Pattern Analysis and Applications (2024) 27:80
https://doi.org/10.1007/s10044-024-01294-9

THEORETICAL ADVANCES

Boosting person ReID feature extraction via dynamic convolution

Elif Ecem Akbaba[1] · **Filiz Gurkan**[2] · **Bilge Gunsel**[1]

Received: 17 August 2023 / Accepted: 14 June 2024 / Published online: 8 July 2024

Abstract
Extraction of discriminative features is crucial in person re-identification (ReID) which aims to match a query image of a person to her/his images, captured by different cameras. The conventional deep feature extraction methods on ReID employ CNNs with static convolutional kernels, where the kernel parameters are optimized during the training and remain constant in the inference. This approach limits the network's ability to model complex contents and decreases performance, particularly when dealing with occlusions or pose changes. In this work, to improve the performance without a significant increase in parameter size, we present a novel approach by utilizing a channel fusion-based dynamic convolution backbone network, which enables the kernels to change adaptively based on the input image, within two existing ReID network architectures. We replace the backbone network of two ReID methods to investigate the effect of dynamic convolution on both simple and complex networks. The first one called Baseline, is a simpler network with fewer layers, while the second, CaceNet represents a more complex architecture with higher performance. Evaluation results demonstrate that both of the designed dynamic networks improve identification accuracy compared to the static counterparts. A significant increase in accuracy is reported under occlusion tested on Occluded-DukeMTMC. Moreover, our approach achieves a performance comparable to the state-of-the-art on Market1501, DukeMTMC-reID, and CUHK03 with a limited computational load. These findings validate the effectiveness of the dynamic convolution in enhancing the person ReID networks and push the boundaries of performance in this domain.

Keywords Person re-identification · Deep learning · Dynamic convolution · Channel fusion

1 Introduction

Person re-identification (Re-ID), which aims to match individuals across non-overlapping camera views, has a wide range of applications, such as video surveillance, public safety, and monitoring systems [1, 2]. Despite significant progress in the latest methods, ReID remains a difficult task because the appearance of a person may not be reliable for accurately matching individuals, often under challenging conditions [3]. These include occlusion, which occurs when a person is partially hidden by objects or other individuals, and pose change where individuals may appear in various body orientations (Fig. 1). All these factors can result in incomplete feature representations and the similarity between the same person with different poses may be low.

To attain high-quality, person-localized features and overcome these challenges inherent in the ReID task, numerous methodologies have been proposed. These approaches enhance the global, local or both features by employing part-based methods, attention mechanisms, and transformer architectures so on [4–13]. Despite that local features provide more detailed information, most of the methods are highly affected by background noise. Moreover, due to occlusion, the misalignment of body parts in local patches complicates the extraction of representative feature embeddings, consequently affecting model performance [3, 5, 14].

Filiz Gurkan and Bilge Gunsel have contributed equally.

Coding: Elif Ecem Akbaba.

✉ Filiz Gurkan
filiz.gurkan@medeniyet.edu.tr

Elif Ecem Akbaba
akbabael@itu.edu.tr

Bilge Gunsel
gunselb@itu.edu.tr

[1] Electronics and Comm. Engineering, Istanbul Technical University, Istanbul, Turkey

[2] Electrical and Electronics Engineering, Istanbul Medeniyet University, Istanbul, Turkey

Fig. 1 Challenging cases in person ReID

In addressing these challenges, some methods utilize extra pre-trained human parsing models or pose estimators to locate the human body parts [7, 15, 16] and personal belongings [6]. However, these methods can be computationally expensive, as they need auxiliary processes such as segmentation or pose estimation. An alternative approach is through the utilization of attention mechanisms, which emphasize the informative regions within the images [8, 9]. But, methods that highlight important regions within an individual image, exhibit limitations in learning of the crucial regions between query-gallery pairs which are important for matching. Due to this some approaches consider related informative regions based on an image pair [10, 11]. However, the efficacy of these methods in extracting informative features is heavily dependent on the performance of the underlying backbone. In recent years, transformer architectures have increased in popularity and they are being frequently employed in ReID tasks due to their ability to capture long-range dependencies [12, 13] While transformer-based frameworks offer advantages, including enhanced feature embedding, they also have higher computational cost compared to CNN-based architectures.

As in many tasks, a trade-off exists between computational load and performance in the context of ReID. It is significant to obtain discriminative features to improve the performance of person ReID while managing computational resources efficiently. However, because images can vary a lot in resolution, target sizes, and background clutter, it's tough to extract rich feature embeddings and properly match images using fixed models with limited complexity [17] dynamically matches local features and calculates the distance between aligning patches, instead of directly measuring similarity, by computing the shortest path distance [18] propose a transformer-based dynamic prototype mask that automatically aligns and selects a visible pattern subspace for each input image. On the other hand, dynamic convolutional-based networks have been widely employed in various tasks, such as classification, and key-point detection because of performance increment by using input-specific convolution filters [19–22]. Unlike conventional convolution, dynamic convolution allows neural networks to adaptively change kernel weights to focus on the informative parts of input images. This adaptability leads to more efficient and effective feature extraction without the necessity of complex and resource-intensive approaches. Since person re-identification requires capturing fine-grained details and learning discriminative features from images, dynamic convolution significantly impacts their performance.

To this end, in this paper, we propose a novel ReID network by integrating channel fusion-based dynamic convolution into the backbone architecture of an existing ReID method. Because the backbone serves as a feature extractor, the proposed method enhances the feature extraction process, enabling our model to capture discriminative and person-specific information adaptively.

- We propose the dynamical convolution as an effective tool that enables channel-wise attention as well as channel fusion to deal with occlusion and pose changes that constitute two challenging problems in person ReID. It is shown that the dynamical convolution empowers the ReID network to adapt its convolution kernels to the specific characteristics of each input without a significant increase in parameter size. To the best of our knowledge, our research is the first work to adopt the dynamic convolution to person ReID task.
- We designed DY-ResNet50 backbone architecture by replacing the convolutional layers of ResNet50 with the dynamic counterparts. Two cost-effective ReID networks, "Dynamic Baseline (DY-BL)" and "Dynamic CaceNet (DYCace)", are designed and investigated how input-dependent convolution kernels increase the feature discrimination capability. We trained both networks in an end-to-end manner and demonstrated that the dynamic networks reach higher performance at earlier training epochs compared to their conventional counterparts with the help of an input-dependent adaptive feature extraction process.
- In addition to existing ones, we present two novel evaluation metrics, first-l accuracy and mAP$_l$ that provide valu-

able insights into the model's performance on the first-l correct matches. The primary purpose of these metrics is to assess the model's performance in a more realistic scenario, where only top l correct candidates are considered.
- Our proposed method also reduces the matching distances between query and gallery images during the inference step. This reduction means higher confidence levels, as it enables more reliable identification of matching pairs.
- We evaluate the proposed dynamic ReID networks on four commonly used datasets. Numerical results demonstrate that DY-BL reaches higher performance compared to its static counterpart. Besides, DY-Cace exceeded state-of-the-art performance with a limited computational cost, especially in a challenging scenario like occlusion. The source code and trained models for all datasets are accessible at https://github.com/msprITU/DY-REID.

2 Related work

Over the past years, several methods have been proposed for person re-identification and this section summarizes the ones related to our model. The quality and discriminative power of extracted features are highly important in ReID, as these features serve as the foundation for matching images. Under challenging conditions, such as occlusion or pose change, insufficient feature representations can lead to false matching and decreased performance. To address these challenges, recent methods have been worked on to improve the quality of feature representations, where the features can be global, local or a combination of them [1–3]. Methods that use global features focus on capturing the overall information of the person from the entire image, local features, on the other hand, focus on local body regions to provide fine-grained information about the images. These features can be extracted and used in various ways, including part-based methods, attention mechanisms, transformer architecture so on. In part-based methods, the idea is to divide the person's body into different patches and extract features independently from each part. The motivation behind part-based methods is to handle variations in pose, viewpoint, and occlusion.

In particular, [4] utilizes part-based features by dividing the feature maps obtained from the backbone network into equal sub-regions to extract fine-grained local features [5] introduces a coarse-to-fine pyramid model that incorporates local and global information and integrates gradual cues between them to match images at different scales. While these approaches seek to enrich feature representations, they may require a significant amount of computational resources due to the complexity of the models. Additionally, the extraction of representative feature embeddings becomes challenging when body parts in local patches are misaligned due to occlusion or pose change [6] introduces a method that employs human semantic parsing to segment both body parts and personal belongings where the body parts/belongings are identified with a cascade clustering algorithm. For identification, the method utilizes features extracted from the visible parts of individuals. However, it's important to note that it requires additional time for clustering, (approximately 5 h for Market1501 dataset). In [7], local features from various body parts are extracted by using a pose estimation model. Furthermore, the incorporation of graph convolution aims to capture informative relationships between local and global features. However, the requirement of a pre-trained pose estimation model limits its effectiveness and robustness.

In addition to these methods, some approaches employ attention mechanisms to focus on the most informative parts of the images, while ignoring distracting regions. To alleviate background clutter and focus on the person, [8] proposes a technique that includes segmentation of input image into the body and background. In addition to the global features, the body and the background features are extracted separately by utilizing the attention mechanism. However, the requirement of the segmentation masks and processing of the body, background, and entire image limits the computational efficiency [9] proposes a method focused on the most informative regions through an attention enhancement branch. On the other hand, the attention suppression branch is applied to erase some regions to force the network to extract additional information from the remaining areas. However, while the method effectively focuses on the most informative regions in individual images, it does not consider regions between the query and the gallery images which is important for ReID. In contrast to the aforementioned methods, [11] applies an attention module to automatically select decisive visual clues based on the visual content of query-gallery individuals and pairs, where the clues are used to extract conditional features with a graph convolutional network. This approach allows for a comprehensive understanding of the relationships within and between images, resulting in enhanced feature extraction and improved ReID performance. Because the conditional embedding branch also improves the individual feature extraction, to limit the computational load during inference, only individual features are extracted and used.

In the past few years, transformer-based methods have been proposed to improve the performance of person ReID [12] encodes images as patch sequences and enhances the transformer baseline. Proposed jigsaw patch module aims to rearrange patch embeddings for robust features, while side information embeddings mitigate bias towards camera/view variations by incorporating non-visual

clues through learnable embeddings [13] leverages pose information to disentangle semantic components, such as human body or joint parts, and selectively matches non-occluded parts. It comprises four modules: vision transformer, pose-guided feature aggregation, pose-view matching, and pose-guided push loss and achieves state-of-the-art performance. However, the transformer-based frameworks have higher computational costs compared to CNN-based models. Also, one drawback includes potential challenges in adjusting the hyper-parameters because the model has many layers and parameters.

In recent years, various approaches have been developed to enhance the extraction of high-quality and discriminative features across different tasks. Dynamic convolution is one of the methods that has been proposed to introduce input adaptability to neural networks, facilitating the cost-effective extraction of high-quality and localized features [23]. Unlike conventional static convolution-based networks which apply fixed filter weights to input during inference, dynamic convolution-based networks exhibit higher levels of flexibility and adaptability by dynamically adjusting filter weights based on input patterns. This adaptability enables the capture of fine-grained details and the effective handling of variations. Due to its effective feature extraction capability, dynamic convolution has wide usage across diverse domains [19] proposes an approach that aggregates multiple static kernels based on input-dependent attention for image classification and detection tasks. In [20] and [21], a small sub-network generates the kernel weights, and then generated kernels are applied the to corresponding input for instance segmentation and few-shot object detection tasks, respectively [22] proposes a new approach to dynamic convolution via matrix decomposition. It introduces dynamic channel fusion, which not only enables significant dimension reduction of the latent space but also mitigates the joint optimization difficulty. The proposed method is easier to train and requires significantly fewer parameters without decreasing accuracy. Since person re-identification requires capturing fine-grained details and extracting efficient and effective features from images, dynamic convolution significantly impacts their performance.

In this work, we propose the utilization of a dynamic convolutional backbone network, DY-ResNet50, that leverages the channel-wise attention and channel fusion [22] within two existing ReID network architectures [10, 24]. The proposed DY-BL network solely matches query and gallery images using global feature embedding while DY-Cace utilizes global as well as the local feature embeddings. Considering low-cost network architecture of the proposed ReID networks, it can be concluded that the dynamic backbone constitutes a promising solution to feature embedding to improve robustness to occlusion and pose changes.

3 Discriminative feature embedding by dynamic backbone

We propose a dynamic backbone network architecture to deliver robust feature embeddings for enhancing more discriminative details of the query and gallery images in person ReID tasks. This is achieved by implementing channel-wise attention and dynamically fusing feature channels in a latent space. SubSect. 3.1 formulates the dynamic convolution executed as the core operation of each convolutional layer. SubSect. 3.2 presents details of our backbone network architecture.

3.1 Dynamic convolution via channel fusion

A robust person ReID system needs to accurately match a query image to the gallery images of the same person captured from different cameras. This requires feature embedding robust to occlusion and abrupt pose changes. However, a standard static convolution layer of a CNN outputs a feature embedding extracted with a fixed receptive field that prevents adaptation to the query of an unseen person or parts of the person. In order to alleviate this problem, we propose using dynamic convolution that enables learning a number of kernels and tuning them to a new query image at the inference stage.

Vanilla dynamic convolution simultaneously learns multiple convolution kernels and applies an attention based aggregation for kernel fusion. Equation 1 formulates the aggregated kernel $\mathbf{W}(\mathbf{x})$ of a dynamic convolutional unit.

$$\mathbf{W}(\mathbf{x}) = \sum_{k=1}^{K} \pi_k(\mathbf{x})\mathbf{W}_k \tag{1}$$

where K is the number of kernels, $\mathbf{W}_k$ is the k^{th} kernel of size $\mathrm{u}\times\mathrm{u}$. $\mathbf{x}$ is the input image at the first layer (the output feature map of the previous layer). $\pi_k(\mathbf{x})$ refers to the dynamic attention coefficient for the k^{th} kernel that models impact of the k^{th} kernel as a function of the input $\mathbf{x}$ [25].

Equation 1 models the dynamic convolution for a one channel input $\mathbf{x}$ of size $\mathrm{M}\times\mathrm{N}$. Hence the number of learnable kernel parameters for a C_{in} input, C_{out} output channel data becomes $\mathrm{K}\times\mathrm{C}_{out}\times\mathrm{C}_{in}\times\mathrm{u}\times\mathrm{u}$. Therefore, although it provides a more discriminative feature embeddings, the number of parameters in a vanilla dynamic convolutional layer is increased by K times compared to the static counterpart. Additionally, the network learns K aggregation parameters where each of them is generated by a specialized branch having extra learnable parameters [19, 25].

In order to minimize the computational load while increasing the performance, in our person ReID backbone architecture, we adapt a decomposition based formulation

proposed in [22]. Specifically, each of the individual kernels is decomposed into the summation of a mean kernel and a residual kernel, as in Eq. 2,

$$\mathbf{W}_k = \mathbf{W}_0 + \Delta\mathbf{W}_k, k \in 1, \ldots, K \tag{2}$$

The mean kernel, denoted as $\mathbf{W}_0$, is computed as the average of individual kernels and can be represented as $\mathbf{W}_0 = \frac{1}{K}\sum_{k=1}^{K}\mathbf{W}_k$. On the other hand, the residual kernel, denoted as $\Delta\mathbf{W}_k = \mathbf{W}_k - \mathbf{W}_0$, captures the deviation of each individual kernel $\mathbf{W}_k$ from the mean.

Inserting Eq. 2 into Eq. 1 and after decomposing the residual part by dynamic convolution decomposition [22], the aggregated kernel $\mathbf{W}(\mathbf{x}) \in \mathrm{R}^{\mathrm{C}_{in}\times u\times u}$ is formulated in tensor form for a C_{in} channel input as in Eq. 3. Specifically, the first and the second term of Eq. 3, respectively, enables the channel-wise attention and the channel fusion in execution of the dynamic convolution

$$\mathbf{W}(\mathbf{x}) = \mathbf{\Lambda}(\mathbf{x})\mathbf{W}_0 + \mathbf{P}\mathbf{\Phi}(\mathbf{x})\mathbf{Q}^T \tag{3}$$

In order to clarify the proposed attention-based and fusion-based derivations, we elaborate the notation given in Eq. 3 for a C_{in} channel input $\in \mathrm{R}^{\mathrm{C}_{in}\times\mathrm{M}\times\mathrm{N}}$. We learn an aggregated kernel to extract the embedding of each output thus the tensor $\mathbf{W}(\mathbf{x}) \in \mathrm{R}^{\mathrm{C}_{out}\mathrm{x}\mathrm{C}_{in}\times(u\times u)}$ denotes the aggregated kernel shown in Eq. 3, where C_{out} refers the number of output channels.

We apply the channel-wise attention to localize the discriminative features extracted at different channels of embedding. Hence $\mathbf{W}_0 \in \mathrm{R}^{\mathrm{C}_{out}\mathrm{x}\mathrm{C}_{in}\times(u\times u)}$ shown in Eq. 3 models the mean kernels in tensor form. Specifically, for each input data channel we learn a $(u \times u)$ mean kernel and to simplify the notation we consider 1×1 kernels for the rest of this subsection. Thus the convolution of $\mathbf{X}$ by $\mathbf{W}_0$ yields $\mathbf{S} \in \mathrm{R}^{\mathrm{C}_{out}\times\mathrm{M}\times\mathrm{N}}$.

Hence a diagonal matrix $\mathbf{\Lambda}(\boldsymbol{x}) \in \mathrm{R}^{\mathrm{C}_{out}\mathrm{x}\mathrm{C}_{out}}$ (Eq. 3) is learned during the training where $\boldsymbol{\alpha}_i$ is the attention factor assigned to the output channel i (Eq. 4). In Eq. 4, the matrix $\mathbf{A} \in \mathrm{R}^{\mathrm{C}_{out}\times\mathrm{M}\times\mathrm{N}}$ models the output of the channel-wise attention module.

$$\mathbf{A} = \mathbf{\Lambda}(\mathbf{x})\mathbf{S} = \begin{bmatrix} \boldsymbol{\alpha}_1 & 0 & \ldots & 0 \\ 0 & \boldsymbol{\alpha}_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \boldsymbol{\alpha}_{\mathrm{C}_{out}} \end{bmatrix} \begin{bmatrix} s_1^1 & s_1^2 & \ldots & s_1^{(MxN)} \\ s_2^1 & s_2^2 & \ldots & s_2^{(MxN)} \\ \vdots & \vdots & \ddots & \vdots \\ s_{\mathrm{C}_{out}}^1 & s_{\mathrm{C}_{out}}^2 & \ldots & s_{\mathrm{C}_{out}}^{(MxN)} \end{bmatrix} \tag{4}$$

We also apply a channel fusion to enhance the feature embedding. The second additional term of Eq. 3 models the channel fusion performed in a low dimensional latent space. Specifically, the channel fusion enables learning the residual term in a L dimensional latent space where the constraint $L << \mathrm{C}_{in}$ ensures L^2 is much smaller than its static counterpart KL. Hence, during the training, the dimension of the input $\mathbf{X} \in \mathrm{R}^{\mathrm{C}_{in}\times\mathrm{M}\times\mathrm{N}}$ is lowered to $\mathbf{X}_L^f \in \mathrm{R}^{\mathrm{L}\times\mathrm{M}\times\mathrm{N}}$ by using a learned projection matrix $\mathbf{Q}^T \in \mathrm{R}^{\mathrm{C}_{in}\times\mathrm{L}}$. In the latent space L channels are fused using an input-dependent learnable channel fusion matrix $\mathbf{\Phi}(\mathbf{x}) \in \mathrm{R}^{\mathrm{L}\times\mathrm{L}}$. The dynamic channel fusion matrix $\mathbf{\Phi}(\mathbf{x})$ retains the representation power needed to extract discriminative feature embedding. Afterward, the embedding is projected to the higher dimensional space by a learned upsampling matrix $\mathbf{P} \in \mathrm{R}^{\mathrm{LxC}_{out}}$. that yields the output of channel fusion denoted as $\mathbf{F} \in \mathrm{R}^{\mathrm{C}_{out}\times\mathrm{M}\times\mathrm{N}}$. Finally, the feature embedding tensor $\mathbf{E} \in \mathrm{R}^{\mathrm{C}_{out}\times\mathrm{M}\times\mathrm{N}}$ is extracted by slice-wise summation of $\mathbf{A} \in \mathrm{R}^{\mathrm{C}_{out}\times\mathrm{M}\times\mathrm{N}}$ and $\mathbf{F} \in \mathrm{R}^{\mathrm{C}_{out}\times\mathrm{M}\times\mathrm{N}}$.

Figure 3 visually demonstrates the formulated dynamic convolution adapts its convolution kernels to the specific characteristics of the input image (Fig. 3a), resulting in more localized feature embedding (Fig. 3c) without a significant increase in parameter size. Regarding the parameter complexity, assume the number of input and output channels are equal to C and the kernel size is 1×1. Then, static convolution and vanilla dynamic convolution require C^2 and KC parameters, respectively. The fusion based dynamic convolution formulated above requires C^2, CL and CL parameters for the matrices$\mathbf{W}_0$, $\mathbf{P}$ and$\mathbf{Q}$, respectively. An additional $(2C + L^2)C/r$ parameters are required by the dynamic branch to generate $\mathbf{\Lambda}(\mathbf{x})$ and $\mathbf{\Phi}(\mathbf{x})$ where r is the reduction rate of the first FC layer which is set to 16 in this paper. Then the total complexity is about $(\frac{3}{16})C^2$ is much less then $4C^2$ the parameter complexity for vanilla dynamic convolution with $K = 4$ [22].

3.2 Dynamic backbone architecture

Our person ReID network employs the dynamic ResNet50 (DY-ResNet50) as the backbone architecture. DY-ResNet50 is implemented by replacing the static convolutional layers of ResNet50, which is a widely utilized backbone architecture [26], with the dynamic convolutional layers formulated in Sect. 3.1.

The DY-ResNet50 network comprises four execution stages (Fig. 2a) and each stage outputs five feature maps with 64, 256, 512, 1024, and 2048 channels, respectively. Moreover, each stage includes two bottlenecks, the fundamental building blocks of residual networks. In the DY-ResNet50 architecture, like static ResNet50 [26], each bottleneck consists of three convolution kernels and shortcut connections. Figure 2b illustrates the first bottleneck of stage 1 of the DY-ResNet50 architecture. The reason that we keep one of the projection shortcuts with a static kernel is to initialize the feature maps. Table 9 in Appendix A presents the architecture of DY-ResNet50 in detail.

The dynamic convolution kernel executes the channel-wise attention and the channel fusion formulated in

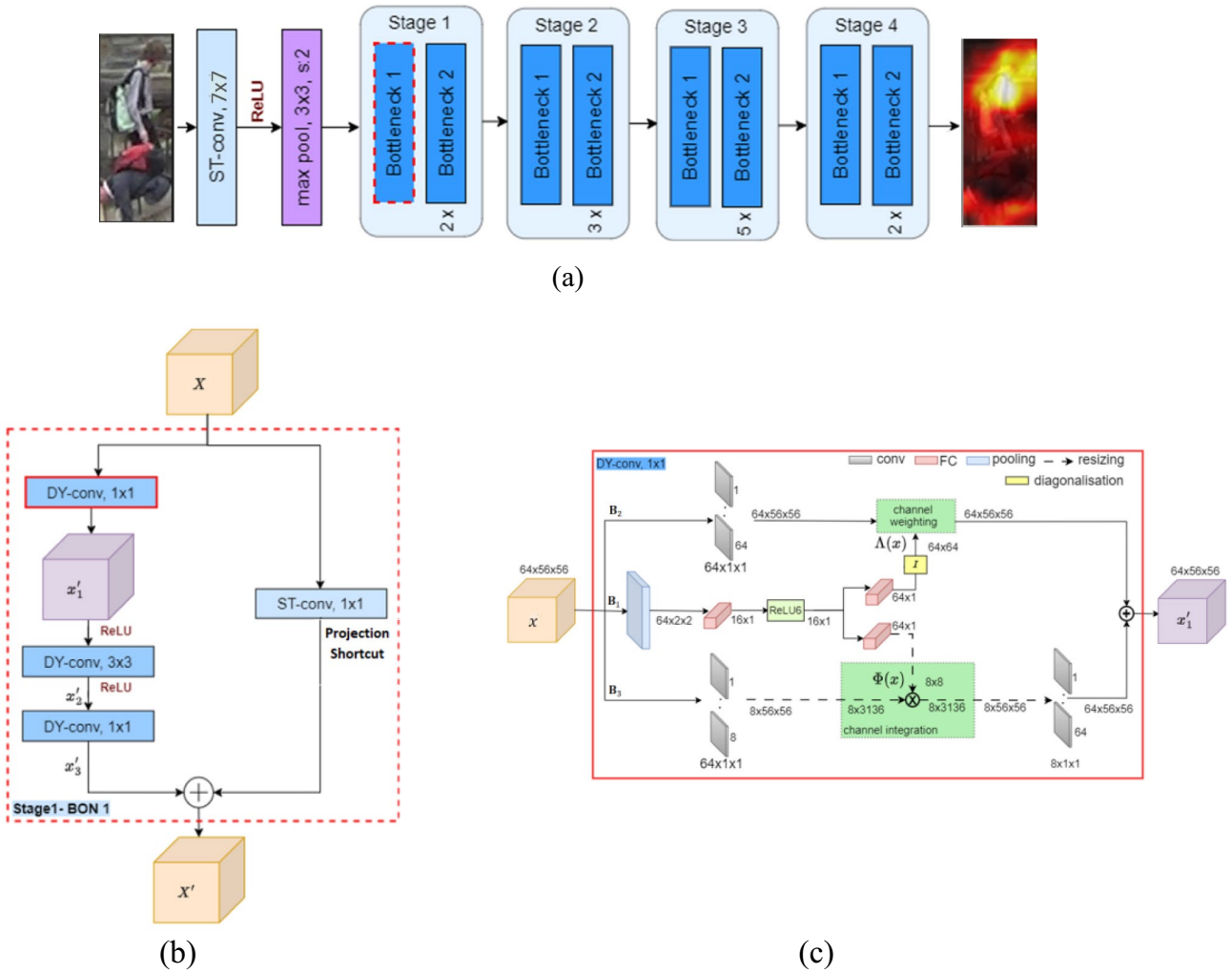

Fig. 2 **a** Architecture of DY-ResNet50. **b** First bottleneck of Stage1 **c** First dyn conv layer of Stage1-bottleneck1

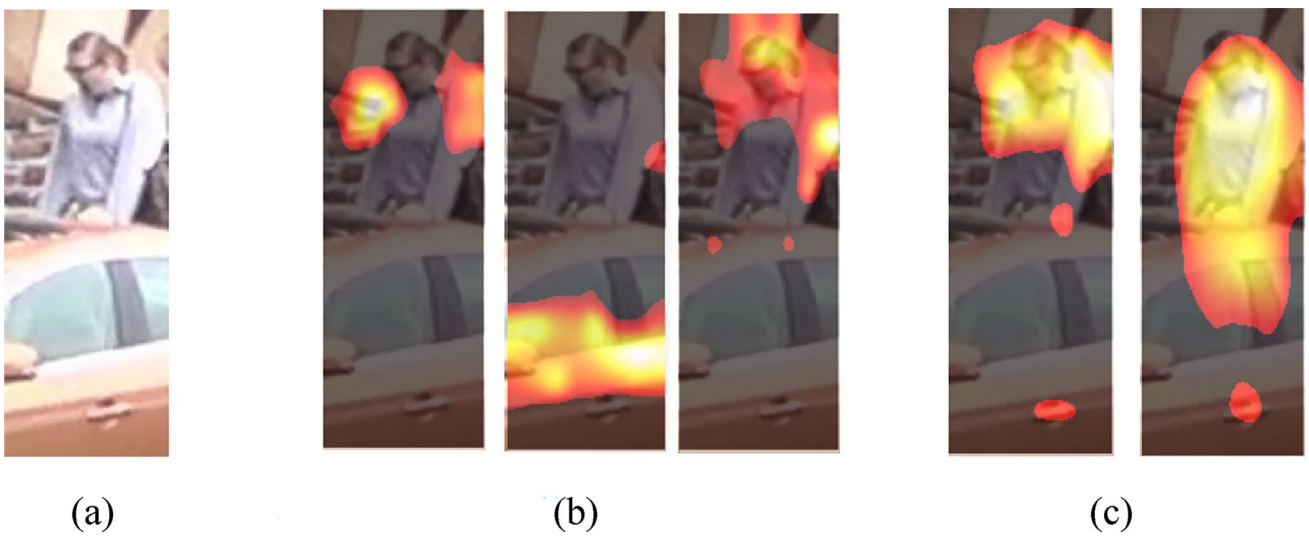

Fig. 3 **a** Original query image. **b** DY-ResNet50 features at channel 600, 1000 and 1700. **c** Feature embedding of DY-ResNet50 and ST-ResNet50

Fig. 4 **a** Robustness to occlusion and pose change. **a–c** Original query image, feature embedding of DY-ResNet and ST-ResNet50 (left to right)

subSect. 3.1. Main execution blocks for the first dynamic convolution kernel of the first bottleneck at stage 1 are illustrated in Fig. 2c. During the training stage, the input $\mathbf{X} \in R^{C_{in} \times M \times N}$ is passed through three branches. For each output channels, **B1** called the dynamic branch enables learning the channel-wise attention weight tensor $\mathbf{\Lambda}(\mathbf{x})$ and the channel-fusion tensor,$\mathbf{\Phi}(\mathbf{x})$. Branch **B2** serves in joint learning of the average kernel $\mathbf{W}_0 \in R^{C_{in} \times (u \times u)}$. Similarly **B3** is the branch that serves in learning the down and up sampling matrices, **Q** and**P**, respectively. At the inference stage, the dynamic convolution operates as follows: Branch **B1** generates the matrices $\mathbf{\Lambda}(\mathbf{x})$ and $\mathbf{\Phi}(\mathbf{x})$ depending on the given input **X**. In particular, for all query as well as the gallery images, the channel-wise attention and channel fusion weights are dynamically adapted to the input. Over the branch **B2**, the channel-wise attention is applied on the input convolved with $\mathbf{W}_0$ is weighted by $\mathbf{\Lambda}(\mathbf{x})$ to generate the channel weighted output. Furthermore, a third branch **B3**, referred as the residual kernel branch, applies channel fusion on **X** in the latent space. Finally, the output feature map is obtained by summing the outputs of the average kernel branch and the residual kernel branch.

The input-adaptive nature of dynamic convolution provides more discriminative features and improves the representation power of the network. Figure 3a shows a query image from OccludedDukeMTMC-ReID dataset. Figure 3b illustrates the feature embeddings generated at channels 600, 1000 and 1700 (left to right) by DY-ResNet50 output of our end-to-end trained DY-BL ReID network. Effectiveness of the channel-wise attention and channel fusion is clearly observable form the features that localized discriminative regions of the input. Combined final feature embeddings generated by DY-ResNet50 and ST-ResNet50 are shown at Fig. 3c and d, respectively. Figure 4 illustrates the feature embeddings generated by DY-ResNet50 and ST-ResNet50 for three images where high pose change encountered in the first one and in the others the object of interest is occluded. Despite the tolerable increase in the number of parameters of DY-ResNet50 architecture, the input-adaptive nature of dynamic convolution enables more compact feature embedding that provides robustness to occlusion and pose change.

As will be seen in the subsequent sections, despite the slight increase in the number of parameters in the dynamic ResNet50 architecture, the input adaptive nature of dynamic convolution provides more discriminative features and improves the representation power of the network.

4 Person ReID via dynamic convolution

We designed two ReID networks with the objective of robustness to especially occlusion and pose changes with low cost. The first one is a baseline deep network architecture having a dynamic ResNet-50 backbone (Sect. 3.2) and a few ReID head layers on top of the backbone. The network referred as DY-BL matches query and gallery images using global feature embeddings. The second one employs the same backbone, DY-ResNet-50, and integrates global as well as the local embeddings to improve the representation power of ReID feature embeddings. Hence the second person ReID network referred as DY-Cache has a more complex ReID head architecture but still low cost compared to most of the existing deep learners. This Section presents the proposed DY-BL and DY-Cache ReID networks by giving reference to their static counterparts.

4.1 DY-BL: person ReID via global feature embedding

Network architecture of DY-BL takes a commonly used ReID network as the baseline. Specifically it is designed by replacing the static convolutional kernels of the baseline

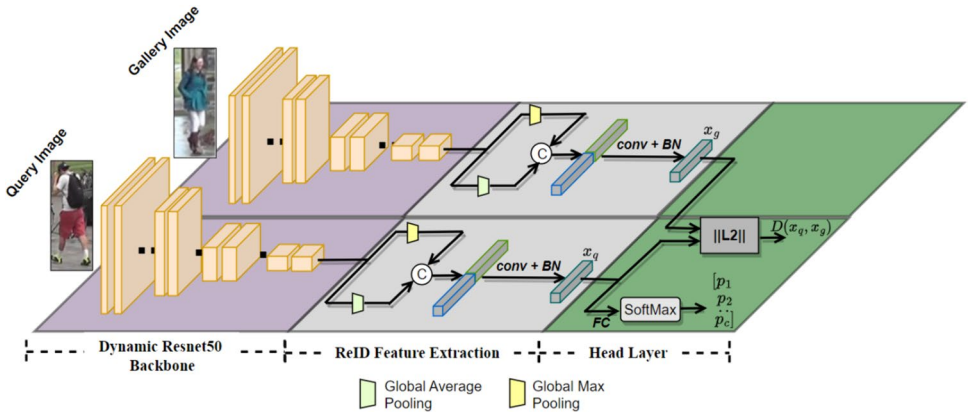

Fig. 5 DY-BL training architecture

ReID network with dynamic counterparts and end-to-end trained on different ReID datasets. Figure 5 illustrates DY-BL network architecture, where a ReID feature extraction head that personalizes the feature embedding extracted by DY-Resnet-50 backbone, is placed at the top of the backbone. As shown in Fig. 5, a query and gallery image pair is fed into the DY-ResNet50 backbone where it outputs the global individual feature maps with 2048 channels. These individual feature maps are then passed through two parallel pooling layers: an average pooling layer and a maximum pooling layer. The output of these pooling layers are concatenated to generate the final global feature embedding after passing through a convolutional layer followed by the batch normalization (BN). x_q and x_g respectively denote the final global feature embedding of the query and gallery image. The DY-BL network head layer (Fig. 5) includes a regressor that works on the triplet branch to output cosine distance for the input image pair. Also a SoftMax layer produces the estimated class score vector for each gallery image. At the inference, each query is matched to a number of gallery images ranked based on the cosine distance between the embeddings x_q and x_g, each having the size of 728. During the training of DY-BL, a bach of query and gallery image pairs are fed into the network and a training procedure explained in the following is applied.

The components of DY-BL network are jointly trained for multi-task learning (e.g. classification and re-identification), similar to its static counterpart. Hence the total loss function of the DY-BL network is formulated as in Eq. 5

$$L_{BL} = L_{LS-CE} + L_{Htri} \tag{5}$$

The first term of the loss function, L_{LS-CE}, is known as the label smooth cross entropy loss [27], which is utilized to improve ID classification accuracy by penalizing incorrect predictions and encouraging more confident and accurate predictions. It can be formulated as in Eq. 6.

$$L_{LS-CE} = -\frac{1}{C}\sum_{i=1}^{C}\left((1-\varepsilon)y^{(q)} + \frac{\varepsilon}{C}\right)\log(p_i^{(q)}) \tag{6}$$

In Eq. 6, C represents the number of different person IDs (classes), $y^{(q)}$ is the ground truth label for the corresponding query image (where y is 1 for the correct class and 0 otherwise), ε is a hyper-parameter for label smoothing, and it is set to $\varepsilon = 0.1$ for both ST-BL and DY-BL. $p_i^{(q)}$ denotes the predicted class probability of the query image for the i.

The second term of Eq. 5, L_{Htri}, is known as the hard triplet loss [28], which is employed to enhance the re-identification performance. Equation 7 formulates the hard triplet loss.

$$L_{Htri} = \sum_{i=1}^{P}\sum_{q=1}^{R}[m + \max_{p=1\ldots R} D\left(\mathbf{x}_q^i, \mathbf{x}_{g_p}^i\right) - \min_{\substack{j=1\ldots P \\ n=1\ldots R \\ j\neq i}} D(\mathbf{x}_q^i, \mathbf{x}_{g_n}^i)] \tag{7}$$

where R is the number of query images collected from each person ID and P denotes the total number of different person IDs included in a batch. m refers to the margin hyper-parameter and is set to $m = 0.5$ for both ST-BL and DY-BL.

As in the conventional form, the hard triplet loss aims to minimize the distance $D\left(\mathbf{x}_q, \mathbf{x}_{g_p}\right)$ between the query embedding $\mathbf{x}_q$ and embedding of the positive gallery sample $\mathbf{x}_{g_p}$ while maximizing the distance to the embedding of negative sample, $D(\mathbf{x}_q^i, \mathbf{x}_{g_n}^i)$. In our training for each batch, the positive sample is chosen as the one having the same ID but the highest cosine distance to the query whereas the negative sample is taken as the closest one with a different ID.

We trained the ST-BL and DY-BL networks in an end-to-end manner, initializing the backbone with a pretrained model trained as a classifier on the ImageNet dataset [29]. In DY-BL architecture, we also observed the performance by replacing the static convolutional kernels of ReID feature extraction head with the dynamic convolutional kernels.

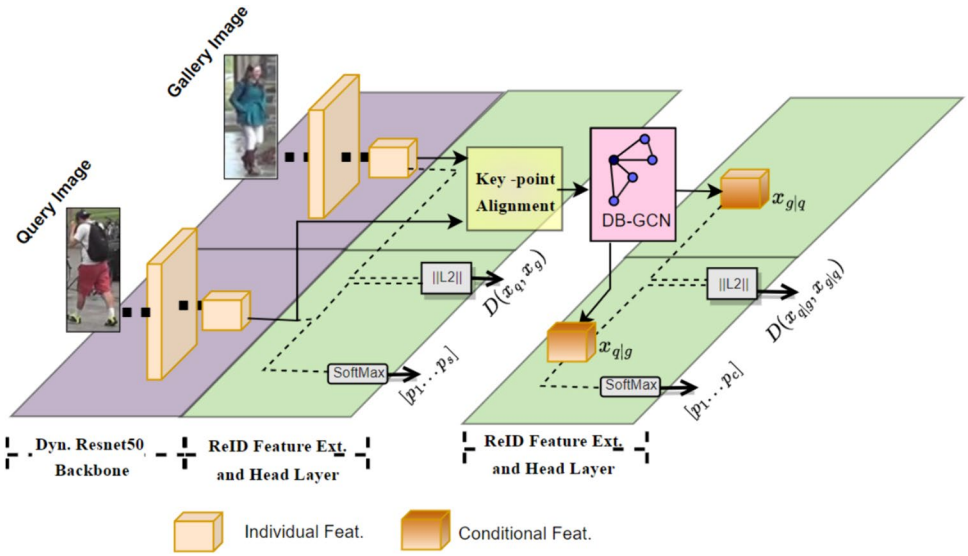

Fig. 6 DY-Cace training architecture

However these replacement did not provide a significant improvement on the overall performance. Hence we utilized dynamic convolution layers only in the backbone network architecture.

4.2 DY-Cace: ReID via integration of global and conditional feature embeddings

In ReID tasks, relying solely on global information for matching is not reliable especially when the target person is occluded or the pose change is high. To deal with these challenges, it is common to employ global as well as local features. With this objective, we designed our second ReID network by taking CaceNet (Clue Alignment and Conditional Embedding) [10, 11] as the baseline. In addition to the global information, CaceNet employs the conditional embedding to dynamically adjust the query and gallery features. Moreover the pairwise correspondence attention and discrepancy-based graph convolutional network are also integrated with the ReID pipeline resulting in efficient embedding. The new ReID network referred as DY-Cace comprises the DYBL network presented in Sect. 4.1. Figure 6 illustrates the training network architecture of DY-Cace where DY-BL is placed in the network as the first two stages. The dynamic backbone architecture is DY-ResNet50 (Fig. 6) as in DYBL. Moreover DY-Cace has extra modules to work on local correspondences of query and gallery image pairs.

In particular, the feature maps generated by DY-ResNet50 for the query and gallery images are fed into the key-point alignment (KPA) stage of the network for further processing. KPA employs a correspondence attention module that outputs the crucial matching locations within the individual feature maps as well as between the query and gallery feature maps. After filtering outliers, the selected matching points are fed into a graph convolutional network that generate the conditional feature embeddings. In this section we highlight our contribution on the baseline CaceNet and the detailed formulation can be found in [11].

DY-Cace ReID network is trained end-to-end to minimize the loss function shown in Eq. 8, similar to its static counterpart ST-Cace.

$$L_{Cace} = L_{LS-CE} + L_{Htri} + L_{mixup} + L_{Htri_{cond}} \tag{8}$$

The first two terms of the loss function are formulated same as in Eq. 6 and Eq. 7 and models the label smooth cross entropy loss L_{LS-CE} and the hard triplet loss L_{Htri}, respectively.

Impact of the local feature embeddings are modeled by the two additional loss terms, L_{mixup} and $L_{Htri_{cond}}$. In particular, L_{mixup} referred as the mix-up loss is calculated by Eq. 9

$$L_{mixup_i} = \sum_{q=1}^{P} \sum_{g=1}^{R} \alpha L_{CE}(y^{(q)}, \mathbf{x}_{q|g}) + (1-\alpha) L_{CE}(y^{(g)}, \mathbf{x}_{q|g}) \tag{9}$$

where $\mathbf{x}_{q|g}$ represents the conditional feature map of the query image conditioned on the gallery. Similarly $\mathbf{x}_{g|q}$ represents the conditional feature map of the gallery image conditioned on the query. $y^{(q)}$, and $y^{(g)}$, are the ground truth labels of the query and the gallery, respectively. α denotes the mix-up coefficient and it is set to 0.9 in training of both the DY-Cace and ST-Cace architectures. Furthermore, $L_{Htri_{cond}}$ shown in Eq. 8, denotes the hard triplet loss and it is calculated by Eq. 7 where the feature embeddings are replaced by the conditional feature embeddings.

We conducted training on four different datasets with varying difficulty levels for both the ST-Cace and DY-Cace networks, as detailed in Sect. 5. In the inference step, both

ST-Cace and DY-Cace follow the same procedure as ST-BL and DY-BL. To simplify the networks during inference, only the individual embedding stage is utilized, and the images are matched solely based on the individual feature vectors. Our evaluation results reported in Sect. 5 demonstrate that, in spite of the ReID performance improvement achieved by DY-Cace, it slightly increases the parameter complexity. In particular the number of learnable parameters of DY-Cace is 34M where DY-BL has 31M parameters. This is mainly because of the global feature embedding and the conditional feature embedding branches share the DY-ResNet50 backbone parameters.

5 Performance evaluation

In this section, we report overall performance of the proposed dynamic ReID networks, DY-BL and DY-Cace, compared to their static counterparts as well as the state-of-the-art. Both static and dynamic ReID networks are trained and tested on Market1501 [30], DukeMTMC-reID [31] and CUHK03 [32], which are three widely used ReID datasets having different difficulty levels. To evaluate robustness to occlusion and pose changes, the networks are trained and tested on challenging Occluded Duke-reID [33] data set. After summarizing the content of each dataset, the detailed results are reported in the following subsections.

Market-1501 dataset [30] consists of 32,668 images of 1,501 identities captured by six different cameras. The training set comprises 12,936 images of 751 identities, while the testing data includes the remaining images of 750 identities.

DukeMTMC-reID dataset [31] contains 36,411 images of 1,404 identities captured by eight different cameras. The training set consists of 16,522 images of 702 identities, while the test set includes 2,228 query images and 17,661 gallery images of 702 identities.

CUHK03 dataset [32] provides manually labeled bounding boxes for 14,096 images captured by six different cameras. It comprises a total of 1,467 identities, with 767 identities used for training and the remaining identities for testing.

Occluded-DukeMTMC dataset [33] is derived from the DukeMTMC-reID dataset. It is characterized by the presence of occlusions in 9% of the training images, 10% of the gallery images, and all the query images. This dataset includes 15,618 training images, 2,210 query images, and 17,661 gallery images, making it one of the largest datasets for occluded person ReID.

Implementation details We employed the SGD with momentum optimizer to train each network, with weight decay and momentum values set to 5×10^{-4} and 0.9, respectively. The total number of epochs for both networks was set to 80. The initial learning rate was set to 5×10^{-2} for the ST-BL and DY-BL networks, and 6.25×10^{-3} for the ST-Cace and DY-Cace networks. The learning rate was increased linearly from 0 to the initial learning rate value during the first 5 epochs, using the cosine method similar to [11]. Subsequently, the learning rate was gradually decreased to 0 by the end of the training. For the ST-BL and DY-BL networks, a batch size was set to 128, while for the ST-Cace and DY-Cace networks, a batch size of 16 was used. To incorporate the dynamic backbone, which was pre-trained on the ImageNet dataset, we modified the code available at https://github.com/liyunsheng13/dcd and integrated it to the Baseline[1] and CaceNet ReID networks [34].

5.1 Evaluation metrics

In our evaluations we employed mAP, Rank-k accuracy, and Pr_k, three of the commonly used metrics in re-identification tasks. Furthermore two novel metrics, mAP_l and first-l-accuracy are formulated. By introducing these metrics, we aim to provide a more comprehensive and nuanced evaluation of the matching capabilities, allowing for a deeper understanding of the performance of ReID systems.

Mean average precision (mAP) mAP is a conventional evaluation metric employed to assess the overall ReID performance [1]. As shown in Eq. 10, mAP quantifies the average precision where AP_q denotes the average precision for person ID q and Q is the total number of individual person IDs.

$$\text{mAP} = \frac{1}{\text{Q}} \sum_{q=1}^{Q} AP_q \tag{10}$$

The average precision, AP_{q}, for each query person ID q is by Eq. 11.

$$AP_q = \frac{1}{N_q} \sum_{i=1}^{n} \text{I}_i Pr_i \tag{11}$$

where N_q represents the number of gallery images associated with the query ID q, n denotes the total number of matching executed to correctly retrieve all the gallery images having ID q. Note that n reflects the ReID performance, higher the matching accuracy smaller the n. I_i is an indicator function that takes a value of 1 if the i^{th} matching has ID q and 0 otherwise. Precision, Pr_i quantifies the fraction of correctly matched gallery images up to the i^{th} matching. For a comprehensive analysis, we also report Pr_k which measures the percentage of correct matches in the top-k ranked results. If the query matches k positive samples within the first k matching, Pr_k is set to 1. Otherwise, it ranges between 0

[1] https://github.com/TencentYoutuResearch/PersonReID-YouReID

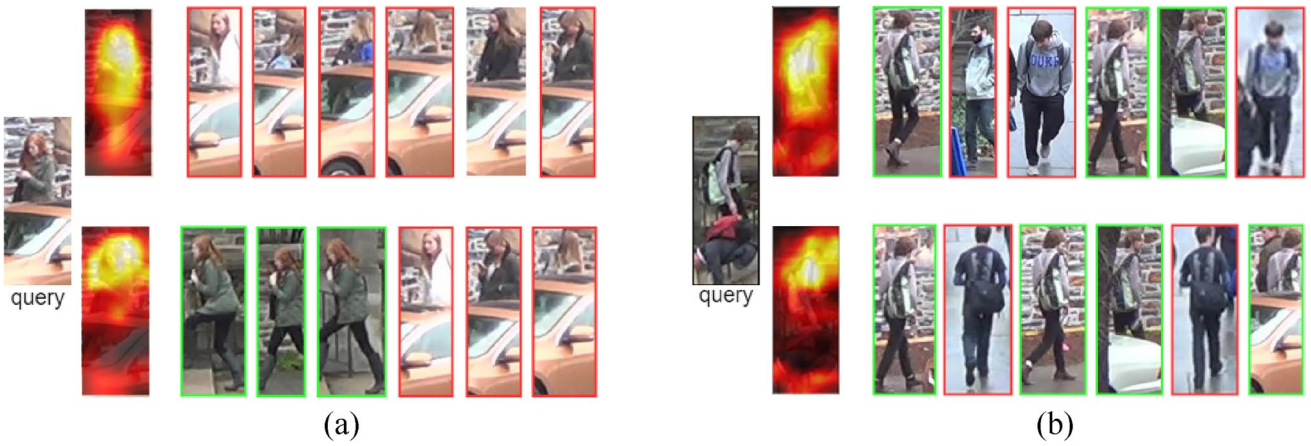

Fig. 7 Identification results for the query image with **a** ID 90 and **b** ID 35 taken from Occluded-DukeMTMC. For both query images, the first row illustrates the individual feature embedding extracted at the last layer of static ResNet-50 backbone and first-6 matching results of ST-BL. The second row illustrates the feature embedding extracted by dynamic ResNet-50 backbone and first-6 matching results of DY-BL. The green boxes represent the true matches while red boxes indicate the false matches

and 1, indicating the proportion of true matches within the top-k range.

mAP$_l$: mAP requires execution of a new matching till reaching to all images assessed to the query ID q. However, this is not trackable especially when the gallery is large. It is also important to evaluate fraction of the correctly matched IDs within a short search period. Therefore, we formulate a novel metric mAP$_l$ as in Eq. 12 by fixing the l, number of correctly matched gallery samples.

$$\text{mAP}_l = \frac{1}{\text{Q}} \sum_{q=1}^{Q} \frac{1}{l} \sum_{i=1}^{y} \text{I}_i Pr_i \tag{12}$$

where y denotes the total number of matching executed to correctly retrieve l gallery images for ID q.

rank-k accuracy Another well-known metric in person re-identification is Rank-k accuracy, which represents the probability of at least one correct sample ID match in the top-k ranked samples for a given query [1]. In re-identification tasks, it is common to report rank-1 accuracy, which quantifies the model's ability to correctly identify the accurate match from the entire gallery as the highest-ranked result.

first-l accuracy The proposed metric quantifies the time spent for the first l^{h} correct match of a query ID within the top-y ranked samples by reporting the *l/y* ratio where the y varies depending on l. Differing from rank-k accuracy and Pr_k, first-l-accuracy metric aims to give credit to the speed of matching.

5.2 Robustness to occlusion and pose change

In order to demonstrate the superiority of dynamic convolution in effectively addressing the challenges associated with occluded person ReID, in this section, we conduct a comparative analysis of the proposed DY-BL/DYCace, with ST-BL/ST-Cace as well as the state-of-the-art methods. Therefore the proposed person ReID networks are trained and tested on OccludedDukeMTMC which is one of the largest datasets for occluded person ReID. Results reported numerically and visually demonstrate the backbone network is a substantial module in feature extraction therefore our ReID networks with the dynamic ResNet-50 backbone are capable of extracting localized features.

In order to demonstrate the essential role of the backbone network in the extraction of discriminative feature embedding so as the accurate person reidentification, we report the results obtained with the proposed low cost DYBL ReID where the architecture is designed by adding a few head layers on top of a dynamic backbone, DY-ResNet-50. As a visual illustration, Fig. 7a and b display two query images selected from the Occluded-DukeMTMC dataset,[2] along with the extracted feature embedding and matched gallery images obtained by DY-BL (second row) and its static counterpart ST-BL (first row). We observe that in hard cases, in particular when the query is highly occluded Fig. 7a or under pose change (Fig. 7b, DY-BL is able to extract much localized feature embedding for the query person that enables more accurate matching. Numerically, for ID 90, as illustrated in Fig. 7a, mAP achieved by ST-BL is reported as 14.84% while it is increased to 83.93% by DY-BL.

We report our detailed evaluation results on Occluded-DukeMTMC dataset at Table 1. Not that for comparison with existing work, all performance metrics are reported

[2] https://github.com/lightas/Occluded-DukeMTMC-Dataset

Table 1 mAP (%) and rank-1 (%) achieved by ST-BL and DY-BL on DukeMTMC-reID dataset

		10	20	30	40	50	60	70	80
mAP	ST-BL	24.47	27.61	30.45	36.22	41.73	44.79	47.50	47.96
	DY-BL	25.06	32.52	33.43	40.18	43.12	47.11	49.75	50.20
rank-1	ST-BL	35.58	36.65	38.96	44.52	51.49	54.25	55.84	57.06
	DY-BL	34.12	41.36	42.35	50.09	53.35	58.57	59.91	60.14

Table 2 Matching speed of DY-BL compared to ST-BL (Occluded-DukeMTMC dataset)

	k/l	Pr_k(%)	first-l (%)	mAP_l(%)
ST-BL	1	57.1	64	64
	5	54.7	54	58.9
	10	56.3	49.7	59.3
	20	58.6	45.1	61.2
DY-BL	1	3.1	2.7	2.7
	5	2.5	2	2.4
	10	2.5	2.4	2.5
	20	1.1	1.1	1.2

as percentages (%) even though their original scale ranges from 0 to 1. To clarify the impact of dynamic network on the learning speed and matching capability, performance achieved by the model after 80 epochs training as well as the earlier training stages are reported with mAP and rank-1 metrics. According to the test results reported after 80 epochs training, DYBL provides 2.25% improvement in mAP and 3.8% improvement in rank-1, compared to ST-BL. Moreover, the dynamic network increases the convergence speed and respectively leads to a 4% and 5.5% higher mAP and rank-1 accuracy, after 40 epochs training.

In addition to the overall re-identification accuracy, we have also evaluated the impact of dynamic learning on the query matching speed, specifically the accuracy achieved for the first l true matching. Table 2 reports the performance achieved by DY-BL and ST-BL on the Occluded-DukeMTMC dataset in terms of Pr_k and the proposed metrics mAP_l and first-l-accuracy. As shown in Table 2, the dynamic network consistently demonstrates higher performance across various metrics. Note that k / l is increased at most to 20 to keep the person re-identification speed reasonable for real-time applications. Specifically, when the value of l is set to 1 and 10, the first-l-accuracy of DY-BL demonstrates 2.7% and 2.4% improvements, respectively. These findings demonstrate the capability of DY-BL to identify the most relevant 1 or 10 individuals faster. On the other hand, with the metric Pr_k, which quantifies the percentage of true matches within the top-k rank, 3.1% and 2.5% increase are reported for $k=1$ and $k=5$, respectively.

Table 3 Overall person ReID performance of the proposed ReID networks compared to the state-of-the-art

Method	mAP	rank-1
PCB [4]	33.8	42.6
PGFA [33]	37.3	51.4
HOReID [35]	43.8	55.1
OAMN [36]	46.1	62.6
Visibility-aware [15]	46.3	62.2
SGSFA [37]	47.4	62.3
ST-BL	48.0	57.1
EGMN [7]	49.2	60.5
DY-BL	50.2	60.1
CaceNet [11]	50.8	58.8
ISP [6]	52.3	**62.8**
DY-Cace	**53.6**	**62.8**

Boldface to show the method providing the highest performance

To conduct a detailed assessment of our proposed methods against other advanced techniques, Table 3 provides a comprehensive comparison of inference performances between static and dynamic networks, along with the performance achieved by several state-of-the-art methods. As can be seen from Table 3, DY-Cace outperforms most of the existing methods including static Baseline. Moreover, DY-BL attains a ReID performance comparable to that of CaceNet [11] which has a more complex architecture. Considering low cost network architecture of the proposed ReID networks, it can be concluded that the dynamic backbone constitutes a promising solution to feature embedding to improve robustness to occlusion and pose changes.

5.3 Overall performance

In order to report a comprehensive evaluation of the performance, apart from the challenging Occluded-DukeMTMC dataset, we trained and tested both the proposed dynamic networks as well as their static counterparts on different datasets having more generalized content. All evaluations are performed on Market-1501, DukeMTMC-reID and

Table 4 Performance of ST-BL and DY-BL on DukeMTMC-reID and ST-Cace and DY-Cace on CUHK03 dataset (mAP (%))

		10	20	30	40	50	60	70	80
Duke	ST-BL	53.26	58.28	63.03	67.63	70.97	74.86	76.51	76.70
	DY-BL	55.51	63.92	67.10	69.34	74.88	77.25	78.46	79.01
CUHK	ST-Cace	39.04	55.91	61.35	70.92	76.17	78.77	79.47	80.03
	DY-Cace	44.43	60.07	65.64	73.79	79.64	80.50	80.86	82.07

Table 5 Performance of ST-BL and DY-BL on DukeMTMC-reID and ST-Cace and DY-Cace on CUHK03 dataset ((rank-1(%))

		10	20	30	40	50	60	70	80
Duke	ST-BL	71.36	75.72	79.62	82.85	84.47	86.89	87.30	87.48
	DY-BL	74.55	78.68	83.03	83.62	87.21	88.20	89.09	89.41
CUHK	ST-Cace	38.93	57.0	63.43	73.07	77.93	80.50	80.79	82.43
	DY-Cace	44.43	60.07	65.64	73.79	79.64	80.50	80.86	82.07

CUHK03 datasets, however, because of the space limitation, the most informative results are reported for different test cases. Comparative results with the-state-of-the-art are also reported.

5.3.1 Matching accuracy

We followed the test cases described in Sect. 5.2 for a fair comparison and first focused on the ReID accuracy evaluated by mAP at different stages of the training. Hence the models generated at each 10 epochs interval are employed at the inference where the training is completed in 80 epochs. Table 4 illustrates the performance achieved by the proposed DY-BL and its static counterpart ST-BL on DukeMTMC-reID dataset (first row). Moreover, the second row of Table 4 presents the accuracy achieved by the proposed DY-Case and its static counterpart ST-Cace. In particular, we observe a significant accuracy improvement for the Baseline model, amounting to 5.6% and 2.3%, respectively, when assessing the models trained for 20 and 80 epochs on DukeMTMC-reID dataset. Moreover, as shown in Table 4, DY-Cace exhibits a noticeable performance enhancement compared to ST-Cace when trained on CUHK03 dataset. Specifically, it achieves a significant increase of 5.4% and 2% on the 10th and 80th epochs, respectively. Since the dynamic networks are designed to better encode the important features of the dataset by allowing the parameters of each convolution kernel to be adjusted dynamically based on the input query image, this enhances the learning capability of ReID network and improves the model's ability to acquire useful representations in early stages of training.

Table 6 Matching speed of DY-Cace achieved on DukeMTMC-reID dataset compared to ST-Cace

	k/l	Pr_k(%)	first-l (%)	mAP_l(%)
ST-Cace	1	90	92.6	92.6
	5	88.3	86.9	90.1
	10	86.4	80.3	88.4
	20	82.2	64.4	85.2
DY-Cace	1	+1.5	+0.8	+0.8
	5	+0.3	+0.2	+0.5
	10	+0.1	+0.6	+0.4
	20	+1.3	+2.1	+0.8

5.3.2 Ranking accuracy

In addition to the matching accuracy, we focus on reporting the rank-1 accuracy, which quantifies the model's ability to correctly identify the accurate match from the entire gallery as the highest-ranked result. In Table 5, we present a comparative analysis of the rank-1 accuracy obtained from static and dynamic networks on DukeMTMC-reID and CUHK03 datasets for Baseline and CaceNet models, respectively. As in the preceding section, we report the results for every 10th epoch to demonstrate that the dynamic network exhibits superior rank-1 accuracy and compared to its static counterparts, particularly at earlier training stages. This observation highlights the dynamic network's ability to learn rapidly and converge more efficiently. We have observed a significant accuracy improvement for the Baseline model on DukeMTMC-reID dataset, with an increase of 3.2% at epoch 10 and 2% at epoch 80. Furthermore, Table 5 presents the performance comparison between DY-Cace and ST-Cace achieved by the model trained on CUHK03

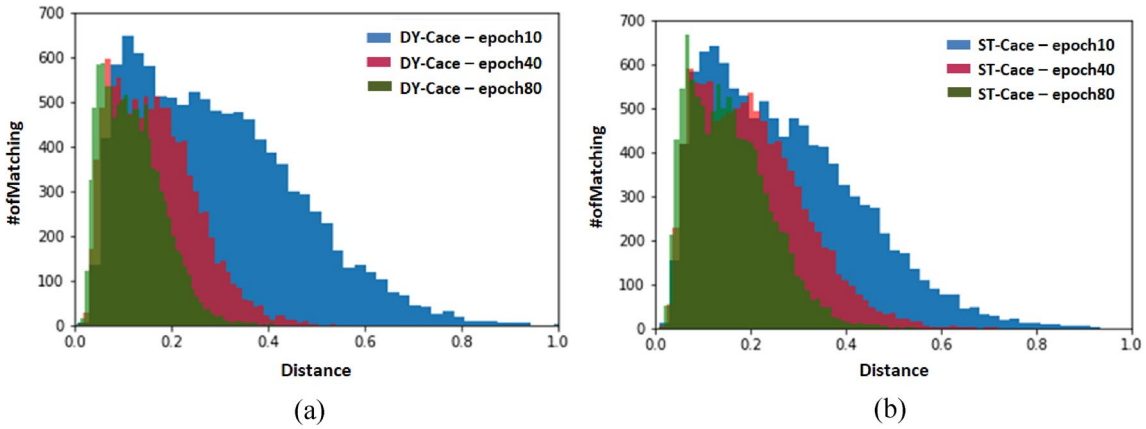

Fig. 8 Distribution of the cosine distance registered in matching the query and gallery images of CUHK03 dataset by **a** DY-Cace and **b** ST-Cace

dataset. More specifically, differences are 5.5% and 1.7% on 10th and 50th epoch, respectively, while both have similar accuracy at epoch 80.

On the other hand, the rank-1 accuracy focuses solely on determining whether the most confident match is within the top-1 ranks or not. This approach excludes the evaluation of other matches, potentially overlooking valuable information. Also by observing how the performance changes in terms of the metrics Pr_k, first-l-accuracy and mAP_l with different values of k / l, we gain a more understanding of the model's ranking capabilities and its effectiveness in identifying the most relevant matches (Table 6). When considering the Pr_k and first-l-accuracy metrics, it is observed that our proposed method, DY-Cace, achieves an improvement of 1.3% and 2.1% for k / l = 20 compared to STCace, respectively. This indicates that for a given query image, our method is more efficient at accurately matching the first-l gallery images in a shortened top-k order. Moreover, DY-Cace demonstrates a slight improvement of 0.8% in mAP_l when compared to ST-Cace. This demonstrates that our approach provides better precision among the first-l correct matching compared to its static counterpart.

5.3.3 Confidence of matching

In addition to quantify the accuracy of different person ReID models according to the metrics formulated in sub-Sect. 5.1, we have also investigated the confidence of matchings that highlight trustability of the ReID model. This is achieved based on the cosine distance metric. Specifically, during the inference the head layer of the proposed ReID network outputs a 768-dimensional feature embedding for each query as well as the gallery image. The query image is matched with the gallery images ranked in terms of the cosine distance between the query and gallery embedding where the one with the lowest distance is retrieved as the rank-1 matching. This implies that lower the cosine distance higher the confidence for that matching. Hence we first investigated the cosine distances calculated by rank-1 matching for all the queries in each dataset. For CUHK03 dataset, a significant number of correct matches, specifically 99.43%, are identified as having a lower matching distance in comparison to the static ReID network. This percentage is calculated as 96.92%, 80.58% and 57.90% for Occluded-DukeMTMC, Market-1501, and DukeMTMC-reID, respectively. These findings demonstrate that the utilization of the dynamic backbone network architecture in CaceNet significantly enhances the confidence of matching during the inference.

Another test case is conducted by evaluating statistics of the matching distances. Figure 8 illustrates distribution of the cosine distances registered for correctly matched feature embeddings. The distributions are plotted by repeating the matching at different stages of the training, in particular after 10, 40 and 80 epochs training (blue, red and green plots). It is evident that for both DY-Cace (Fig. 8a) and ST-Cace (Fig. 8b) networks, the feature embeddings become more discriminative from 10 to 80 epochs training. This implies that both the dynamic and static networks increase the confidence of matching by reducing the distance between embeddings as the training progresses. Additionally, in both Fig. 8a and b, the distributions for epoch 80 (green plots) demonstrate that the variance for DY-Cace is smaller than that of the static model. This indicates that the dynamic network much more effectively reduces inter-class variability by pushing the embeddings of the queries from same person ID (class) closer. This is because the dynamic backbone of DY-Cace is capable of adapting the convolutional kernels to the input query image that yields more compact ID feature embeddings. As a result, the dynamic network exhibits a

Table 7 Comparison of the performance achieved by the proposed DY-BL, DY-Cace and state-of-the-art ReID networks

Method	DukeMTMC-reID		CUHK03		Market-1501	
	mAP	rank-1	mAP	rank-1	mAP	rank-1
MGCAM [8]	–	-	50.2	50.1	74.3	83.9
MLFN [38]	62.8	81.0	49.2	54.7	74.3	90.0
MSLG [39]	65.0	82.0	44.0	70.0	71.0	90.0
PCB-RPP [40]	69.2	83.3	–	–	81.6	93.8
CBN + BoT [41]	70.1	84.8	–	–	83.6	94.3
Mancs [42]	71.8	84.9	63.9	69.0	82.3	93.1
VMP [43]	72.6	83.6	–	–	80.8	93.0
SNR [44]	72.4	84.4	–	–	84.7	94.4
CAMA [45]	72.9	85.8	66.5	70.1	84.5	94.7
CASN [14]	73.7	87.7	68.0	73.7	82.8	94.4
IANet [46]	73.4	87.1	–	–	83.1	94.4
SPReID [47]	73.3	85.9	–	–	83.4	93.7
DSA [48]	74.3	86.2	75.2	78.9	87.6	95.7
DG-Net [49]	74.8	86.6	–	–	86.0	94.8
HOReID [35]	75.6	86.9	–	–	84.9	94.2
EGMN [7]	75.6	87.7	–	–	84.6	94.3
Baseline	77.2	88.3	–	–	87.6	94.8
MHN [50]	77.2	89.1	72.4	77.2	85.0	95.1
BAT-net [51]	77.3	87.7	76.1	78.6	87.4	94.1
MGN [52]	78.4	88.7	67.4	68.0	86.9	95.6
ABD-Net [53]	78.6	89.0	–	–	88.3	95.6
RelationNet [54]	78.6	89.7	75.6	77.9	88.9	95.2
PISNet [55]	78.7	88.8	–	–	87.1	95.6
Pyramid-101 [5]	79.0	89.0	76.9	78.9	88.2	95.7
SCSN [56]	79.0	90.1	83.3	**86.3**	88.5	95.7
DY-BL	79.0	89.4	73.1	75.5	88.3	95.5
RGA-SC [57]	–	–	77.4	81.1	88.4	96.1
CaceNet [11]	81.3	90.9	–	–	**90.3**	96
OPReID [58]	81.3	91.1	–	–	89.3	**96.1**
DY-Cace	81.6	**91.5**	**80.1**	82.1	89.9	95.7
TransReID [12]	82.0	90.7	–	–	88.9	95.2
PFD [13]	**83.2**	91.2	–	–	89.7	95.5

For each dataset the best score is boldfaced. "–" denotes the result is not reported for the corresponding dataset. "–"

Table 8 Complexity and inference time of the proposed DY-BL and DY-Cace compared to the existing ReID networks

	ST/DY-BL	ST/DY-Cace	TransReid [12]	PFD [13]
# Parameters	27M/31M	31M/34M	103M	138M
ModSize(MB)	210.1/242.6	217.3/248.6	395	530
InferTime(s)	55.7/79.5	55.7/79.5	253.3	427.9

Inference times are evaluated on DukeMTMC-reID dataset

tendency to identify the correct gallery image with a lower distance score, enhancing its matching confidence.

5.3.4 Performance compared to State-of-the-art

We evaluated the overall re-identification performance of the proposed DY-BL and DY-Cace ReID architectures against several state-of-the-art ReID networks on DukeMTMC-reID, CUHK03, and Market-1501 datasets. Some of these methods such as Pyramid and RelationNet, employ part-based models, some of them including VPM, HOReID

and SCSN, RGA-SC apply alignment and attention-based approaches, respectively.

Results comparing the proposed DY-BL, DY-Cace, and existing methods are reported in Table 7. In spite of its low cost architecture, DY-BL consistently achieves comparable performance compared to the existing methods. Moreover, DY-Cace which has a more complex architecture, emerges as one of the top three methods across all datasets. In particular, DY-Cace achieves a 0.3% increment in rank-1 compared to the top-performing PFD method on the DukeMTMC-reID dataset, while it takes the third place in terms of mAP.

In order to emphasize the strengths of the proposed methods, at Table 8 we present the parameter complexity, the model size and the inference time of DY-Cace and top two methods listed in Table 7. Note that, inference times for all methods are obtained by NVIDIA Tesla T4 GPU with batch size 256. The execution times reported at Table 8 correspond to the inference time on DukeMTMC-reID dataset. Specifically, both TransReid and PFD are transformer based methods and provide slightly better performance with the expense of three or four times higher learnable parameters and significantly higher inference times. Moreover, although DY-Cace achieves comparable performance on all datasets, it provides the highest rank-1 accuracy on DukeMTMC-reID dataset, and top mAP on CUHK03. The comparative results reported across different datasets highlight the effectiveness of the dynamic network architecture in capturing input-specific features and enhances the discriminative power of the model for accurate person re-identification.

6 Conclusions

This paper presents a deep re-identification framework that leverages dynamic convolution with channel fusion in the backbone architecture. We investigate the impact of dynamic convolution, using two ReID networks of varying complexity; a simpler network with fewer layers, DY-BL, and a more complex architecture, DY-Cace. Our study employs the ResNet50 backbone, known for its exceptional feature extraction performance. However, the proposed architecture can be integrated into any convolutional neural network used in ReID.

In this paper, we demonstrate the superiority of our proposed DY-BL and DY-Cace over their static counterparts ST-BL and ST-Cace across all datasets. Dynamic convolution enhances the ability to acquire useful representations even in early training stages by extracting more discriminative features. Comprehensive results, reported on three person re-ID datasets by comparing to state-of-the-art methods, show the effectiveness of our model. The proposed method also outperforms most of the existing ReID methods on the occluded dataset which demonstrates the ability of dynamic convolution to solve challenging cases like occlusion.

Additionally, we introduce two novel evaluation metrics designed to assess ReID performance, with an emphasis on the highest-ranked correct matches. We believe that this contributes to the advancement of the field. Consequently, all results attest to dynamic networks' potential as a powerful and efficient solution for their adoption in future ReID research and applications.

Existing re-identification methods are mostly designed for a specific domain that results in significant performance loss at different domains. It would be possible to achieve domain adaptation by including a fine tuning scheme into the proposed re-ID network that may improve the accuracy and robustness of person re-identification systems. Moreover, recurrent neural networks could potentially enhance the model's ability to handle temporal dependencies in video-based person re-identification tasks. Therefore, integration of the dynamic convolution with RNN architectures could be considered as a future direction.

Appendix A

Table 9 gives the detailed dimension of matrices and kernels employed in DY-ResNet50 compared to ResNet50 network. In static ResNet50, each bottleneck consists of three convolution kernels (enclosed in squared brackets), while DY-ResNet50 incorporates a combination of static convolution kernels and dynamic matrices that is shown for first dynamic convolutional kernel within the first bottleneck of stage 1 in Fig. 2c. Notably, the number of input and output channels and the sizes of input and output in dynamic ResNet50 are the same with the sizes in the corresponding layers of the static ResNet50 as shown in Table 9.

Table 9 Comparison of ResNet-50 and DYN-ResNet-50 in terms of parameters of the network architecture

Out.Size	Static Conv	Dynamic Conv	
112 *x* 112	7 *x* 7, 64 stride 2 3 *x* 3, max pool stride 2		
		Bottleneck1	Bottleneck2 *x* 2
56 *x* 56	$\begin{bmatrix} 1x1, 64 \\ 3x3, 64 \\ 1x1, 256 \end{bmatrix}$	$\begin{bmatrix} \begin{pmatrix} \Lambda(\mathbf{x}) : 64x64 \\ \mathbf{W}_0 : 1x1, 64 \\ Q : 1x1, 8 \\ \Phi(\mathbf{x}) : 8x8 \\ P : 1x1, 64 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 64x64 \\ \mathbf{W}_0 : 3x3, 64 \\ Q : 3x3, 8 \\ \Phi(\mathbf{x}) : 8x8 \\ P : 1x1, 64 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 256x256 \\ \mathbf{W}_0 : 1x1, 256 \\ Q : 3x3, 8 \\ \Phi(\mathbf{x}) : 8x8 \\ P : 1x1, 256 \end{pmatrix} \end{bmatrix}$	$\begin{bmatrix} \begin{pmatrix} \Lambda(\mathbf{x}) : 64x64 \\ \mathbf{W}_0 : 1x1, 64 \\ Q : 3x3, 16 \\ \Phi(\mathbf{x}) : 16x16 \\ P : 1x1, 64 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 64x64 \\ \mathbf{W}_0 : 3x3, 64 \\ Q : 1x1, 8 \\ \Phi(\mathbf{x}) : 8x8 \\ P : 1x1, 64 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 256x256 \\ \mathbf{W}_0 : 1x1, 256 \\ Q : 1x1, 8 \\ \Phi(\mathbf{x}) : 8x8 \\ P : 1x1, 256 \end{pmatrix} \end{bmatrix}$
7 *x* 7	$\begin{bmatrix} 1x1, 512 \\ 3x3, 512 \\ 1x1, 2048 \end{bmatrix}$	$\begin{bmatrix} \begin{pmatrix} \Lambda(\mathbf{x}) : 512x512 \\ \mathbf{W}_0 : 1x1, 512 \\ Q : 1x1, 32 \\ \Phi(\mathbf{x}) : 32x32 \\ P : 1x1, 512 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 512x512 \\ \mathbf{W}_0 : 3x3, 512 \\ Q : 1x1, 22 \\ \Phi(\mathbf{x}) : 22x22 \\ P : 1x1, 512 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 2048x2048 \\ \mathbf{W}_0 : 3x3, 2048 \\ Q : 1x1, 22 \\ \Phi(\mathbf{x}) : 22x22 \\ P : 1x1, 2048 \end{pmatrix} \end{bmatrix}$	$\begin{bmatrix} \begin{pmatrix} \Lambda(\mathbf{x}) : 512x512 \\ \mathbf{W}_0 : 1x1, 512 \\ Q : 1x1, 45 \\ \Phi(\mathbf{x}) : 45x45 \\ P : 1x1, 512 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 512x512 \\ \mathbf{W}_0 : 3x3, 512 \\ Q : 1x1, 22 \\ \Phi(\mathbf{x}) : 22x22 \\ P : 1x1, 512 \end{pmatrix} \\ \begin{pmatrix} \Lambda(\mathbf{x}) : 2048x2048 \\ \mathbf{W}_0 : 3x3, 2048 \\ Q : 1x1, 22 \\ \Phi(\mathbf{x}) : 22x22 \\ P : 1x1, 2048 \end{pmatrix} \end{bmatrix}$

Square brackets indicate bottlenecks and Λ, W_0, Q, Φ and P are Dy-conv. Parameters

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). The authors did not receive support from any organization for the submitted work.

Data availability The datasets analyzed during the current study are available in the, Market-1501dataset https://www.v7labs.com/open-datasets/market-1501, DukeMTMC-reID dataset https://exposing.ai/duke_mtmc/, Occluded-DukeMTMC-Dataset https://github.com/lightas/Occluded-DukeMTMC-Dataset and CUHK03 dataset https://www.ee.cuhk.edu.hk/~xgwang/CUHK_identification.html.

Code availability The datasets and source code are accessible at https://github.com/msprITU/DY-REID.

Declarations

Conflict of interest The authors declare that they have no conflict of interests.

References

1. Ye M, Shen J, Lin G, Xiang T, Shao L, Hoi SCH (2022) Deep learning for person re-identification: A survey and outlook. IEEE Trans on PAMI 44(6):2872–2893
2. Ming Z, Zhu M, Wang X, Zhu J, Cheng J, Gao C, Yang Y, Wei X (2022) Deep learning-based person re-identification methods: A survey and outlook of recent works. Image Vis Comput 119:104394
3. Ning E, Wang C, Zhang H, Ning X, Tiwari P (2024) Occluded person re-identification with deep learning: a survey and perspectives. Expert Syst Appl 239:122419
4. Sun Y, Zheng L, Yang Y, Tian Q, Wang S (2018) Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In: Proceedings of the European conference on computer vision (ECCV), pp. 480–496

5. Zheng F, Deng C, Sun X, Jiang X, Guo X, Yu Z, Huang F, Ji R (2019) Pyramidal person re-identification via multi-loss dynamic training. In: Proceedings of the IEEE CVPR, pp. 8514–8522
6. Zhu K, Guo H, Liu Z, Tang M, Wang J (2020) Identity-guided human semantic parsing for person re-identification. In: Proceedings of the ECCV, pp. 346–363
7. Zhou S, Zhang M (2023) Occluded person re-identification based on embedded graph matching network for contrastive feature relation. Pattern Anal Appl 26:487–503
8. Song C, Huang Y, Ouyang W, Wang L (2018) Mask-guided contrastive attention model for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1179–1188
9. Yu Y, Yang S, Hu H, Chen D (2022) Attention-guided multi-clue mining network for person re-identification. Neural Process Lett 54(4):3201–3214
10. Yu F, Jiang X, Gong Y, Zhao S, Guo X, Zheng W-S, Zheng F, Sun X (2021) Devil's in the details: aligning visual clues for conditional embedding in person re-identification. In: Proceedings of the IEEE CVPR
11. Yu F, Jiang X, Gong Y, Zheng W-S, Zheng F, Sun X (2022) Conditional feature embedding by visual clue correspondence graph for person reidentification. IEEE Trans on Image Processing 31:6188–6199
12. He S, Luo H, Wang P, Wang F, Li H, JiangW (2021) Transreid: Transformer-based object re-identification. In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 15013–15022
13. Wang T, Liu H, Song P, Guo T, Shi W (2022) Pose-guided feature disentangling for occluded person re-identification based on transformer. In: Proceedings of the AAAI conference on artificial intelligence, vol. 36, pp. 2540–2549
14. Zheng M, Karanam S, Wu Z, Radke RJ (2019) Re-identification with consistent attentive siamese networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5735–5744
15. Yang J, Zhang J, Yu F, Jiang X, Zhang M, Sun X, ChenY-C, Zheng W-S (2021) Learning to know where to see: a visibility-aware approach for occluded person re-identification. In: Proceedings of the IEEE/CVF International conference on computer vision, pp. 11885–11894
16. Miao J, Wu Y, Yang Y (2021) Identifying visible parts via pose estimation for occluded person re-identification. IEEE Trans Neural Netw Learn Syst 33:4624–4634
17. Luo H, Jiang W, Zhang X, Fan X, Qian J, Zhang C (2019) Alignedreid++: dynamically matching local information for person reidentification. Pattern Recogn 94:53–61
18. Tan L, Dai P, Ji R, Wu Y (2022) Dynamic prototype mask for occluded person re-identification. In: Proceedings of the 30th ACM International conference on multimedia, pp. 531–540
19. Chen Y, Dai X, Liu M, Chen D, Yuan L, LiuZ (2020) Dynamic convolution: attention over convolution kernels. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11030–11039
20. Zhang L, Zhou S Guan J, Zhang J (2021) Accurate few-shot object detection with support-query mutual guidance and hybrid loss. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 14424–14432
21. Liu J, Bao Y, Xie G-S, et al. (2022) Dynamic prototype convolution network for few-shot semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11553– 11562
22. Li Y, Chen Y, Dai X, Liu M, Chen D, Yu Y, Yuan L, Liu Z, Chen M, Vasconcelos N (2021) Revisiting dynamic convolution via matrix decomposition. In: Proceedings ICLR
23. Han Y, Huang G, Song S, Yang L, Wang H, Wang Y (2022) Dynamic neural networks: a survey. IEEE Trans on PAMI 44(11):7436–7456
24. URL-1 https://github.com/TencentYoutuResearch/PersonReID YouReID. access time: 12.05.2022
25. Yang B, Bender G, Le QV, Ngiam J (2019) Condconv: conditionally parameterized convolutions for efficient inference. In: Proceedings NIPS, pp. 1307–1318
26. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778
27. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818–2826
28. Hermans A, Beyer L, Leibe B (2017) In defense of the triplet loss for person re-identification. In: Proceedings IEEE CVPR
29. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Proceedings NIPS, vol. 25
30. Zheng L, Shen L, Tian L, Wang S, Wang J, Tian Q (2015) Scalable person re-identification: A benchmark. In: Proceedings of the IEEE International conference on computer vision, pp. 1116–1124
31. Ristani E, Solera F, Zou R, Cucchiara R, Tomasi C (2016) Performance measures and a data set for multi-target, multi-camera tracking. In: European conference on computer vision, pp. 17–35
32. Li W, Zhao R, Xiao T, Wang X (2014) Deepreid: deep filter pairing neural network for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 152–159
33. Miao J, Wu Y, Liu P, DingY, Yang Y (2019) Pose-guided feature alignment for occluded person re-identification. In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 542–551
34. Akbaba EE (2023) Deep learning via dynamic convolution with channel fusion mechanism. Master's thesis, Istanbul Technical University
35. Wang G, Yang S, Liu H, Wang Z, Yang Y, Wang S, Yu G, Zhou E, Sun J (2020) High-order information matters: Learning relation and topology for occluded person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 6449–6458
36. Chen P, Liu W, Dai P, Liu J, Ye Q, Xu M, Chen Q, Ji R (2021) Occlude them all: occlusion-aware attention network for occluded person re-id. In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 11833–11842
37. Ren X, Zhang D, Bao X (2020) Semantic-guided shared feature alignment for occluded person re-identification. In: Asian Conference on Machine Learning, pp. 17–32
38. Chang X, Hospedales TM, Xiang T (2018) Multi-level factorisation net for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2109–2118
39. Liu J, Tiwari P, Nguyen TG, Gupta D, Band SS (2022) Multi-scale local-global architecture for person re-identification. Soft Comput 26(16):7967–7977
40. Sun Y, Zheng L, Li Y, Yang Y, Tian Q, Wang S (2019) Learning partbased convolutional features for person re-identification. IEEE Trans on PAMI 43(3):902–917
41. Zhuang Z, Wei L, Xie L, Zhang T, Zhang H, Wu H, Ai H, Tian Q (2020) Rethinking the distribution gap of person re-identification with camera-based batch normalization. In: Computer Vision-ECCV 2020: 16th European Conference, pp. 140–157
42. Wang C, Zhang Q, Huang C, Liu W, Wang X (2018) Mancs: a multi-task attentional network with curriculum sampling for

person re-identification. In: Proceedings of the European conference on computer vision (ECCV), pp. 365–381
43. Sun Y, Xu Q, Li Y, Zhang C, Li Y, Wang S, Sun J (2019) Perceive where to focus: Learning visibility-aware part-level features for partial person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 393–402
44. Jin X, Lan C, Zeng W, Chen Z, Zhang L (2020) Style normalization and restitution for generalizable person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3143–3152
45. Yang W, Huang H, Zhang Z, Chen X, Huang K, Zhang S (2019) Towards rich feature discovery with class activation maps augmentation for person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 1389–1398
46. Hou R, Ma B, Chang H, Gu X, Shan S, Chen X (2019) Interaction-and-aggregation network for person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9317–9326
47. Kalayeh MM, Basaran E, Gökmen M, Kamasak ME, Shah M (2018) Human semantic parsing for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1062–1071
48. Zhang Z, Lan C, Zeng W, Chen Z (2019) Densely semantically aligned person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 667–676
49. Zheng Z, Yang X, Yu Z, Zheng L, Yang Y, Kautz J (2019) Joint discriminative and generative learning for person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 2138–2147
50. Chen B, Deng W, Hu J (2019) Mixed high-order attention network for person re-identification. In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 371–381
51. Fang P, Zhou J, Roy SK, Petersson L, Harandi M (2019) Bilinear attention networks for person retrieval. In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 8030–8039
52. Wang G, Yuan Y, Chen X, Li J, Zhou X (2018) Learning discriminative features with multiple granularities for person re-identification. In: Proceedings of the 26th ACM international conference on Multimedia, pp. 274–282
53. Chen T, Ding S, Xie J, Yuan Y, Chen W, Yang Y, Ren Z, Wang Z (2019) Abd-net: attentive but diverse person re-identification. In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 8351–8361
54. Park H, Ham B (2020) Relation network for person re-identification. In: Proceedings of the AAAI conference on artificial intelligence, pp. 11839–11847
55. Zhao S, Gao C, Zhang J, Cheng H, Han C, Jiang X, Guo X, Zheng W-S, Sang N, Sun X (2020) Do not disturb me: person re-identification under the interference of other pedestrians. In: Computer Vision-ECCV 2020 16th European Conference, pp. 647–663
56. Chen X, Fu C, Zhao Y, Zheng F, Song J, Ji R (2020) Yang Y Salience-guided cascaded suppression network for person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3300–3310
57. Zhang Z, Lan C, Zeng W, Jin X, Chen Z (2020) Relation-aware global attention for person re-identification. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition, pp. 3186–3195
58. Yan C, Pang G, Jiao J, Bai X, Feng X, Shen C (2021)Occluded person re-identification with single-scale global representations. In: Proceedings of the IEEE/CVF international conference on computer vision pp. 11875–11884 (2021)