Parameter-Efficient Person Re-identification in the 3D Space

Zhedong Zheng, Xiaohan Wang, Nenggan Zheng, and Yi Yang, *Senior Member, IEEE*

***Abstract*—People live in a 3D world. However, existing works on person re-identification (re-id) mostly consider the semantic representation learning in a 2D space, intrinsically limiting the understanding of people. In this work, we address this limitation by exploring the prior knowledge of the 3D body structure. Specifically, we project 2D images to a 3D space and introduce a novel parameter-efficient Omni-scale Graph Network (OG-Net) to learn the pedestrian representation directly from 3D point clouds. OG-Net effectively exploits the local information provided by sparse 3D points and takes advantage of the structure and appearance information in a coherent manner. With the help of 3D geometry information, we can learn a new type of deep re-id feature free from noisy variants, such as scale and viewpoint. To our knowledge, we are among the first attempts to conduct person re-identification in the 3D space. We demonstrate through extensive experiments that the proposed method (1) eases the matching difficulty in the traditional 2D space, (2) exploits the complementary information of 2D appearance and 3D structure, (3) achieves competitive results with limited parameters on four large-scale person re-id datasets, and (4) has good scalability to unseen datasets. Our code, models and generated 3D human data are publicly available at https://github.com/layumi/person-reid-3d.**

***Index Terms*—Person re-identification, 3D human representation, Image retrieval, Point cloud, Graph convolutional networks.**

I. Introduction

PERSON re-identification is usually regarded as an image retrieval problem of spotting the person in non-overlapping cameras [1]–[6]. Due to the rising demand of public safety and the fast development of camera network, person re-id has received increasing interests. These studies aim to save the human resource and efficiently find the person of interest, *e.g.*, lost child in the airport, from thousands of candidate images. In recent years, the advance of person re-id is mainly due to two factors: 1) the availability of large-scale datasets and 2) the deeply-learned person representation. On one hand, deeply-learned models are usually data-hungry. The large-scale datasets [7]–[10] facilitate the data-driven approaches. On the other hand, the development of Convolutional Neural Network (CNN) also provides the technical breakthrough of the pedestrian representation learning. Many efforts have been paid to improve the CNN-based model capability [11]–[14]. Recently, some researchers and companies also claim that the model can surpass the human performance [15].

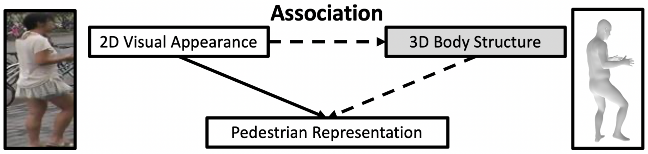


Fig. 1. Our brain generally associates the 2D appearance with prior knowledge of the 3D body shape. In this work, we intend to simulate this process and explore robust pedestrian representation with a lightweight model. (Dash arrows are missing in prevailing re-id methods.)

However, one inherent problem still remains: does the model really understand the person? People live in a 3D world. In contrast, we notice that most prevailing person re-id methods ignore the prior knowledge that human is a 3D non-rigid object, and only focus on learning the representation in 2D space. Although some pioneering works [16], [17] consider the 3D human structure, the pedestrian representation is still learned from the projected 2D images. For instance, one of the existing works, PersonX [17], has applied the game engine to build 3D person models. However, representation learning is conducted in the 2D space by projecting the 3D model back to 2D images. This line of works is effective in data augmentation but might be sub-optimal in representation learning. It is because the 2D data space intrinsically limits the model to understand the 3D geometry information of the person.

Inspired by the human ability of associating the 2D appearance with the 3D geometry structure (see Figure 1), we argue that the key to learning an effective and scalable person representation is to consider the complementary information of 2D human appearance and 3D geometry structure. With the prior knowledge of 3D human geometry information, we could learn a depth-aware model, thus making the representation robust to real-world scenarios. As shown in Figure 2, we map the visible surface to the human mesh, and make the person free from the 2D space. The intuition is that after mapping to the 3D space, the appearance information is correlated/aligned with the human structure. Without the need to worry about the part matching from two different viewpoints, the 3D data structure eases the matching difficulty in nature. The model could concentrate on learning the identity-related features, and dealing with the other intra-class variants, such as illumination conditions.

To fully take advantage of the 3D structure and 2D appearance, we propose a novel Omni-scale Graph Network for person re-id in the 3D space, called OG-Net. OG-Net is

Zhedong Zheng is with Sea-NExT Joint Lab, School of Computing, National University of Singapore, Singapore 118404. E-mail: zdzheng@nus.edu.sg (This research is supported by the Sea-NExT Joint Lab.)

Xiaohan Wang, Nenggan Zheng and Yi Yang are with the School of Computer Science, Zhejiang University, Hangzhou 310027, China. E-mail: xiaohan.wang@zju.edu.cn, zng@cs.zju.edu.cn, yangyics@zju.edu.cn.

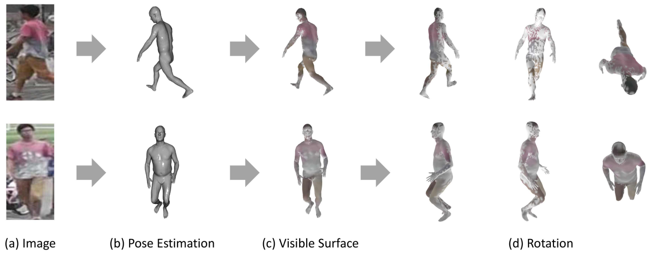


Fig. 2. Person is a 3D non-rigid object. In this work, we conduct the person re-identification in the 3D space, and learn a new type of robust re-id feature. Given one 2D image **(a)**, we first **(b)** estimate the 3D pose via the off-the-shelf model [18], followed by **(c)** mapping the RGB color of visible surfaces to corresponding points. The invisible parts are made transparent for visualization. **(d) The appearance information is aligned with the human structure. We make the person free from the 2D space, and thus ease the matching difficulty.**

a parameter-efficient model based on graph neural network (GNN) to communicate between the discrete cloud points of arbitrary locations. Given the 3D point cloud and the corresponding color information, OG-Net predicts the person identity and outputs the robust human representation for subsequent matching. Following the spirit of the conventional convolutional neural network (CNN), we utilize 3D points to build the location topology, and deploy the corresponding RGB color to extract appearance information. In particular, we propose Omni-scale module to aggregate the feature from multiple 3D receptive fields, which leverages multi-scale information in 3D data. Even though the basic OG-Net only consists of four Omni-scale modules, it has achieved competitive performance on four person re-id datasets.

Contribution. Our contributions are as follows. (1) We study person re-identification in the 3D space - a realistic scenario which could better reflect the nature of the 3D non-rigid human. To our knowledge, this work is among the early attempts to address this problem. (2) We propose a novel Omni-scale Graph Network to learn the feature from both human appearance and 3D geometry structure in a coherent manner. OG-Net leverages discrete 3D points to capture the multi-scale identity information. (3) Extensive experiments on four person re-id benchmarks show the proposed method could achieve competitive performance with limited parameters. A more realistic transfer learning setting is also studied in this paper. We observe that OG-Net has good scalability to the unseen person re-id dataset.

II. Related work

A. Semantic Space for Person Re-id

Recent years, convolutional neural network (CNN) models have been explored to map the pedestrian inputs, *e.g.*, images, into one shared semantic space, where the data of the same identity is close and the data of different identities is apart from each other [2], [19]. Different optimization objectives have been studied. For instance, the contrastive loss is widely-used to discriminate different identities [4], [20], [21], while the identification loss deploys the identity classification as the pretext task [3], [22], [23]. To simultaneously minimize the intra-class difference and maximize the inter-class gap, the triplet loss with different hard sampling strategies are also widely-studied [24]–[28]. Xiao *et al.* [29] propose the online instance matching loss to view the unlabeled data as negative samples, while Zheng *et al.* [9] design one label smooth loss to take advantage of synthetic data. Besides, several works [30]–[34] utilize person attributes, *e.g.*, gender, to help the model learning intermediate features. Some works also explore the post-processing approaches to further build the relation between the instances [35], [36]. This line of works is orthogonal to our work - any semantic spaces or optimization objectives can be used in our work and better ones can benefit our approach. In this work, we do not intend to pursue the best semantic space, but focus on verifying the effectiveness of the 3D space and the proposed OG-Net. We, therefore, deploy the basic identification loss for a fair comparison.

B. Part Matching for Person Re-id

To obtain the discriminative pedestrian representation, one line of research works resorts to mining local patterns, such as bodies, legs and arms, on 2D image inputs. The part matching is usually conducted on two different levels, *i.e.*, the pixel level [37]–[39] and the feature level [40]–[43]. The pixel-level part matching directly transforms the input image to one unified form. For instance, Su *et al.* [37] and Zheng *et al.* [39] deploy the off-the-shelf pose estimator [44] to predict the human key points, followed by cropping and resizing body parts for representation learning. Similarly, Zhang *et al.* [38] utilize the semantic segmentation predictor to crop and align body parts densely. Instead of cropping body parts, Saquib *et al.* [45] concatenate the RGB input with key point heatmap as input, and let model to learn the part attention by itself. In contrast, another line of works align the parts coarsely on the feature level, given that pedestrians usually stand in the image and are horizontally aligned in nature. Based on this assumption, Sun *et al.* [40], [46] propose to split feature maps horizontally and learn the part feature in a relatively large receptive field. Taking one more step, MGN [41] explores more partition strategies as multiple knowledge representation [47] and fuses different loss functions, further improving the performance. Zhao *et al.* [48] harness the salience map to learn the discriminative harshing code for fast person re-id. To obtain more fine-grained information, several works [49]–[53] introduce one extra human parsing branch to provide part matching information in the feature level. Some pioneering works also explore the neural architecture search to learn fine-grained visual representation [54]–[56]. Besides, to address the misdetection of the input image, Zheng *et al.* [57] apply the spatial transformer network [58] to re-align feature maps. Different from existing works on part alignment in 2D space, the proposed method explores the 3D body structure, which is more close to the prior knowledge of human - a 3D non-rigid object.

C. Learning from Synthetic Data

Another active research line is to leverage the synthetic human data. Although most datasets [7], [59] provide more training data in recent years, the number of images per person is still limited [9]. Therefore, the intra-class variants of every training pedestrian are limited, which largely compromise the model learning and hurt the model scalability to real-world scenarios. To address the data limitation, one line of existing works leverages the generative adversarial network (GAN) [60] to synthesize more high-quality training images, and let the model "see" more appearance variants to learn the robust representation [9], [22], [61]–[66]. Zheng *et al.* [9] first propose a new label smooth regularization for outliers to leverage imperfect generated images. In a similar spirit, Huang *et al.* [67] deploy the pseudo label learning to assign refined labels for synthetic data. Qian *et al.* [64] modify the generation model and add pedestrian images with different poses into training set, yielding the pose-invariant features. Inspired by the conventional encoder-decoder manner, Ge *et al.* [62] propose FD-GAN to learn one pose-invariant feature when encoding the input image. Zhao *et al.* [68] propose a generative occlusion block to dynamically simulate the occlusion in real-world applications. DG-Net [61] disentangles the pedestrian image to two embeddings, *i.e.*, appearance code and structure code, to generate diverse and realistic synthetic images. With the high-quality synthetic data, more discriminative feature can be learned, in turn, improving re-id performance. Furthermore, several works [10], [22], [69]–[71] also apply GAN, *i.e.*, CycleGAN [72], to cross-domain person re-identification by training the model with the target-style synthetic data. In contrast, another line of works [17], [73], [74] is close to our work, which applies the game engine to build 3D models. Sun *et al.* [17] build a large number of 3D person models, and map models to 2D plane for generating more 2D training data. Yao *et al.* [74] and Tang *et al.* [73] manipulate the generation setting and leverage attributes, *e.g.*, color and pose, to enable multi-task learning on 2D synthetic data. Lin *et al.* [75] also leverage the synthetic data to learn the common knowledge of human structure, improving the model scalability on real data. However, different from our work, the above-mentioned studies are mostly investigated in the 2D space, and neglect the 3D geometry information of human bodies. In this work, we argue that the 3D space with the geometry knowledge could help to learn a new type of feature free from several intra-class visual variants, such as viewpoints.

D. *Learning from Point Clouds*

The point cloud is a flexible geometric representation of 3D data structure, which could be obtained by most 3D data acquisition devices, such as radar. The point cloud data is usually unordered, and thus the conventional convolutional neural network (CNN) could not directly work on this kind of data. One of the earliest works, *i.e.*, PointNet [76], proposes to leverage the multi-layer perceptron (MLP) networks and max-pooling layer to fuse the information from multiple points. PointNet++ [77] takes one more step by introducing the sampling layer to distill salient points. To address the limitation in decoding, FoldingNet [78] adds one constant 2D plane to simulate the surface of 3D objects. However, the communication between the 3D points is still limited, and each point is treated independently most of the time. Therefore, Wang *et al.* [79] propose to leverage Graph Neural Network (GNN) [80] to enable the information spread between the k-nearest points. Li *et al.* [81] take one more step and propose to deploy a deeper graph neural network structure, further boosting the performance. Similarly, in this work, we regard every person as one individual graph, while every RGB pixel and the corresponding location are viewed as one node in the graph. More details are provided in Section III.

III. Method

We show a schematic overview of our framework in Figure 3. We next introduce some notations and assumptions, followed by the details of how to learn from 3D points, and how to take advantage of 2D appearance information and 3D structure in one coherent manner.

A. *Preliminaries and Notations*

To conduct person re-identification in the 3D space, we first change the data structure of inputs. In particular, given one person re-id dataset, 2D images are mapped to the 3D space via the off-the-shelf 3D pose estimation [18]. We apply this mapping function to every image in the dataset to obtain 3D point clouds aligned with the 2D appearance. We denote the generated point sets and identity labels as $S = \{s_n\}_{n=1}^N$ and $Y = \{y_n\}_{n=1}^N$, where N is the number of samples in the dataset, $y_n \in [1, K]$, and K is the number of the identity categories. We utilize the matrix format to illustrate the point cloud $s_n \in \mathbb{R}^{m\times 6}$, where m is the number of points, and 6 is the channel number. The former 3 channels contain 3D coordinates XYZ, while the latter 3 channels contain the corresponding RGB information. Given one 3D data $s_n \in \mathbb{R}^{m\times 6}$, our work intends to learn a mapping function F which projects the input s_n to the identity-aware representation $f_n = F_\Theta(s_n)$ with learnable parameters Θ. Unlike the conventional image format, the 3D point clouds are unordered and discrete. We can not directly apply the traditional 2D convolutional layer on $m\times 6$ to capture the local information, *e.g.*, one 3×3 receptive field, since unordered neighbor points may have limited connections to the center point. To address the limitation, we follow the idea of graph neural networks [80] to build the graph $\mathcal{G}$ based on the distance between points. Next we illustrate one basic component, *i.e.*, dynamic graph convolution, to learn from the graph $\mathcal{G}$.

B. *Dynamic Graph Convolution*

To model the relationship between neighbor points, we adopt the k-nearest neighbor (KNN) graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the vertex set, and $\mathcal{E}$ denotes the edge set ($\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$). The KNN graph is directed, and includes self-loop, meaning $(i, i) \in \mathcal{E}$. It is worth to noticing that the selection of the k-nearest neighbors is based on the value of vertexes (points) rather than the initial input order, evading the problem of unordered 3D point clouds. Besides, recent works [79], [81] also show the dynamic graph is superior to the fixed graph structure during training GCN, which alleviates the over-smoothing problem and enlarges the receptive field

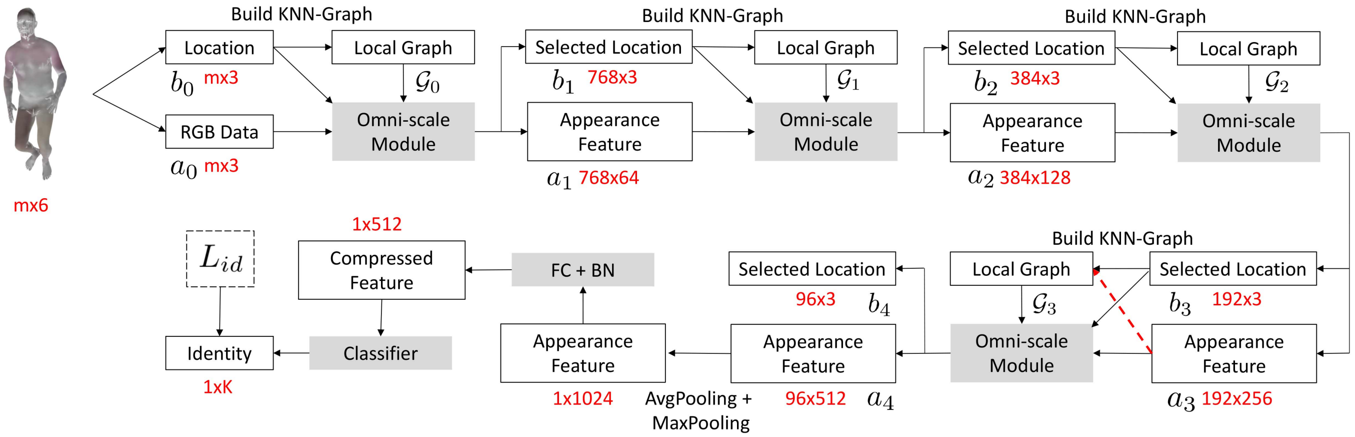
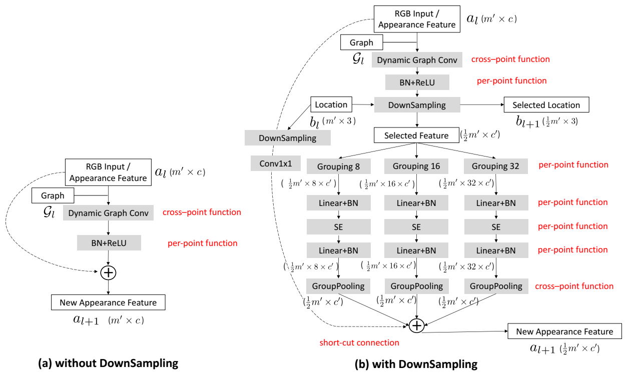


Fig. 3. **OG-Net Architecture**. OG-Net is simply built via stacking Omni-scale Modules. $(m' \times c)$ denotes the feature of m' points with c-dim attribute. Given the point cloud of $(m \times 6)$, we split the geometry location b_0 and the RGB color data a_0. The 3D location information, *i.e.*, (x,y,z), is to build the KNN graph, while the RGB data is to extract the appearance feature as the conventional 2D CNNs. We progressively downsample the number of selected points $\{m, 768, 384, 192, 96\}$, while increasing the appearance feature length $\{3, 64, 128, 256, 512\}$. For the last KNN Graph, we concatenate the position b_3 and the appearance feature a_3 to yield a non-local attention (see the red dash arrow). Finally, we concatenate the outputs of average pooling and max pooling layer, followed by one fully connected (FC) layer and one batch normalization (BN) layer. We adopt the conventional pretext task, *i.e.*, identity classification L_{id}, as the optimization objective to learn the pedestrian representation. When testing, we drop the last classifier and extract the compressed feature of 512 dimensions as the pedestrian representation for matching.



(a) without DownSampling

(b) with DownSampling

Fig. 4. Visualization of Omni-scale Module. We provide the feature shape as the format of $(\cdot)$. For instance, $(m' \times c)$ denotes the feature of m' points with c-dim attribute. (a) We show the basic Omni-scale module without downsampling. (b) We show the Omni-scale module with downsampling, which is similar to the conventional pooling layer. The module distills the number of the points and improves the training efficiency. The dash line denotes the short-cut connection. Besides, we highlight two function types, *i.e.*, cross-point functions and per-point functions, in red. The cross-point function aggregates the feature among neighbor points, while the per-point function only considers the single-point feature. The proposed Omni-scale module consists of these two kinds of functions.

of every node. Following the spirit of the dynamic graph, the KNN graph used in our work is not fixed, and we re-build the graph after every down-sampling layer. The down-sampling layers are to progressively remove redundant points (vertexes), and thus the computation cost of the proposed method is much less than the conventional implementation in [79], [81].

To learn representation from the topology structure of the graph, we follow the spirit of the traditional 2D CNN and deploy one local convolutional layer based on neighbor points with connected edges. In particular, given one node feature x_i, the output x'_i of the dynamic graph convolution could be formulated as:

$$x'_i = \sum_{j:(i,j)\in\mathcal{E},\ j\neq i} (\theta_i x_i + \theta_j x_j) \tag{1}$$

where x_j is the feature of neighbor points in the graph, and there is one edge from i to j. θ is the learnable parameter in Θ. The main difference with the traditional convolution is the definition of the neighbor set. **In this work, we combine two kinds of neighbor choices, *i.e.*, position similarity and feature similarity.** If the graph $\mathcal{G}$ is based on the 3D coordinate similarity, dynamic graph convolution equals to the conventional 2D CNN to capture the local pattern based on the position. We note that this operation is translation invariant, since the global translation, such as ShiftX, ShiftY and Rotation, could not change the connected neighbors in $\mathcal{E}$. On the other hand, if the graph $\mathcal{G}$ is built on the appearance feature, the dynamic graph convolution works as the non-local self-attention as [82], [83], which ignores the local position but pays attention to the area with similar appearance patterns. We next take advantage of the dynamic graph convolution function to build the basic module - Omni-scale module.

C. *Omni-scale Module*

To leverage the rich multi-scale information as the prevailing 2D CNNs, we propose one basic Omni-scale module, which could be easily stacked to form the whole network. The module treats the 3D location and the RGB input differently (see Figure 4 (b)). We denote $l \in [0, L-1]$ as the layer index. The RGB input is the first appearance feature a_0 of $m \times 3$, while the initial 3D position is b_0 of $m \times 3$. Different from the conventional graph CNN, the local k-nearest graph $\mathcal{G}_l$ is dynamically generated according to the input location b_l or the concatenation of a_l and b_l. Given the appearance feature a_l of $m' \times c$, the location b_l of $m' \times 3$ and the KNN graph $\mathcal{G}_l$, the Omni-scale module outputs the appearance feature a_{l+1} and the selected locations b_{l+1}. From the top to the bottom of the module, we first apply Dynamic Graph Convolution to aggregate the k-nearest neighbor features, which is similar to the conventional convolutional layer. Dynamic Graph

Convolution does not change the number of points, and thus the shape of the output feature is $m' \times c'$. If down-sampling points is not applied, we will remain the channel number $c' = c$ following the conventional residual learning [84] to obtain a_{l+1} followed by one batch normalization layer and one ReLU (see Figure 4 (a)). If down-sampling points is applied, we generally set $c' = 2c$ to enlarge the feature channel before downsampling. Then we downsample the location according to the farthest point sampling (FPS) [77]. FPS selects the most distinguish points in the 3D space. We note that only the 3D position b_l is used to calculate the distance and decide the selected points when downsampling. According to the selected location, we also downsample the appearance feature, and only keep the feature of the selected location. Therefore, the shape of the selected location is $\frac{1}{2}m' \times 3$, while the selected feature shape is $\frac{1}{2}m' \times c'$. Next we deploy three branches with different grouping rates $r = \{8, 16, 32\}$, and the three branches do not share weights. In this way, we could capture the information with different receptive fields as the conventional 2D CNNs, *i.e.*, InceptionNet [85]. Each branch consists one grouping layer, two linear layers, two batch normalization (BN) layers, one squeeze-excitation (SE) block [86] and one group max pooling layer to aggregate the local information. Specifically, grouping-r layer is to sample and duplicate the r nearest points for each point, followed by the linear layers, batch normalization and the SE block. We introduce SE-block [86] as one adaptive gate function to re-scale the weight of each branch before the summarization of three branches. Group max pooling layer is to maximize the feature within each group. Finally, we adopt the 'add' to calculate the sum of three branches rather than concatenation, so that the different scale pattern of the same part, such as cloth logos, could be accumulated. The shape of the new appearance feature a_{l+1} is $\frac{1}{2}m' \times c'$, and the shape of the corresponding 3D position b_{l+1} is $\frac{1}{2}m' \times 3$. Alternatively, we could add the short-cut connection to take advantage of the identity representation as ResNet [84].

To summarize, the key of Omni-scale Module is two cross-point functions. The cross-point function indicates the function considers the neighbor points, while the pre-point function only considers the feature of one point itself. One cross-point function is the dynamic graph convolution before down-sampling, which could be simply formulated as $\sum h(x_i, x_j)$, where h denotes a linear function. It mimics the conventional 2D CNN to aggregate the local patterns according to the position. The other is the max group pooling layer in each branch, which could be simply formulated as $\max h(x_i)$. It maximizes neighbor features in each group as the new point feature. Now we have the Omni-scale module to learn from both of the appearance and the geometry structure information in a coherent manner, and next we will utilize Omni-scale modules to build the Omni-scale Graph Network (OG-Net).

D. OG-Net Architecture

The structure of OG-Net is as shown in Figure 3, consisting four Omni-scale modules. We progressively decrease the number of selected points as the conventional CNN. Every time the point number decreases, the channel number of the appearance feature is doubled. After four Omni-scale modules, we could obtain 96 points with 512-dim appearance feature. Similar to [79], we apply the max pooling as well as average pooling to aggregate the point feature, and concatenate the two outputs, yielding the 1024-dim feature. We add one fully-connected layer and one batch normalization layer to compress the feature to 512 dimensions as the pedestrian representation. When inference, we drop the last linear classifier for the pretext classification task, and extract the 512-dim feature to conduct image matching.

Training Objective. We adopt the conventional identity classification as the pretext task to learn the identity-aware feature. The vanilla cross-entropy loss could be formulated as:

$$L_{id} = \mathbb{E}[-log(p(y_n|s_n))] \tag{2}$$

where $p(y_n|s_n)$ is the predicted possibility of s_n belonging to the ground-truth class y_n. The training objective demands that the model could discriminate different identities according to the input points. Besides, other training objectives are orthogonal to our work. 1) In this work, we intend to show the strong potential ability of the 3D space and the proposed OG-Net. We, therefore, only deploy the basic identification loss for a fair comparison with other networks. 2) We deploy the new-released circle loss [87] to show that our work can be fused with better loss functions for further performance boost.

Relation to Existing Methods. The main difference with existing GNN-based networks [78], [79] is three-fold: (1) We extract the multi-scale local information via the proposed Omni-scale Block, which can deal with the common scale variants in 3D person data; (2) We split the XYZ position information and RGB color information, and treat them differently. RGB inputs are used to extract appearance features, while the geometry position is to build the graph for local representation learning; (3) Due to a large number of points in 3D person, we progressively reduce the number of nodes in the graph, facilitating efficient training for 3D person data. On the other hand, compared with PointNet [76] and PointNet++ [77], the proposed OG-Net contains more cross-point functions, and provides topology information, enriching the representation power of the network. The graph could be built on the two kinds of neighbor choices, *i.e.*, position similarity or feature similarity.

IV. Experiment

A. Implementation Details

OG-Net is trained with a mini-batch of 36. We deploy Adam optimizer [88] with amsgrad [89] and the initial learning rate is set to $8e-4$. We gradually decrease the learning rate via the cosine policy [90], and the model is trained for 1000 epochs. To regularize the training, we transfer some traditional 2D data augmentation methods, such as random scale and position jittering, to the 3D space. For instance, position jittering is to add zero-mean Gaussian noise to every point. Following the setting in DGCNN [79], we set the neighbor number of KNN-graph to $k = 20$. The dynamic graph convolution in OG-Net can be any of the existing graph convolution operations, such as EdgeConv [79], SAGE [91] and GAT [92]. In practise, we adopt EdgeConv [79]. Dropout with 0.7 drop probability is used before the last linear classification layer. Since the basic OG-Net is shallow, we do not use the short-cut connection. For the person re-id task, the input image is resized to 128×64, and there are 8192 points with RGB color information. After mapping to the 3D space, we uniformly sample half points to train the OG-Net, and thus the number of input m in Figure 3 equals to 4096. We note that, for other competitive 2D CNN methods, we still follow the common setting, and the 2D image input is resized to 256×128 [40], [61] for a fair comparison.

OG-Net. The channel number of the four Omni-scale Module in OG-Net is {64, 128, 256, 512}. The parameter number is $1.95M$, which is much less than the prevailing CNN structure ResNet-50 ($24.56M$).

OG-Net-Small. To compare with lightweight models, we also introduce OG-Net-Small with fewer channel numbers, *i.e.*, {48, 96, 192, 384}. The parameter number of the model is $1.20M$, which is less than both widely-adopted mobile models, *i.e.*, ShuffleNetV2 ($1.78M$) and MobileNetV2 ($4.16M$).

OG-Net-Deep. We build one deep OG-Net with more Omni-scale Modules. The channel numbers are {48, 96, 96, 192, 192, 384, 384}. The short-cut connection is enabled. Further discussion on short-cut connection is provided in Table V. The parameter number is $2.47M$.

The models are trained from scratch on 3D point clouds. The whole training process costs about 2 days, with one NVIDIA 2080Ti. During testing, we extract the 512-dim feature before the classifier as the pedestrian representation. The feature is L2-normalized. Given one query image, we calculate the cosine similarity between the query feature and the candidate features of gallery images. We sort gallery images and return the ranking list according to the cosine similarity. **3D Reconstruction Details.** The pre-processing 3D body reconstruction is modified from the 3D pose estimation code in [18]. We modify the code and make our code publicly available at [1]. In particular, we obtain the 3D human body mesh as [18], and get the XYZ coordinate for every body vertex. It is worth noting that some XYZ vertex may share the same XY coordinate after projecting to the pixels. It is because some visible foreground overlaps the invisible parts. For instance, the XY coordinates for the human back and chest are usually shared after projection. Therefore, the denoising process is necessary. We map the color of the RGB pixel to the most close 3D vertex according to the XY coordinate. Then we harness the Z coordinate (depth) to remove the wrong mapping for invisible parts. In this way, only the vertex close to the camera has the RGB color and it also ensures that one RGB pixel is only mapped to one corresponding foreground vertex. One output sample is shown in Figure 5(b).

The pixel of the background does not find any matched human body vertex. We directly map such pixels to the XYZ coordinates via setting the Z as the mean depth of all existing body points. In this way, every RGB pixel has the XYZ coordinate. We can obtain the output sample with the 2D background as shown in Figure 5(c).

B. Datasets

We verify the effectiveness of the proposed method on four large-scale person re-id datasets, *i.e.*, Market-1501 [7], DukeMTMC-reID [9], [59], MSMT-17 [10], and CUHK03-NP [35], [93].

Market-1501 [7] is collected in a university campus by 6 cameras, containing $12,936$ training images of 751 identities, $3,368$ query images and $19,732$ gallery images of the other 750 identities. There are no overlapping identities (classes) between the training and test set. Every identity in the training set has 17.2 photos on average. All images are automatically detected by the DPM detector [94].

DukeMTMC-reID [9], [59] consists $16,522$ training images of 702 identites, $2,228$ query images of the other 702 identities and $17,661$ gallery images, which is mostly collected in winter by eight high-resolution cameras. It is challenging in that most pedestrians are in the similar clothes, and may be occluded by cars or trees.

MSMT-17 [10] is one of the newly-released large-scale datasets, including $126,441$ images collected in both indoor and outdoor scenarios with 15 cameras. It contains $32,621$ images of $1,041$ identities for training, $11,659$ query images with $82,161$ gallery images.

CUHK03-NP [93] is one of the early person re-identification datasets. We follow the new protocol in [35] to split 767 identities as the training set, and the rest 700 identities are deployed to verify the model. We utilize the pedestrian images detected by DPM [94] for training and testing, which is more close to real-world scenarios.

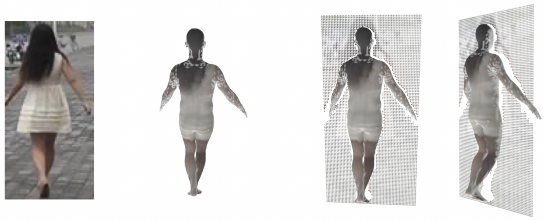
Evaluation Metrics. We report Rank-1 accuracy (R@1) and mean average precision (mAP). Rank-i denotes the probability of the true match in the top-i of the retrieval results, while AP denotes the area under the Precision-Recall curve. The mean of the average precision (mAP) for all query images reflects the precision and recall rate of the retrieval performance. Besides, we also provide the number of model parameters (#params).

Data Limitation. Before the experimental analysis, we would like to illustrate several data limitations. It is mainly due to lossy mapping in the 2D-to-3D process. Due to the restriction of the 3D human model, we could not build the 3D model for several body outliers, such as hair, bag, dress. However, these outliers contain discriminative identity information. For instance, as shown in Figure 5 (a) and (b), the 3D model based on the visible part drops some part of hair and dress of the girl, which is not ideal for representation learning. We think it could be solved via the depth estimation devices, such as Kinect [95], or more sophisticated human models in the future. In this paper, we do not solve the 3D human reconstruction problem, but focus on the person re-identification task. Therefore, as a trade-off, we still introduce the 2D background, and project the corresponding pixel to the XY plane (see Figure 5 (c)).

C. Quantitative Results

Comparisons to the 2D Space. We compare the results on three kinds of inputs, *i.e.*, 2D input, 3D Visible Part and 3D Visible Part with 2D Background. For a fair comparison, the grid of the 2D input is also transformed to the point cloud

[1] https://github.com/layumi/hmr



(a) 2D Image (b) 3D Visible Part (c) 3D Visible Part + 2D Background

Fig. 5. **(a,b)** Visualization of lossy compression in the 2D-to-3D mapping, which drops the body outliers, *e.g.*, hair and dress. **(c)** We still introduce the 2D background to 3D space.

TABLE I
ABLATION STUDY OF DIFFERENT INPUTS ON MARKET-1501, DUKEMTMC-REID AND CUHK03-NP. †: FOR A FAIR COMPARISON, THE MODEL IS TRAINED ON THE TRADITIONAL 2D IMAGE INPUTS WITH EXTRA 3D COORDINATES $(\mathbf{x}, \mathbf{y}, \mathbf{0})$.

Inputs	Market-1501		DukeMTMC-reID		CUHK03-NP	
	R@1	mAP	R@1	mAP	R@1	mAP
3D Visible Part	77.64	54.52	59.52	37.25	**42.79**	**38.29**
2D Image†	85.72	67.28	75.49	55.98	**40.14**	**36.33**
3D Visible Part + 2D Background	**86.79**	**67.92**	**77.33**	**57.74**	43.07	38.06

format as $(\mathbf{x}, \mathbf{y}, \mathbf{0})$, while $\mathbf{z}$ is set to 0. We train OG-Net on three kinds of input data with the same hyper-parameters. As shown in Table I, we observe that the retrieval result of the pure 3D Visible Part input is inferior to that of 2D Image. As discussed in Section IV-B, we speculate that it is due to the lossy 2D-to-3D mapping, which drops several discriminative parts, such as hair, dress, and carrying. In contrast, the 3D Visible Part + 2D Background has achieved superior performance 86.79% Rank@1 and 67.92% mAP to the result of 2D Image (85.72% Rank@1 and 67.28% mAP), which shows that the 3D position information is complementary to 2D color information. Similar results also can be observed on the DukeMTMC-reID dataset and the CUHK03-NP dataset. The 3D information yields consistent accuracy improvements, *i.e.*, +1.84% Rank-1 and +1.76% mAP on DukeMTMC-reID, and +2.93% Rank-1 and +1.73% mAP on CUHK03-NP.

We add one experiment on CUHK03-NP and find that Only 3D Visible Part also works well. We notice that, except for the loss in 2D to 3D conversion, the dataset bias is another important reason. For instance, some identities of Market-1501 only appear in several cameras with the playground background. In this case, 2D background biases are shortcut to recognize one people. The model usually leans to over-fit the 2D background shortcuts and achieves a high performance. To minimize the background bias, we add the ablation study on CUHK03-NP. The CUHK03-NP is collected in the subway station, and the 2D background is almost same (with white wall). Therefore, the background bias plays a limited role. We could observe that the 3D Visible Part can achieve similar Rank@1 and mAP with the 3D Visible Part + 2D Background on CUHK03-NP, surpassing the result on 2D image.

Person Re-id Performance. We compare the proposed method with three groups of competitive methods, *i.e.*, prevailing 2D CNN models, light-weight CNN models, and popular point classification models. We note that the model pre-trained on the large-scale datasets, *e.g.*, ImageNet [99], could yield the performance boost. For a fair comparison, models are trained from scratch with the same optimization objective, *i.e.*, the cross-entropy loss. Since the proposed method is orthogonal to different metric learning losses, we also run experiments with the prevailing circle loss [87]. As shown in Table II, we can make the following observations:

(1) OG-Net has achieved competitive results of 69.02% mAP, 57.92% mAP, 21.57% mAP, and 39.28% mAP on four large-scale person re-id benchmarks with limited training parameters $1.95M$. The mobile OG-Net-Small of less channel width also achieves a close result only with the cross-entropy loss.

(2) Comparing with the point-based methods, such as PointNet++ [77] and DGCNN [79], both OG-Net and OG-Net-Small have surpassed this line of works by a clear margin, which validates the effectiveness of the proposed Omni-scale module in capturing multi-scale neighbor information on point clouds.

(3) Comparing with light-weight 2D CNN models, *i.e.*, ShuffleNetV2 [96] and MobileNetV2 [97], OG-Net-Small has achieved competitive performance with fewer parameters $(1.20M)$.

(4) We apply the same setting to train the model with Circle loss. The strong supervision mechanism of Circle loss sometime compromises the training process. The training process is quite challenging, especially when the class number largely increases in the MSMT-17 dataset. We observe that the proposed model is shallow and relatively easy to converge, so Circle loss generally works well with the proposed structure, and yields performance boost.

(5) Comparing with prevailing 2D CNN models, *i.e.*, ResNet-50 [84] and DenseNet-121 [98], the proposed OG-Net surpasses these models. Furthermore, OG-Net-Deep with deeper structure has achieved better Rank@1 and mAP accuracy. Besides, we also observe that OG-Net is more robust than 2D CNNs, when facing the unseen data. We will discuss this aspect in the following section.

Transferring to Unseen Datasets. To verify the scalability of OG-Net, we train the model on dataset A and directly test the model on dataset B (with no adaptation), which is close to the real-world deployment. We denote the direct transfer learning protocol as $A \rightarrow B$. Three groups of related works are considered. We observe that the modern CNN models are typically over-parameterized, which is prone to over-fit the training dataset. As shown in Table III, both ResNet-50 and DenseNet-121 do not perform well given more parameters. The 3D point cloud-based methods are competitive to the conventional 2D methods. It is worth noting that the proposed OG-Net has outperformed the point-based methods as well as prevailing 2D networks. The results suggest that the proposed method has the potential to adapt one new re-id dataset of unseen environments.

Comparison with Existing Methods. Some existing works [45], [46], [49], [53] harness large backbone models and enlarge input sizes to mine more detailed information, while the

TABLE II
WE MAINLY COMPARE THREE GROUPS OF MODELS TRAINED FROM SCRATCH ON FOUR LARGE-SCALE PERSON RE-ID DATASETS, *i.e.*, MARKET-1501 [7], DUKEMTMC-REID [9], [59], MSMT-17 [10] AND CUHK03-NP [35], [93]. WE REPORT RANK1(%), MAP(%) AND THE NUMBER OF MODEL PARAMTERS (M). THE FIRST GROUP CONTAINS THE POINT-BASED METHODS THAT WE RE-IMPLEMENTED. THE SECOND GROUP CONTAINS THE LIGHTWEIGHT CNN MODELS. THE THIRD GROUP CONTAINS PREVAILING 2D CNN MODELS WITH MORE PARAMETERS.

Method	Input Type	Loss Function	#params(M)	Market-1501		DukeMTMC-reID		MSMT-17		CUHK03-NP	
				R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP
DGCNN [79]	point clouds	CE	1.37	28.89	13.33	29.17	15.16	2.84	1.19	3.4	3.6
PointNet++ (SSG) [77]	point clouds	CE	1.59	61.79	37.89	55.70	35.16	22.94	9.61	14.57	13.97
PointNet++ (MSG) [77]	point clouds	CE	1.87	72.51	47.21	60.23	39.36	28.99	12.52	21.14	19.79
PointNet++ (MSG) [77]	point clouds	CE + Circle	1.87	76.04	52.44	64.23	44.19	22.57	9.55	21.36	19.86
ShuffleNetV2 [96]	images	CE	1.78	79.75	56.80	68.81	48.09	36.80	15.70	25.29	22.90
ShuffleNetV2 [96]	images	CE + Circle	1.78	79.78	58.50	69.34	49.04	33.16	14.00	25.43	23.56
MobileNetV2 [97]	images	CE	4.16	81.95	59.28	71.05	50.45	42.53	18.62	29.57	26.45
MobileNetV2 [97]	images	CE + Circle	4.16	79.39	57.40	69.70	49.75	29.19	11.79	29.14	25.46
OG-Net-Small	point clouds	CE	1.20	86.79	67.92	77.33	57.74	42.44	20.31	43.07	38.06
OG-Net-Small	point clouds	CE + Circle	1.20	87.38	70.48	77.15	58.51	43.84	21.79	46.43	41.79
OG-Net	point clouds	CE	1.95	86.82	69.02	76.53	57.92	44.27	21.57	44.00	39.28
OG-Net	point clouds	CE + Circle	1.95	87.80	70.56	78.37	60.07	45.28	22.81	48.29	43.73
DenseNet-121 [98]	images	CE	8.50	83.14	63.36	73.16	55.08	46.32	21.50	33.64	29.45
DenseNet-121 [98]	images	CE + Circle	8.50	84.26	65.79	74.28	55.75	41.06	18.46	36.21	33.52
ResNet-50 [84]	images	CE	24.56	84.59	65.31	73.20	55.96	46.88	22.25	35.43	32.09
ResNet-50 [84]	images	CE + Circle	24.56	85.27	67.55	74.15	56.83	37.35	16.98	37.29	34.12
OG-Net-Deep	point clouds	CE	2.47	88.36	71.27	76.97	59.23	44.56	21.41	45.71	41.15
OG-Net-Deep	point clouds	CE+Circle	2.47	**88.81**	**72.91**	**78.50**	**60.70**	**47.32**	**24.07**	**49.43**	**45.71**

proposed method intends to provide one parameter-efficient choice for re-id task. We note that the proposed OG-Net is trained from scratch, while most existing method is based on the ImageNet pretrained backbone. As shown in Table VI, the proposed method is still competitive with limited parameters and small input shape. In particular, the proposed method even achieves the close performance with the searched architectures [55], [100] with less parameters. We hope that the proposed method could provide one parameter-efficient alternative choice between the parameter and performance.

D. Qualitative Results

Visualization of Retrieval Results. As shown in Figure 6, we provide the original query, the corresponding 3D query and the top-5 retrieved candidates. Two different cases are studied. One is the typical case that the 3D human reconstruction is relatively good. OG-Net can successfully retrieve the true-matches of different viewpoints (see Figure 6 (a)). On the other hand, we also show the challenging case, including the partially detected query and occlusion. Thanks to the prior knowledge of the human geometry structure, OG-Net can still provide reasonable retrieval results with large scale variants (see Figure 6 (b)). It also verifies the robustness of the proposed approach.

V. FURTHER ANALYSIS AND DISCUSSIONS

Effect of Different Components. In this section, we intend to study the mechanism of the Omni-scale Module. First, we compare the OG-Net without KNN Graph, *i.e.*, $k = 1$. For a fair comparison, we apply one linear layer to replace the dynamic graph convolution. As shown in the second and the third column of Table IV, the performance of OG-Net without leveraging the KNN neighbor information drops from 69.33% mAP to 65.50% mAP. The result suggests that the dynamic graph captures effective local information, which could not be replaced by pre-point function, *e.g.*, linear layer. On the other hand, if we include too many neighbors, *e.g.*, $k = 64$, the model loses the discriminative feature of local patterns, thus compromising the retrieval performance as well. To validate this points, we evaluate the sensitivity analysis on $k = \{4, 8, 16, 32, 64\}$ (see Figure 7). The observation is consistent with the conventional k nearest neighbor algorithms [108] on the neighbor number.

Next, we intend to verify the effectiveness of the last non-local graph. The last graph is built on the k-nearest neighbor of the appearance feature. (In practice, we append the 3-channel position to the appearance feature for building the graph, which prevents duplicate nodes with the same node attribute in the graph.) For a fair comparison, we replace the last non-local graph with the graph based on 3D position only. As shown in the third and the fourth column of Table IV, OG-Net with the last non-local block has surpassed the model with position graph $+1.15\%$ mAP, indicating that the last non-local graph provides effective long-distance attention.

Finally, we study two alternative components, *i.e.*, SE block and short-cut connection. By default, Omni-scale Module deploys SE block but does not add the short-cut connection. As shown in the first and second column in Table IV, we can observe that SE Block improves about $+0.85\%$ mAP from 64.65% to 65.50%. On the other hand, the short-cut connections do not provide significant improvement or performance drop on OG-Net, since OG-Net is relatively shallow with four Omni-scale blocks. As shown in Table V, we deploy the OG-Net-Deep to further validate this point. The observation is consistent with ResNet [84]. The short-cut connection works well on the relatively deep network structure. The performance is improved from 68.49% mAP to 72.91% mAP, and the short-cut connections help the model optimization.

Sensitivity Analysis on the Point Density. Our model is trained with 50% points, *i.e.*, 4096, and thus the best performance is achieved with 50% points remaining. In practice, different depth estimation devices may provide different scan point density. To verify the robustness of the proposed OG-

TABLE III
TRANSFERRING TO UNSEEN DATASETS. HERE WE DIRECTLY DEPLOY THE MODEL TRAINED ON THE DATASET A TO THE UNSEEN DATASET B. WE DENOTE THIS SETTING AS $A \rightarrow B$, WHICH COULD REFLECT THE SCALABILITY OF THE MODEL IN DIFFERENT SCENARIOS. WE OBSERVE THAT OGNET IS GENERALLY SUPERIOR TO THE RESNET-50 AND DENSENET-121 AS WELL AS LIGHTWEIGHT MODELS, SUCH AS SHUFFLENETV2 AND MOBILENETV2.

Method	Input	Loss	Market→Duke		Duke→Market		Market→MSMT		MSMT→Market		Duke→MSMT		MSMT→Duke	
	Type	Function	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP
DGCNN [79]	point clouds	CE	7.4	2.9	13.4	4.4	1.4	0.4	10.0	3.7	1.8	0.5	7.3	2.7
PointNet++ (SSG) [77]	point clouds	CE	18.6	8.4	28.8	11.3	3.9	1.2	32.4	13.3	5.5	1.7	29.0	15.4
PointNet++ (MSG) [77]	point clouds	CE	23.2	11.0	32.8	12.6	5.0	1.5	30.6	12.9	6.5	1.9	24.3	12.4
PointNet++ (MSG) [77]	point clouds	CE + Circle	25.4	12.2	35.1	14.5	5.4	1.7	35.6	15.1	6.4	1.9	31.4	17.3
ShuffleNetV2 [96]	images	CE	17.2	7.2	36.4	13.9	2.8	0.8	36.5	14.1	5.8	1.5	29.3	15.3
ShuffleNetV2 [96]	images	CE + Circle	18.7	8.5	36.2	13.7	3.4	1.0	36.4	14.4	6.0	1.6	29.4	15.1
MobileNetV2 [97]	images	CE	16.7	7.1	34.3	12.4	3.2	0.9	35.9	14.2	5.5	1.4	30.6	15.4
MobileNetV2 [97]	images	CE +Circle	18.5	8.0	34.1	13.3	3.5	0.9	32.1	13.3	5.3	1.4	30.3	15.5
DenseNet-121 [98]	images	CE	11.7	5.0	32.7	11.6	2.9	0.8	34.2	13.0	5.3	1.5	27.8	13.6
DenseNet-121 [98]	images	CE + Circle	12.3	5.3	32.6	11.9	2.6	0.8	31.9	12.0	5.3	1.4	25.2	12.8
ResNet-50 [84]	images	CE	12.1	5.2	34.3	13.5	2.7	0.7	34.7	13.5	5.4	1.5	28.1	14.4
ResNet-50 [84]	images	CE + Circle	15.5	6.9	35.7	13.9	2.9	0.8	32.4	12.2	6.1	1.6	24.3	12.0
OG-Net	point clouds	CE	**26.5**	13.1	35.9	14.5	**5.9**	**1.7**	**40.1**	**17.6**	**6.8**	**1.9**	35.2	**19.3**
OG-Net	point clouds	CE + Circle	26.4	**13.7**	**36.4**	**14.7**	5.3	1.6	38.8	16.9	6.3	**1.9**	**35.3**	**19.3**

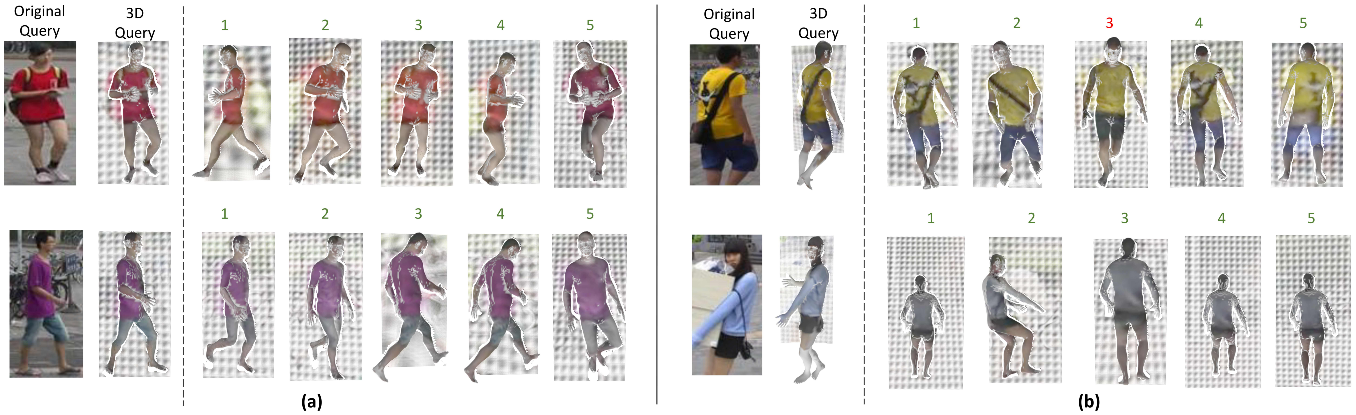


Fig. 6. Visualization of Retrieval Results. **(a)** Given one 3D query, we show the original 2D images and the top-5 retrieval results. **(b)** We also show the challenging case, such as occlusion and the partially detected query. The green index indicates the true-matches, while the red index denotes the false-matches.

TABLE IV
EFFECTIVENESS OF DIFFERENT COMPONENTS. WE COMPARE THE NETWORK VARIANTS, INCLUDING SQUEEZE-EXCITATION (SE), THE USAGE OF KNN GRAPH AND THE LAST NON-LOCAL ATTENTION IN THE MODEL.

Method	Performance			
with Squeeze-excitation?		✓	✓	✓
with KNN Graph?			✓	✓
with Last Non-local?				✓
Rank@1	83.35	84.38	86.43	**87.38**
mAP(%)	64.65	65.50	69.33	**70.48**

TABLE V
EFFECTIVENESS OF THE SHORT-CUT CONNECTION. WE OBSERVE A SIMILAR RESULT WITH [84] THAT THE IMPROVEMENT FROM THE SHORT-CUT CONNECTION IS NOT SIGNIFICANT ON THE "SHALLOW" NETWORK, WHILE IT WORKS WELL ON THE RELATIVELY DEEP NETWORK STRUCTURE.

Method	Short-cut	R@1	mAP
OG-Net	×	86.82	69.02
OG-Net	✓	84.00	65.04
OG-Net-Deep	×	86.28	68.49
OG-Net-Deep	✓	88.81	72.91

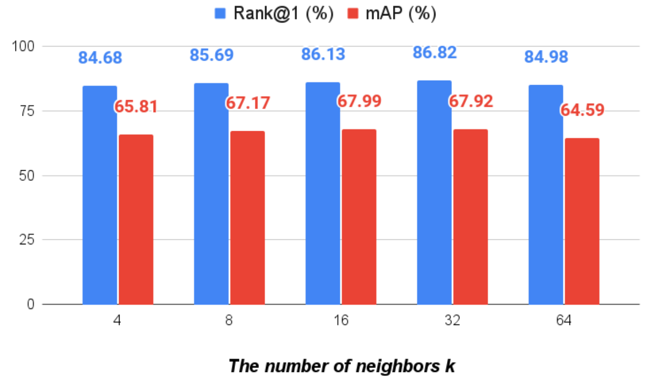


Fig. 7. Sensitivity analysis on the different number of neighbors k. We provide the corresponding re-id performance on Market-1501 in terms of Rank@1(%) and mAP(%).

Net on point density, we synthesize the data similar to that in Figure 8 (left) and conduct the inference. When 25% points remain, OG-Net still could arrive at 84.95% Rank@1 and 65.91% mAP. When 100% points are used, OG-Net arrives at 84.77% Rank@1 and 66.14% mAP. It is because too low/high density impacts the distribution of the k-nearest neighbors, compromising the retrieval performance. Despite

TABLE VI
THE RE-IDENTIFICATION PERFORMANCE ON MARKET-1501 [7] AND DUKEMTMC-REID [9], [59]. WE NOTE THAT THE PROPOSED OG-NET IS TRAINED FROM SCRATCH, WHILE MOST EXISTING METHOD IS BASED ON THE IMAGENET PRETRAINED BACKBONE. THE PROPOSED METHOD IS COMPETITIVE WITH LIMITED PARAMETERS AND SMALL INPUT SHAPE. WE HOPE THAT THE PROPOSED METHOD COULD PROVIDE ONE PARAMETER-EFFICIENT ALTERNATIVE CHOICE BETWEEN THE PARAMETER AND PERFORMANCE. †: TRAINED WITH EXTRA DATA (SPREID [49] IS TRAINED ON 10 RE-ID DATASETS TOGETHER). *: THE MODEL DOES NOT HAVE PRE-TRAINNING ON IMAGENET, AND IS LEARNED FROM SCRATCH.

Method	Backbone	#params(M)	Input Shape	Market-1501		DukeMTMC-reID	
				R@1	mAP	R@1	mAP
PSE [45]	ResNet-50 + extra attention	> 24.6	224×224	87.7	69.0	79.8	62.0
PBR [50]	Inception-V1 + CPM	$7 + 3$	160×80	90.2	76.0	82.1	64.2
SPReID [49]†	$2 \times$ Inception-V3	> 48.0	748×246	92.54	81.34	84.43	70.97
PCB + RPP [46]	ResNet-50 + extra attention	27.2	384×128	93.8	81.6	84.5	71.5
MagnifierNet [53]	ResNet-50 + triple attentions	69.1	384×192	95.8	89.6	90.0	80.7
AACN [101]	Inception-V1 + GCN	> 7	448×192	85.90	66.87	76.85	59.25
DARTS* [54], [100]	-	11.8	164×64	88.3	69.7	79.5	61.3
Auto-ReID* [55]	-	13.1	384×128	90.7	74.6	-	-
OG-Net-Small*	-	1.20	4096	87.38	70.48	77.15	58.51
OG-Net*	-	1.95	4096	87.80	70.56	78.37	60.07
OG-Net-Deep*	-	2.47	4096	88.81	72.91	78.50	60.70

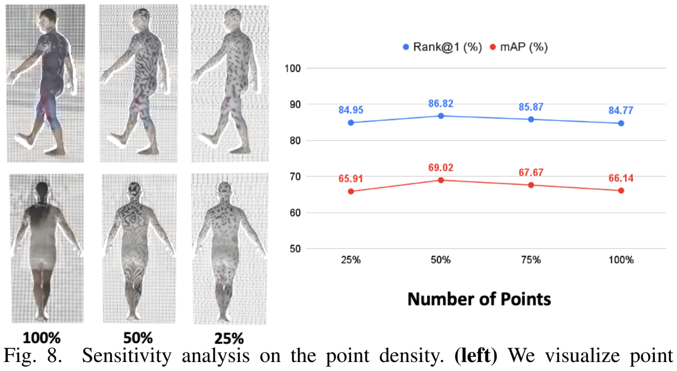


Fig. 8. Sensitivity analysis on the point density. **(left)** We visualize point clouds with different proportion of the point number. **(right)** We provide the corresponding re-id performance in terms of Rank@1(%) and mAP(%) against the point number variants.

TABLE VII
FURTHER ANALYSIS ON THE IMPACT OF DIFFERENT LOSSES. THE PROPOSED NETWORK CAN BE ACCOMPANIED WITH DIFFERENT OPTIMIZATION OBJECTIVES. TO VERIFY THIS POINT, WE FURTHER TRAIN THE OG-NET-SMALL MULTIPLE TIMES WITH WIDELY-USED LOSSES, INCLUDING CONTRAST LOSS [4], TRIPLET LOSS [24], LIFTED LOSS [102], CIRCLE LOSS [87], AND EVALUATE THE IMPACT OF THESE LOSSES ON THE MARKET-1501 DATASET. CE DENOTES THE CROSS-ENTROPY LOSS.

Method	R@1	mAP
CE	86.79	67.92
CE + Lifted [102]	85.99	68.22
CE + Contrast [4]	86.61	68.89
CE + Triplet [24]	86.40	69.28
CE + Contrast + Triplet	86.58	69.72
CE + Circle [87]	87.38	70.48
CE + Triplet + Circle	87.20	70.88

TABLE VIII
CLASSIFICATION RESULTS ON MODELNET [103]. WE DO NOT FOCUS ON THE POINT CLOUD CLASSIFICATION PROBLEM, BUT SHOW THE FEASIBILITY OF THE PROPOSED OG-NET. †: WE PROVIDE RESULTS BASED ON OUR RE-IMPLEMENTATION, WHICH IS SLIGHTLY HIGHER THAN THE REPORTED RESULT IN [77].

Method	#params(M)	Mean-class Accuracy	Overall Accuracy
3DShapeNets [103]	-	77.3	84.7
VoxNet [104]	-	83.0	85.9
PointNet [76]	3.50	86.0	89.2
SpecGCN [105]	2.05	-	91.5
PointNet++(SSG)† [77]	1.62	89.5	92.0
PCNN by Ext [106]	1.40	-	92.2
PointNet++(MSG)† [77]	1.89	90.1	92.7
DGCNN [79]	1.81	90.2	92.9
Point Cloud Transformer [107]	2.88	-	93.2
OG-Net-Small	**1.22**	**90.5**	**93.3**

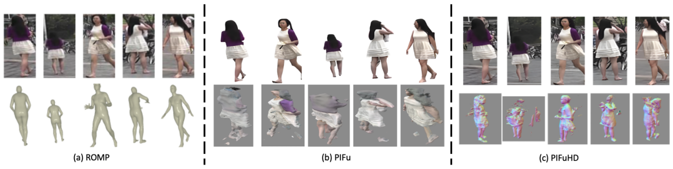


Fig. 9. Visualization of 3D body reconstruction via other three methods, including ROMP [109], PIFu [110] and PIFuHD [111]. We observe that 1) ROMP is based on the body template and close to the hmr [18]. The reconstruction performance is relatively robust but the reconstructed human still misses body outliers, such as dress, hair and bags. 2) The reconstruction body of both PIFu and PIFuHD lack body parts and are not robust to the re-id images. In the future, we think that unsupervised fine-tuning via 2D projection is possible to solve the domain gap [112]. On the other hand, combining these methods, such as PIFu and hmr, is one potential way to make the garment mapping more reliable and accurate. The discussion on better 3D mapping is out of the scope of this paper and we leave these reconstruction choices to the future work.

the density changes, the relative performance drop is small. The result verifies OG-Net is robust to different point density (see Figure 8 (right)).

Different 3D Reconstruction Methods. We try other three existing methods, including ROMP [109], PIFu [110] and PIFuHD [111], to reconstruct the human (see Figure 9). 1) ROMP is similar to hmr [18], which is based on the body template from SMPL [113]. Therefore, the body outliers, such as dress, hair and bags, are still missing. The modified code is available at [2]. We modify the code and add the color mapping. We train the OG-Net-Small baseline on the data generated by ROMP, and achieve 82.42% R@1 and 62.68% mAP on Market-1501, 75.85% R@1 and 56.50% mAP on DukeMTMC-reID. We observe that the performance is slightly lower than the model trained on the data generated by hmr. It is because hmr is trained with in-the-wild images with better

[2] https://github.com/layumi/ROMP

scalability to low-resolution re-id images. 2) PIFu is from [3]. The input is the RGB image with the foreground mask. We leverage the human parsing code to obtain the mask first via the state-of-the-art model on [114] [4]. Then we find that some body parts are missing. Therefore, we skip this method. 3) PIFuHD is from [5] . However, we find that the trained model is not scalable to low-resolution images (in issue [6]), which is common in re-id datasets. Therefore, we first use the super-resolution tool ESRGAN [115] to enlarge the image, and then go through the pipeline. The result is as shown in Figure 9(c), which is not satisfied. Similar to drawbacks on PIFu [110], some body parts are missing. Therefore, we skip this method. As a result, we observe that the default mapping method used in this work, *i.e.*, hmr [18], is simple yet reliable to re-id datasets. In the future, we think that unsupervised fine-tuning via 2D projection is possible to solve the domain gap [112]. On the other hand, combining these methods, such as PIFu and hmr, is one potential way to make the garment mapping more reliable and accurate. The discussion on better 3D mapping is out of the scope of this paper and we leave these reconstruction choices to the future work.

Different Loss Functions. The proposed network is compatible with existing optimization objectives, including Contrast Loss [4], Triplet Loss [24], Lifted Loss [102], Circle Loss [87]. We apply OGNet-Small to evaluate the impact of these losses on the Market-1501 dataset. We report more results in Table VII. It shows that different metric learning losses can be combined with the basic cross-entropy loss (CE) to further improve the performance of learned models.

Evaluation of Point Cloud Classification Task. We also evaluate the proposed OG-Net on the traditional point cloud classification benchmark, *i.e.*, ModelNet [103]. The ModelNet dataset contains 12,311 meshed CAD models of 40 categories. Following the train-test split in [79], 9,843 models are used for training, while the rest 2,468 models are for evaluation. **Note that the ModelNet dataset does not provide RGB information.** To verify the effectiveness of OG-Net, we duplicate the xyz input as the appearance input to train OG-Net. Following other competitive approaches [76], [77], [79], the number of input points is fixed as 1024. As shown in Table VIII, we compare with prevailing models in terms of mean-class accuracy and overall accuracy. Although the proposed method is not designed for cloud point classification task, OG-Net-Small has achieved a competitive result of 90.5% mean-class accuracy and 93.3% overall accuracy with $1.22M$ parameters.

VI. Conclusion

In this work, we provide an early attempt to learn the pedestrian representation in the 3D space, easing the part matching on 2D images. The 3D assumption is aligned with the human visual system of associating the 2D appearance with the 3D geometry structure. Different from existing CNN-based approaches, the proposed Omni-scale Graph Network (OG-Net) takes the advantage of 3D prior knowledge and 2D appearance information in an end-to-end manner, starting from 3D human point clouds. Given 3D points and the nearest neighbour graph, the basic Omni-scale module can aggregate different-scale neighbor information in the topology, enriching the representation ability. This allows the proposed OG-Net efficiently learns discriminative feature via limited network parameters. Extensive experiments suggest that OG-Net exploits the complementary information of 3D geometry information and the 2D appearance, yielding the competitive performance on four person re-id benchmarks. The 3D prior knowledge also benefits the model generalizability on the unseen pedestrian data, which is close to the application in real-world scenarios.

The proposed OG-Net still has room for futher improvements. In experiment, the proposed method learns the representation from the generated 3D point clouds mapping from 2D images. Although it works, the original 2D images are usually resized and compressed in most person re-id datasets, compromising the body shape, *e.g.*, body height. We may consider collecting a new 3D dataset in the future. Furthermore, the proposed method has the potential to many related fields. Similar graph-based models can be employed to other potential fields, *e.g.*, objects with a rigid structure like vehicles [116]–[118] and products [119], [120]. Besides, the efficiency of the 3D mapping process also could be further improved. The 3D mapping time is about 0.008 seconds per image on one Nvidia 2080Ti. The whole Market-1501 dataset can be prepared in about 4 minutes (without consideration of the saving time). The efficiency of 3D mapping may be out of the scope of our paper. Therefore, to our knowledge, we just provide several naïve solutions to improve the efficiency. 1). It is necessary to choose one better backbone. Actually, some recent works also show the improvement on both performance and efficiency by harnessing HRNet [109] or Transformer-based models [121]. In the future, the light-weighted variants, *e.g.*, Lite-HRNet [122] and Swin Transformer [123], could be a strong and efficient alternative. 2). In the real-world scenarios, we may consider the distributed calculation, like federated learning [124], to release the burden of the server. For instance, the 3D mapping can be further bundled with the pedestrian detection [125] on the edge devices.

[3] https://github.com/shunsukesaito/PIFu

[4] https://github.com/PeikeLi/Self-Correction-Human-Parsing

[5] https://github.com/facebookresearch/pifuhd

[6] https://github.com/facebookresearch/pifuhd/issues/60

References

[1] S. Gong, M. Cristani, C. C. Loy, and T. M. Hospedales, “The re-identification challenge,” in *Person re-identification*. Springer, 2014, pp. 1–20.

[2] L. Zhang, T. Xiang, and S. Gong, “Learning a discriminative null space for person re-identification,” in *CVPR*, 2016.

[3] L. Zheng, Y. Yang, and A. G. Hauptmann, “Person re-identification: Past, present and future,” *arXiv:1610.02984*, 2016.

[4] Z. Zheng, L. Zheng, and Y. Yang, “A discriminatively learned cnn embedding for person reidentification,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 14, no. 1, p. 13, 2018.

[5] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, “Deep learning for person re-identification: A survey and outlook,” *TPAMI*, 2020.

[6] Z. Wang, Z. Wang, Y. Zheng, Y. Wu, W. Zeng, and S. Satoh, “Beyond intra-modality: A survey of heterogeneous person re-identification,” *IJCAI*, 2020.

[7] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *ICCV*, 2015.

[8] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks." in *ICML*, 2016.

[9] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," in *ICCV*, 2017.

[10] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer gan to bridge domain gap for person re-identification," in *CVPR*, 2018.

[11] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Omni-scale feature learning for person re-identification," in *ICCV*, 2019.

[12] W. Li, X. Zhu, and S. Gong, "Harmonious attention network for person re-identification," in *CVPR*, 2018.

[13] X. Qian, Y. Fu, Y.-G. Jiang, T. Xiang, and X. Xue, "Multi-scale deep learning architectures for person re-identification," in *ICCV*, 2017.

[14] X. Qian, Y. Fu, T. Xiang, Y.-G. Jiang, and X. Xue, "Leader-based multi-scale attention deep architecture for person re-identification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 2, pp. 371–385, 2019.

[15] X. Zhang, H. Luo, X. Fan, W. Xiang, Y. Sun, Q. Xiao, W. Jiang, C. Zhang, and J. Sun, "Alignedreid: Surpassing human-level performance in person re-identification," *arXiv:1711.08184*, 2017.

[16] I. B. Barbosa, M. Cristani, B. Caputo, A. Rognhaugen, and T. Theoharis, "Looking beyond appearances: Synthetic training data for deep cnns in re-identification," *Computer Vision and Image Understanding*, vol. 167, pp. 50–62, 2018.

[17] X. Sun and L. Zheng, "Dissecting person re-identification from the viewpoint of viewpoint," in *CVPR*, 2019.

[18] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *CVPR*, 2018.

[19] X. Yang, M. Wang, R. Hong, Q. Tian, and Y. Rui, "Enhancing person re-identification in a self-trained subspace," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 13, no. 3, pp. 1–23, 2017.

[20] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Deep metric learning for person re-identification," in *ICPR*, 2014.

[21] K. Lin, J. Lu, C.-S. Chen, J. Zhou, and M.-T. Sun, "Unsupervised deep learning of compact binary descriptors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 6, pp. 1501–1514, 2018.

[22] Z. Zhong, L. Zheng, Z. Zheng, S. Li, and Y. Yang, "Camera style adaptation for person re-identification," in *CVPR*, 2018.

[23] Q. Yu, X. Chang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "The devil is in the middle: Exploiting mid-level representations for cross-domain instance matching," *arXiv:1711.08106*, 2017.

[24] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv:1703.07737*, 2017.

[25] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification," in *CVPR*, 2018.

[26] X. Yang, P. Zhou, and M. Wang, "Person reidentification via structural deep metric learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 10, pp. 2987–2998, 2018.

[27] C. Zhao, X. Lv, Z. Zhang, W. Zuo, J. Wu, and D. Miao, "Deep fusion feature representation learning with hard mining center-triplet loss for person re-identification," *IEEE Transactions on Multimedia*, vol. 22, no. 12, pp. 3180–3195, 2020.

[28] Y. Ding, H. Fan, M. Xu, and Y. Yang, "Adaptive exploration for unsupervised person re-identification," *TOMM*, vol. 16, no. 1, pp. 3:1–3:19, 2020.

[29] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang, "Joint detection and identification feature learning for person search," in *CVPR*, 2017.

[30] Y. Lin, L. Zheng, Z. Zheng, Y. Wu, Z. Hu, C. Yan, and Y. Yang, "Improving person re-identification by attribute and identity learning," *Pattern Recognition*, vol. 95, pp. 151–161, 2019.

[31] J. Wang, X. Zhu, S. Gong, and W. Li, "Transferable joint attribute-identity deep learning for unsupervised person re-identification," in *CVPR*, 2018.

[32] ——, "Attribute recognition by joint recurrent learning of context and correlation," in *ICCV*, 2017.

[33] Z. Zheng, L. Zheng, M. Garrett, Y. Yang, M. Xu, and Y.-D. Shen, "Dual-path convolutional image-text embeddings with instance loss," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 2, pp. 1–23, 2020, doi:10.1145/3383184.

[34] W. Li, X. Zhu, and S. Gong, "Person re-identification by deep joint learning of multi-loss classification," *IJCAI*, 2017.

[35] Z. Zhong, L. Zheng, D. Cao, and S. Li, "Re-ranking person re-identification with k-reciprocal encoding," in *CVPR*, 2017.

[36] S. Bai, X. Bai, and Q. Tian, "Scalable person re-identification on supervised smoothed manifold," in *CVPR*, 2017.

[37] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian, "Pose-driven deep convolutional model for person re-identification," in *ICCV*, 2017.

[38] Z. Zhang, C. Lan, W. Zeng, and Z. Chen, "Densely semantically aligned person re-identification," in *CVPR*, 2019.

[39] L. Zheng, Y. Huang, H. Lu, and Y. Yang, "Pose-invariant embedding for deep person re-identification," *IEEE Transactions on Image Processing*, vol. 28, no. 9, pp. 4500–4509, 2019.

[40] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling," *ECCV*, 2018.

[41] G. Wang, Y. Yuan, X. Chen, J. Li, and X. Zhou, "Learning discriminative features with multiple granularities for person re-identification," in *ACM Multimedia*, 2018.

[42] L. Wu, Y. Wang, J. Gao, M. Wang, Z.-J. Zha, and D. Tao, "Deep coattention-based comparator for relative representation learning in person re-identification," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 2, pp. 722–735, 2020.

[43] R. Hou, B. Ma, H. Chang, X. Gu, S. Shan, and X. Chen, "Iaunet: Global context-aware feature learning for person reidentification," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[44] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *CVPR*, 2016, pp. 4724–4732.

[45] M. Saquib Sarfraz, A. Schumann, A. Eberle, and R. Stiefelhagen, "A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking," in *CVPR*, 2018.

[46] Y. Sun, L. Zheng, Y. Li, Y. Yang, Q. Tian, and S. Wang, "Learning part-based convolutional features for person re-identification," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[47] Y. Yang, Y. Zhuang, and Y. Pan, "Multiple knowledge representation for big data artificial intelligence: framework, applications, and case studies," *Front. Inf. Technol. Electron. Eng.*, vol. 22, no. 12, pp. 1551–1558, 2021, doi:10.1631/FITEE.2100463.

[48] C. Zhao, Y. Tu, Z. Lai, F. Shen, H. T. Shen, and D. Miao, "Salience-guided iterative asymmetric mutual hashing for fast person re-identification," *IEEE Transactions on Image Processing*, vol. 30, pp. 7776–7789, 2021.

[49] M. M. Kalayeh, E. Basaran, M. Gökmen, M. E. Kamasak, and M. Shah, "Human semantic parsing for person re-identification," in *CVPR*, 2018.

[50] Y. Suh, J. Wang, S. Tang, T. Mei, and K. Mu Lee, "Part-aligned bilinear representations for person re-identification," in *ECCV*, 2018.

[51] S. Gao, J. Wang, H. Lu, and Z. Liu, "Pose-guided visible part matching for occluded person reid," in *CVPR*, 2020.

[52] J. Miao, Y. Wu, P. Liu, Y. Ding, and Y. Yang, "Pose-guided feature alignment for occluded person re-identification," in *ICCV*, 2019, pp. 542–551.

[53] Y. Lan, Y. Liu, X. Zhou, M. Tian, X. Zhang, S. Yi, and H. Li, "Magnifiernet: Towards semantic adversary and fusion for person re-identification," *BMVC*, 2020.

[54] Q. Zhou, B. Zhong, X. Liu, and R. Ji, "Attention-based neural architecture search for person re-identification," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[55] R. Quan, X. Dong, Y. Wu, L. Zhu, and Y. Yang, "Auto-reid: Searching for a part-aware convnet for person re-identification," in *ICCV*, 2019.

[56] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, X. Chen, and X. Wang, "A comprehensive survey of neural architecture search: Challenges and solutions," *ACM Computing Surveys (CSUR)*, vol. 54, no. 4, pp. 1–34, 2021.

[57] Z. Zheng, L. Zheng, and Y. Yang, "Pedestrian alignment network for large-scale person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2018, doi:10.1109/TCSVT.2018.2873599.

[58] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *NeurIPS*, 2015.

[59] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *ECCV*, 2016.

[60] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NeurIPS*, 2014.

[61] Z. Zheng, X. Yang, Z. Yu, L. Zheng, Y. Yang, and J. Kautz, "Joint discriminative and generative learning for person re-identification," in *CVPR*, 2019.

[62] Y. Ge, Z. Li, H. Zhao, G. Yin, S. Yi, X. Wang, and H. Li, "Fd-gan: Pose-guided feature distilling gan for robust person re-identification," in *NeurIPS*, 2018.

[63] C. Eom and B. Ham, "Learning disentangled representation for robust person re-identification," in *NeurIPS*, 2019.

[64] X. Qian, Y. Fu, T. Xiang, W. Wang, J. Qiu, Y. Wu, Y.-G. Jiang, and X. Xue, "Pose-normalized image generation for person re-identification," in *ECCV*, 2018.

[65] J. Liu, B. Ni, Y. Yan, P. Zhou, S. Cheng, and J. Hu, "Pose transferrable person re-identification," in *CVPR*, 2018.

[66] Y. Zou, X. Yang, Z. Yu, B. Kumar, and J. Kautz, "Joint disentangling and adaptation for cross-domain person re-identification," *ECCV*, 2020.

[67] Y. Huang, J. Xu, Q. Wu, Z. Zheng, Z. Zhang, and J. Zhang, "Multi-pseudo regularized label for generated data in person re-identification," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1391–1403, 2018.

[68] C. Zhao, X. Lv, S. Dou, S. Zhang, J. Wu, and L. Wang, "Incremental generative occlusion adversarial suppression network for person reid," *IEEE Transactions on Image Processing*, vol. 30, pp. 4212–4224, 2021.

[69] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao, "Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification," in *CVPR*, 2018.

[70] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang, "Learning to adapt invariance in memory for person re-identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[71] Z. Wang, Z. Wang, Y. Zheng, Y.-Y. Chuang, and S. Satoh, "Learning to reduce dual-level discrepancy for infrared-visible person re-identification," in *CVPR*, 2019, pp. 618–626.

[72] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networkss," in *ICCV*, 2017.

[73] Z. Tang, M. Naphade, S. Birchfield, J. Tremblay, W. Hodge, R. Kumar, S. Wang, and X. Yang, "Pamtri: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data," in *ICCV*, 2019.

[74] Y. Yao, L. Zheng, X. Yang, M. Naphade, and T. Gedeon, "Simulating content consistent vehicle datasets with attribute descent," *ECCV*, 2020.

[75] K. Lin, L. Wang, K. Luo, Y. Chen, Z. Liu, and M.-T. Sun, "Cross-domain complementary learning using pose for multi-person part segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[76] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *CVPR*, 2017.

[77] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *NeurIPS*, 2017.

[78] Y. Yang, C. Feng, Y. Shen, and D. Tian, "Foldingnet: Point cloud auto-encoder via deep grid deformation," in *CVPR*, 2018.

[79] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 5, pp. 1–12, 2019.

[80] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2008.

[81] G. Li, M. Muller, A. Thabet, and B. Ghanem, "Deepgcns: Can gcns go as deep as cnns?" in *ICCV*, 2019.

[82] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, 2018.

[83] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," *ICML*, 2019.

[84] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[85] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *AAAI*, 2017.

[86] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018.

[87] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *CVPR*, 2020.

[88] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[89] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of adam and beyond," *ICLR*, 2019.

[90] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *ICLR*, 2017.

[91] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *NeurIPS*, 2017.

[92] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" *ICLR*, 2019.

[93] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *CVPR*, 2014.

[94] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[95] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with microsoft kinect sensor: A review," *IEEE transactions on cybernetics*, vol. 43, no. 5, pp. 1318–1334, 2013.

[96] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *CVPR*, 2018.

[97] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *CVPR*, 2018.

[98] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *CVPR*, 2017.

[99] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[100] H. Liu, K. Simonyan, and Y. Yang, "Darts: Differentiable architecture search," *ICLR*, 2019.

[101] J. Xu, R. Zhao, F. Zhu, H. Wang, and W. Ouyang, "Attention-aware compositional network for person re-identification," in *CVPR*, 2018.

[102] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *CVPR*, 2016.

[103] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *CVPR*, 2015, pp. 1912–1920.

[104] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *IROS*, 2015.

[105] C. Wang, B. Samari, and K. Siddiqi, "Local spectral graph convolution for point set feature learning," in *ECCV*, 2018.

[106] M. Atzmon, H. Maron, and Y. Lipman, "Point convolutional neural networks by extension operators," *ACM Transactions on Graphics*, 2018.

[107] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "Pct: Point cloud transformer," *Computational Visual Media*, vol. 7, no. 2, pp. 187–199, 2021.

[108] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[109] Y. Sun, Q. Bao, W. Liu, Y. Fu, B. Michael J., and T. Mei, "Monocular, one-stage, regression of multiple 3d people," in *ICCV*, 2021.

[110] S. Saito, , Z. Huang, R. Natsume, S. Morishima, A. Kanazawa, and H. Li, "Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization," *ICCV*, 2019.

[111] S. Saito, T. Simon, J. Saragih, and H. Joo, "Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization," in *CVPR*, 2020.

[112] X. Li, S. Liu, K. Kim, S. De Mello, V. Jampani, M.-H. Yang, and J. Kautz, "Self-supervised single-view 3d reconstruction via semantic consistency," in *ECCV*, 2020.

[113] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, vol. 34, no. 6, pp. 248:1–248:16, Oct. 2015.

[114] P. Li, Y. Xu, Y. Wei, and Y. Yang, "Self-correction for human parsing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[115] X. Wang, L. Xie, C. Dong, and Y. Shan, "Real-esrgan: Training real-world blind super-resolution with pure synthetic data," in *ICCVW*, 2021.

[116] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, and J.-N. Hwang, "Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification," in *CVPR*, 2019.

[117] X. Zhang, R. Zhang, J. Cao, D. Gong, M. You, and C. Shen, "Part-guided attention learning for vehicle re-identification," *TITS*, 2020.

[118] Z. Zheng, T. Ruan, Y. Wei, Y. Yang, and T. Mei, "Vehiclenet: Learning robust visual representation for vehicle re-identification," *IEEE Transactions on Multimedia (TMM)*, 2020, doi:10.1109/TMM.2020.3014488.

[119] X.-S. Wei, Q. Cui, L. Yang, P. Wang, and L. Liu, "Rpc: A large-scale retail product checkout dataset," *arXiv:1901.07249*, 2019.

[120] S. Liu, J. Feng, Z. Song, T. Zhang, H. Lu, C. Xu, and S. Yan, "Hi, magic closet, tell me what to wear!" in *ACM Multimedia*, 2012.

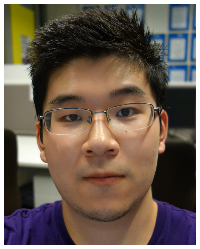
[121] K. Lin, L. Wang, and Z. Liu, "End-to-end human pose and mesh reconstruction with transformers," in *CVPR*, 2021.

[122] C. Yu, B. Xiao, C. Gao, L. Yuan, L. Zhang, N. Sang, and J. Wang, "Lite-hrnet: A lightweight high-resolution network," in *CVPR*, 2021.

[123] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, “Swin transformer: Hierarchical vision transformer using shifted windows,” *ICCV*, 2021.

[124] G. Wu and S. Gong, “Decentralised learning from independent multi-domain labels for person re-identification,” *AAAI*, 2021.

[125] Y. Xu, C. Zhou, X. Yu, B. Xiao, and Y. Yang, “Pyramidal multiple instance detection network with mask guided self-correction for weakly supervised object detection,” *IEEE Transactions on Image Processing*, vol. 30, pp. 3029–3040, 2021, doi:10.1109/TIP.2021.3056887.



Zhedong Zheng received the Ph.D. degree from the University of Technology Sydney, Australia, in 2021 and the B.S. degree from Fudan University, China, in 2016. He is currently a postdoctoral research fellow at Sea-NExT Joint Lab, School of Computing, National University of Singapore. He was an intern at Nvidia Research (2018) and Baidu Research (2020). His research interests include robust learning for image retrieval, generative learning for data augmentation, and unsupervised domain adaptation.

Xiaohan Wang received the Ph.D. degree in computer science from University of Technology Sydney, Australia, in 2021. He received the B.E. degree from University of Science and Technology of China, China, in 2017. He is currently a postdoctoral researcher with the College of Computer Science and Technology, Zhejiang University, China. His research interests are video understanding and multi-modal analysis.

Nenggan Zheng received the B. E. degree in Biomedical Engineering and the Ph. D. degree in Computer Science, both from Zhejiang University, in 2002 and 2009, respectively. He is currently a professor in computer science with the Academy for Advanced Studies, Zhejiang University. His current research interests include artificial intelligence, embedded systems, and brain-computer interface.

Yi Yang received the Ph.D. degree in computer science from Zhejiang University, China, in 2010. He is currently a professor with the College of Computer Science and Technology, Zhejiang University, China. He was a professor with University of Technology Sydney, Australia and a postdoctoral researcher with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA. His current research interest includes machine learning and its applications to multimedia content analysis and computer vision, such as multimedia analysis and video semantics understanding.