.9	10.72	0.97	8.6	25.67	37.72	26.03
Improved Accuracy	0.14	0.14	0.14	0.13	0	-0.01	-0.09	-0.11	-0.38	-0.63	-0.79

for inferring pedestrian attributes. Based on these proposals, weighted text and image attention were applied, leading to sentence generation using a prompt reconstruction module. This module is used not primarily for accurate sentence creation but to guide the image encoder by highlighting areas to focus on for predicting pedestrian attributes.

Table 8-Table 10 show the attribute-specific accuracy for the PETA dataset, comparing the baseline and the proposed method. The proposed method generally demonstrates improved results compared to existing methods. Notably, the attributes "Accessory-Nothing," "Male," and "AgeLess30" showed improvements of 1.29%, 1.36%, and 1.84%, respectively. These attributes require understanding not only of the pedestrian's immediate surroundings but also of the overall context. Furthermore, comprehending the meaning of natural language and its relevance is crucial. Therefore, the proposed method effectively enhances the image encoder by extracting the visual features of pedestrians through the text reconstruction process. The method proposed in this study offers a learning approach that improves pedestrian attribute prediction performance without increasing the computational load or the stored weight capacity of the model during inference. Additionally, this method has been found to positively impact smaller-sized image encoders, confirming its effectiveness in supporting learning.

F. ROBUSTNESS IN VARIED CONDITIONS

To test the robustness of our method, we compared its performance with the baseline method under various environmental conditions, including day, night, and rain. Since the original PAR dataset only included bright and clear conditions, we augmented the data to simulate different natural environments for a more comprehensive evaluation. Only the day condition was used during training, all three conditions were used for evaluation.

TABLE 11. Augmented images for various conditions.

(a) Night condition

(b) Rain condition

Table 11 shows examples of these augmented images. Panel (a) displays images simulating nighttime with reduced light, and panel (b) shows images representing rainy weather. We used these images to assess the methods' performance in different conditions. Table 12 lists the performance metrics for both VTB and our method (VTR) in these conditions. The results show that VTR consistently performs better than VTB. For instance, in the Day condition, VTR improves mA from 83.26% to 85.82%, in the Night condition from 74.93% to 81.79%, and in the Rain condition from 76.93% to 83.14%. Table 13 summarizes the average performance changes and percentage differences from Table 12. It shows that VTR has smaller performance drops compared to the baseline. For example, with the ViT-B/16 encoder, the baseline method's performance decreases by -10.1% in Night and -9.6% in Rain, while VTR's decreases are -4.36% and -7.53%, respectively. Similarly, with the ViT-S/16 encoder, the baseline method's performance drops by

TABLE 12. Comparison of model performance variations based on weather conditions and time.

Dataset	Method	Visual Encoder	Sigma		Day					Night					Rain				
			Multiple	Mixing	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1	mA	Acc	Prec	Recall	F1
PETA	VTB	ViT-B/16	-	-	83.26	76.11	83.23	86.57	84.53	74.93	65.22	75.54	79.32	76.93	75.27	65.80	76.38	79.44	77.41
	VTR(ours)	ViT-B/16	✓	✓	**85.82**	**78.74**	**85.17**	**88.27**	**86.39**	**81.79**	**73.95**	**81.80**	**85.24**	**83.14**	**79.43**	**70.22**	**79.05**	**83.13**	**80.64**
PETAzs	VTB	ViT-S/16	-	-	66.79	53.49	67.60	68.54	67.50	57.17	45.55	61.06	60.59	60.36	62.61	49.68	64.22	65.02	64.10
	VTR(ours)	ViT-S/16	✓	✓	**69.51**	**55.97**	**69.70**	**70.59**	**69.65**	**63.56**	**50.94**	**65.58**	**65.65**	**65.08**	**66.34**	**52.51**	**66.53**	**67.01**	**67.01**

TABLE 13. Comparison of model average performance variations based on weather conditions and time.

Method	Visual Encoder	Average Metric		
		Day	Night	Rain
VTB	ViT-B/16	82.74	74.39(-10.1%)	74.86(-9.6%)
VTR(ours)	ViT-B/16	**84.88**	**81.18**(-4.36%)	**78.49**(-7.53%)
VTB	ViT-S/16	64.78	56.95(-12.09%)	61.13(-5.63%)
VTR(ours)	ViT-S/16	**67.08**	**62.16**(-7.33%)	**63.88**(-4.78%)

−12.09% in Night and −5.63% in Rain, whereas VTR's drops are −7.33% and −4.78%, respectively.

V. CONCLUSION

In this study, we introduce a plug-in Proposal Generation Module and a Masked Prompt Reconstruction Module for pedestrian attribute recognition tasks. These modules enable the vision encoder to gain a deeper understanding of pedestrian attributes by 1) generating natural-language sentences, 2) masking arbitrary regions, and 3) reconstructing them. Our proposed method requires slightly more training time due to additional parameter tuning compared to existing methods. However, extensive testing across various datasets demonstrates that our method generally outperforms others. In experiments evaluating robustness under different day and weather conditions, our approach consistently produced more stable results. Importantly, these modules are removed after training, keeping the model's original parameters and computational cost unchanged. This makes the performance improvements particularly significant, as no additional datasets or tasks are needed. In conclusion, our method not only enhances the accuracy and efficiency of pedestrian attribute recognition but also ensures robustness across varying conditions, making it an effective solution in real-world applications.

REFERENCES

[1] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Berlin, Germany: Springer, 2018, pp. 305–321.

[2] M. Kalfaoglu, S. Kalkan, and A. A. Alatan, "Late temporal modeling in 3D CNN architectures with BERT for action recognition," in *Proc. Eur. Conf. Comput. Vis.*, Glasgow, U.K. Cham, Switzerland: Springer, 2020, pp. 731–747.

[3] M. Fayyaz, E. Bahrami, A. Diba, M. Noroozi, E. Adeli, L. Van Gool, and J. Gall, "3D CNNs with adaptive temporal feature resolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4729–4738.

[4] X. Zhang, S. Cao, and C. Chen, "Scale-aware hierarchical detection network for pedestrian detection," *IEEE Access*, vol. 8, pp. 94429–94439, 2020.

[5] W.-Y. Hsu and W.-Y. Lin, "Adaptive fusion of multi-scale Yolo for pedestrian detection," *IEEE Access*, vol. 9, pp. 110063–110073, 2021.

[6] X. Yang and Q. Liu, "Scale-sensitive feature reassembly network for pedestrian detection," *Sensors*, vol. 21, no. 12, p. 4189, Jun. 2021.

[7] B. A. Kumar and M. Bansal, "Face mask detection on photo and real-time video images using caffe-MobileNetV2 transfer learning," *Appl. Sci.*, vol. 13, no. 2, p. 935, Jan. 2023.

[8] B. A. Kumar and N. K. Misra, "Masked face age and gender identification using CAFFE-modified MobileNetV2 on photo and real-time video images by transfer learning and deep learning techniques," *Expert Syst. Appl.*, vol. 246, Jul. 2024, Art. no. 123179.

[9] D. Li, X. Chen, Z. Zhang, and K. Huang, "Pose guided deep model for pedestrian attribute recognition in surveillance scenarios," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2018, pp. 1–6.

[10] M. S. Saquib, A. Schumann, Y. Wang, and R. Stiefelhagen, "Deep view-sensitive pedestrian attribute inference in an end-to-end model," in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2017, pp. 134.1–134.13.

[10] M. S. Saquib, A. Schumann, Y. Wang, and R. Stiefelhagen, "Deep view-sensitive pedestrian attribute inference in an end-to-end model," in Proc. Brit. Mach. Vision Conf. (BMVC), Sep. 2017, pp. 134.1–134.13.

[11] P. Liu, X. Liu, J. Yan, and J. Shao, "Localization guided learning for pedestrian attribute recognition," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2018, p. 142.

[12] H. An, H. Fan, K. Deng, and H.-M. Hu, "Part-guided network for pedestrian attribute recognition," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2019, pp. 1–4.

[13] K. Han, Y. Wang, H. Shu, C. Liu, C. Xu, and C. Xu, "Attribute aware pooling for pedestrian attribute recognition," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 2456–2462.

[14] D. Gao, Z. Wu, and W. Zhang, "Safe-net: Solid and abstract feature extraction network for pedestrian attribute recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 1655–1659.

[15] H. An, H.-M. Hu, Y. Guo, Q. Zhou, and B. Li, "Hierarchical reasoning network for pedestrian attribute recognition," *IEEE Trans. Multimedia*, vol. 23, pp. 268–280, 2021.

[16] X. Liu, H. Zhao, M. Tian, L. Sheng, J. Shao, S. Yi, J. Yan, and X. Wang, "HydraPlus-net: Attentive deep features for pedestrian analysis," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 350–359.

[17] N. Sarafianos, X. Xu, and I. A. Kakadiaris, "Deep imbalanced attribute classification using visual attention aggregation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 680–697.

[18] S. Zhang, Z. Song, X. Cao, H. Zhang, and J. Zhou, "Task-aware attention model for clothing attribute prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 4, pp. 1051–1064, Apr. 2020.

[19] J. Wang, X. Zhu, S. Gong, and W. Li, "Attribute recognition by joint recurrent learning of context and correlation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 531–540.

[20] X. Zhao, L. Sang, G. Ding, J. Han, N. Di, and C. Yan, "Recurrent attention model for pedestrian attribute recognition," in *Proc. AAAI Conf. Artif. Intell.*, Jul. 2019, vol. 33, no. 1, pp. 9275–9282.

[21] J. Wu, H. Liu, J. Jiang, M. Qi, B. Ren, X. Li, and Y. Wang, "Person attribute recognition by sequence contextual relation learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3398–3412, Oct. 2020.

[22] J. Zhang, Q. Wu, C. Shen, J. Zhang, and J. Lu, "Multilabel image classification with regional latent semantic dependencies," *IEEE Trans. Multimedia*, vol. 20, no. 10, pp. 2801–2813, Oct. 2018.

[23] Q. Li, X. Zhao, R. He, and K. Huang, "Recurrent prediction with spatio-temporal attention for crowd attribute recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 7, pp. 2167–2177, Jul. 2020.

[24] X. Cheng, M. Jia, Q. Wang, and J. Zhang, "A simple visual-textual baseline for pedestrian attribute recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 10, pp. 6994–7004, Oct. 2022.

[25] D. Li, X. Chen, and K. Huang, "Multi-attribute learning for pedestrian attribute recognition in surveillance scenarios," in *Proc. 3rd IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nov. 2015, pp. 111–115.

[26] X. Zhao, L. Sang, G. Ding, Y. Guo, and X. Jin, "Grouping attribute recognition for pedestrian with joint recurrent learning," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3177–3183.

[27] Q. Li, X. Zhao, R. He, and K. Huang, "Visual-semantic graph reasoning for pedestrian attribute recognition," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 33, 2019, pp. 8634–8641.

[28] Y. Wang, Y. Wu, S. Tang, W. He, X. Guo, F. Zhu, L. Bai, R. Zhao, J. Wu, T. He, and W. Ouyang, "Hulk: A universal knowledge translator for human-centric tasks," 2023, *arXiv:2312.01697*.

[29] S. Tang, C. Chen, Q. Xie, M. Chen, Y. Wang, Y. Ci, L. Bai, F. Zhu, H. Yang, L. Yi, R. Zhao, and W. Ouyang, "HumanBench: Towards general human-centric perception with projector assisted pretraining," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21970–21982.

[30] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "MPNet: Masked and permuted pre-training for language understanding," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, Dec. 2020, pp. 16857–16867.

[31] A. Dosovitskiy, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–12.

[32] D. Li, Z. Zhang, X. Chen, H. Ling, and K. Huang, "A richly annotated dataset for pedestrian attribute recognition," 2016, *arXiv:1603.07054*.

[33] D. Li, Z. Zhang, X. Chen, and K. Huang, "A richly annotated pedestrian dataset for person retrieval in real surveillance scenarios," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1575–1590, Apr. 2019.

[34] Y. Deng, P. Luo, C. C. Loy, and X. Tang, "Pedestrian attribute recognition at far distance," in *Proc. 22nd ACM Int. Conf. Multimedia*, Nov. 2014, pp. 789–792.

[35] J. Jia, H. Huang, X. Chen, and K. Huang, "Rethinking of pedestrian attribute recognition: A reliable evaluation under zero-shot pedestrian identity setting," 2021, *arXiv:2107.03576*.

[36] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," 2016, *arXiv:1608.03983*.

[37] A. Vaswani, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–12.

[38] L. Gao, D. Huang, Y. Guo, and Y. Wang, "Pedestrian attribute recognition via hierarchical multi-task learning and relationship attention," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1340–1348.

[39] L. Li, Y. Dong, F. Xiong, and H. Bai, "Spatial and semantic relations for pedestrian attribute recognition," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2019, pp. 1–4.

[40] C. Tang, L. Sheng, Z.-X. Zhang, and X. Hu, "Improving pedestrian attribute recognition with weakly-supervised multi-scale attribute-specific localization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4997–5006.

[41] J. Jia, H. Huang, W. Yang, X. Chen, and K. Huang, "Rethinking of pedestrian attribute recognition: Realistic datasets with efficient method," 2020, *arXiv:2005.11909*.

[42] H. Zeng, H. Ai, Z. Zhuang, and L. Chen, "Multi-task learning via co-attentive sharing for pedestrian attribute recognition," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2020, pp. 1–6.

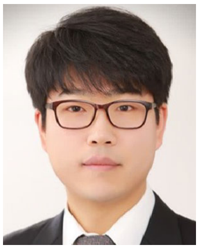
[43] Y. Liu, M. Tian, J. Hou, S. Yi, and Z. Lin, "Pentadent-net: Pedestrian attribute recognition with distance refinement and correlation mining," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 2211–2215.

[44] Z. Tan, Y. Yang, J. Wan, G. Guo, and S. Li, "Relation-aware pedestrian attribute recognition with graph convolutional networks," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2020, vol. 34, no. 7, pp. 12055–12062.

[45] N. Jia, N. Gao, F. He, X. Chen, and K. Huang, "Learning disentangled attribute representations for robust pedestrian attribute recognition," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2022, pp. 1–10.

YEJUN LEE received the B.S. degree in artificial intelligence and software from Gachon University, Seongnam, South Korea, in 2024, where he is currently pursuing the M.S. degree with the School of Artificial Intelligence and Software. His research interests include deep learning and computer vision.

JINAH KIM received the B.S. and M.S. degrees in artificial intelligence and software from Gachon University, Seongnam, South Korea. She is currently pursuing the Ph.D. degree in electrical engineering with Korea University, Seoul, South Korea. Her research interests include deep learning and computer vision.



JUNGCHAN CHO (Member, IEEE) received the B.S. degree from the School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, South Korea, in 2010, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Seoul National University, Seoul, in 2016. From 2016 to 2019, he worked as a Senior Software Engineer at Samsung Electronics. He is currently working as an Associate Professor with the School of Computing, Gachon University, Seongnam, South Korea. His research interests include deep learning, computer vision, and machine learning.

JHONGHYUN AN received the B.S. degree in electrical and electronic engineering and the Ph.D. degree in electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 2013 and 2020, respectively. From 2020 to 2022, he was a Senior Researcher at the Agency for Defense Development, Yuseong-gu, Daejeon, Republic of Korea. Since 2022, he has been a Faculty Member with the School of Computing, AI, and Software Majors, Gachon University, where he is currently working as an Assistant Professor. His current research interests include filtering theory, machine learning, deep learning, computer vision, and their applications in advanced driver assistance systems, computational intelligence, laser scanner-based recognition systems, target detection, target tracking, and unmanned ground vehicles.