A Strong Baseline for Tiger Re-ID and its Bag of Tricks

Jiwen Yu, Haibo Su, Junnan Liu, Zhizheng Yang, Zhouyangzi Zhang,
Yixin Zhu, Lu Yang, Bingliang Jiao
School of Computer Science, Northwestern Polytechnical University, Xi'an, China
vvictoryuki@163.com, {2018262230,liu2270054721}@mail.nwpu.edu.cn
302949125@mail.nwpu.edu.cn, 601738935@qq.com
{zhu_yixin,lu.yang,bingliang.jiao}@mail.nwpu.edu.cn

Paper ID 20

Abstract

As an instance-level recognition task, person re-identification methods always calculate local features by horizontal pooling. It is based on a simple assumption that pedestrians always stand vertically. But as to wildlife re-identification task, we can not make similar assumption since the various view-angles of wildlife. In this paper, we propose a novel dynamic partial matching method. In our module, global feature learning benefits greatly from local feature learning, which performs an alignment/matching by flipping local features and calculating the shortest path between them. Besides the partial matching method, we also consider a series of data augmentation methods such as flip as new id, random whitening, random crop and so on. And we also use an example sampling strategy, i.e., hard negative mining, for training. In addition, we ensemble the models with different backbones and epochs using imagenet pre-trained models. Extensive experiments validate the superiority of our method for tiger Re-ID. Code has been released at https://github.com/vvictoryuki/tiger_reid_pytorch.

1. Introduction

Tiger re-identification (Re-ID) task aims to match tiger appearing in different non-overlapping camera views, which has raised increasing attention in the field of wildlife monitoring and conservation. As an instance-level recognition problem, this task extract discriminative features of each image. Similar to common re-identification tasks such as Person Re-ID [24, 11, 26] and Vehicle Re-ID [15, 12, 6, 21], tiger Re-ID has noisy background and illumination in various monitoring systems. Each tiger has almost the same pattern and shape, which makes tiger Re-ID more challenging. Besides, the number of pictures captured from different perspectives is usually unbalanced. For example, the side face of the tiger is most easily captured by cameras. In this paper, we propose an enhanced Re-ID model guided by metric learning and ID classification jointly and our model won the third place in Plain Re-ID Track of CVWC2019 challenge.

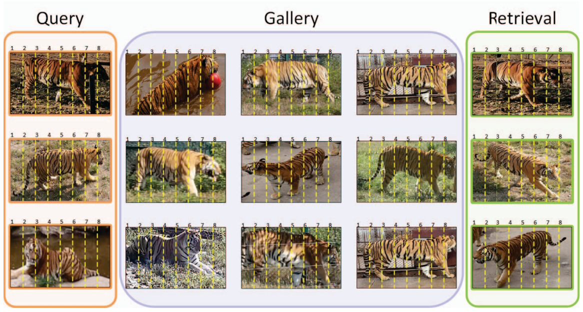


Figure 1: Illustration of retrieval results. Given a query image, we need to pick out similar images across cameras. Besides global feature, we add local feature to help training our model. We compare the similarity of the parallel part of tigers such as head, body and tail.

For a Re-ID task, given a query image, we need to find instances which have same identity in a big database. Therefore, we need a suitable metric learning method which can be applied to compute similarities between the query items and gallery items. An adaptive similarity metric can improve the performance of image matching. Here, we obtain the image features by DCNN and choose the L2 Euclidean distance to compute a similarity score for retrieval task. After that we use triplet loss to pull the features from positive pairs closer and push the features from negative pairs far away. Besides, the cross entropy loss is used to make our model robust and improve its performance.

Our fundamental model can only provide the global feature of each image. Inspired by [24], we take the spatial layout of tigers into consideration. However, as each tiger having different posture, it is difficult to split images in a particular sequence to locate different parts. Here we propose a robust alignment method. In the training phase, we add a branch for learning local features. As shown in Figure 1, the head of a tiger is may towards left or right. To address the partial matching problem, we flip the local features of each image and compute two local distances, one of which uses the flipped features and another uses the original features. In the end, we choose the smaller one as the last local distance which will be used to calculate the triplet loss. Besides, we apply a batch normalization layer after global average pooling layer and achieve improvement. In the inference phase, same as [24], we abandon the local branch and only use the global feature.

For augmenting the tiger dataset, we flip pictures which have the same identity left and right as a new ID for training set. Besides, we choose random whitening and random crop to expand our training set. To improve discriminant ability of our model, we use a specific sampling strategy, *i.e.*, hard negative mining strategy. We use global distance to find the hard samples and make the majority of train set comprised of hard samples.

We train and evaluate our models on the ATRW dataset [7], and employ a specific kind of ensemble strategy to count up distance matrices of models which have different setting for final ranking. Finally we won the third place in CVWC2019 challenge. After obtaining the image features, we adopt the L2 Euclidean distance as measurement for ranking. To further improve Re-ID accuracy, we employ an additional re-ranking [20, 28, 1]. To sum up, our main contributions are as follows:

- We propose a robust alignment method to extract local features by selecting minimum local distances of flipped features and original features.
- We apply a series of data augmentation methods such as flip as new id, random whitening and crop.
- We use global distance to find the hard samples and make the majority of train set comprised of hard samples for hard negative mining.
- We employ a specific kind of ensemble strategy by adding distance matrices for different models up.

2. Related Work

In this section we briefly review recent relevant re-identification works in the fields of metric learning [13, 9, 25, 2] and multi-feature learning [3, 17, 14, 16, 22]. We focus our discussion tiger Re-ID on metric learning and obtain a set of diverse triplet loss with multi-feature which make the same identities close to each other, while different identities are pushed away.

Metric learning. Many re-identification methods based on metric learning show promising performances. S. Paisitkriangkrai *et al.* [13] aims at maximizing the relative distance between images of different individual and optimizes the probability that any of these top k matches are correct using structured learning. Liao *et al.* [9] propose an effective feature representation which is called Local Maximal Occurrence (LOMO), and a subspace and metric learning method called Cross-view Quadratic Discriminant Analysis (XQDA). Zheng *et al.* [25] explore a transfer local relative distance comparison (t-LRDC) model to solve the open-world person re-identication problem by one-shot group-based verication. And Cheng *et al.* [2] propose a transfer metric learning method to simultaneously learn the similarity measurement from different scenarios. Some methods use identification loss also render significant improvement. Lin *et al.* [10] propose a siamese attention architecture that jointly learns spatio-temporal video representations and their similarity metrics. Zheng *et al.* [27] propose a new siamese network that concurrently calculates identification loss and verification loss which learns a similarity measurement and a discriminative embedding at the same time and make full usage of the annotations.

Multi-feature learning. Global features extract the most discriminative clues of the whole image but may fail to capture discriminative local details. Thus, many approaches make use of both the global and local feature. Li *et al.* [8] explore a CNN architecture for Learning Multi-Loss (JLML) of both global and local discriminative feature optimisation subject simultaneously to the same Re-ID labelled information. Wang *et al.* [19] design a feature learning structure combining global and local information in different granularities. Su *et al.* [4] propose PDC(Pose-driven Deep Convolutional) model learns the global representation characterizing the whole body and local representations characterizing body parts simultaneously. Yang *et al.* [23] explore a novel attention-driven multi-branch network which learns robust and discriminative human representation from global whole-body images and local body-part images concurrently. Fu *et al.* [5] exploit average and max pooling strategies to explain person-specic discriminative information in a global-local manner. Wang *et al.* [18] propose a novel Relative Local Distance (RLD) method that integrates a relative local distance constraint into convolutional neural networks (CNNs) which is the rst time that the relative local constraint is proposed to guide the global feature representation learning.

In this paper, we propose a new framework which ex-

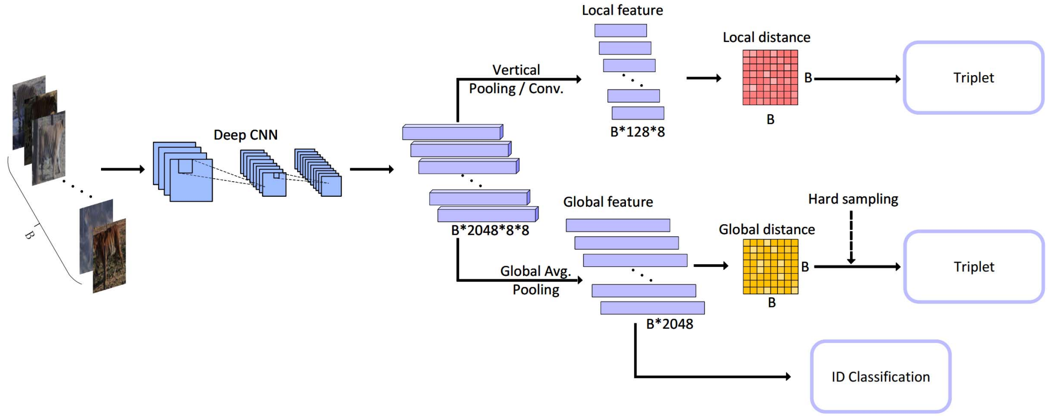


Figure 2: Framework of the Tiger Re-ID. We can use the same feature map from the Deep CNN to get the global distance and the local distance. The global feature is applied by the global average pooling and the local feature is applied by the average pooling in the horizontal direction.

tract global feature and local features of images on CNN, optimize these features with Triplet loss and id classification loss and apply several methods for data augmentation such as flip, random whitening, random crop and so on. Besides, we proposed an example sampling strategy for training using hard negative mining. Finally, we ensemble models with different backbones and training degree.

3. Methods

In this section, we present our solution details.

3.1. Overall Structure

The overall structure is shown in Figure 2. We resize image to 256×256, and use a pre-trained CNN model to extract the feature map which is the output of the last convolution layer. After that, we use two different kinds of pooling layers, *i.e.*, vertical pooling and global average pooling to extract the global features and local features respectively. At last, we simply use euclidean distance of global features as global distance. And we use the method which proposed in [24] to get local distance from local features. Apart from triplet loss, we also apply the classification loss to our architecture. During the training stage, triplet loss from local features, triplet loss from global features and classification loss are all taken into consideration. Each loss will be given a fixed wight and then added together. To make the solution simple, the fixed wight we choose for each loss is just roughly equal.

3.2. Global Representation

As shown in Figure 2, for each feature map ($2048 \times 8 \times 8$), we use global average pooling to generate the global feature (2048×1). The global distance is simply used the euclidean distance between the global features.

For triplet loss, we choose the most dissimilar one with the same ID and the most similar one with different ID to obtain a triplet. There are two reasons that we only use global distances to mine hard samples. First, there is no significant difference in mining hard samples using both distances. Second, the computation speed of global features is much faster than that of local features.

3.3. Local Representation

In order to get local features, We divided the feature map into 8 parts in the horizontal direction, and then do average pooling at each part. Then we use a 1×1 convolution to reduce the channel number from 2048 to 128. In this way, we get 8 local features as shown in Figure 2), the dimension of local feature for each image is 128×8. Each local feature represents a part of the image.

For the local distance, we dynamically match the local parts from left to right to find the alignment of local features with the minimum total distance. When computing the local distance, we assume that the same part between two tigers is comparatively similar. However, for each picture, we can only see one side of the tiger. And we don't know if the heads of the two tigers are in the same direction (left or right). We flip the local features of each image and compute a pair of local distances, one of the local distances uses the flipped features and the other one uses the original features. The final local distance is the smaller one between them.

The following are the details of computation of local distance. First, we divide local feature into eight parts. The first image is marked as a_1,a_2...a_8, and the second image is marked as b_1,b_2...b_8. And then we normalize the distance to

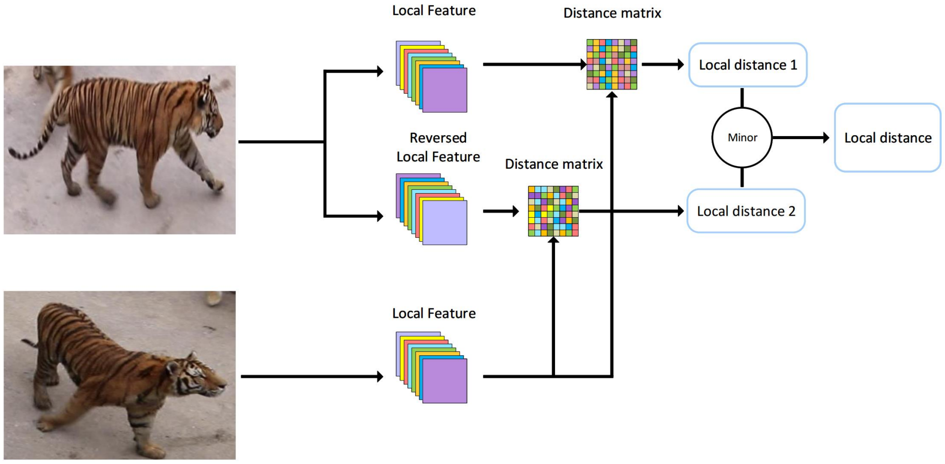


Figure 3: Example of the calculation of the local distance. We compute the distance matrix of the original photos first, and then compute the distance matrix between the photo and the inverse photo. Then we apply the dynamic programming method to both two matrices to get two corresponding image distances, after which we choose the smaller one of the two image distances as the final local distance.

$[0, 1)$ by an element-wise transformation:

$$d_{i,j} = \frac{e^{||a_i - b_j||_2} + 1}{e^{||a_i - b_j||_2} - 1}, i, j \in 1, 2...8, \tag{1}$$

In this formula, $d_{i,j}$ is the distance between the i-th vertical part of the first image and the j-th vertical part of the second image. After getting a local distance matrix D , we can calculate the image distance through dynamic programming as follows:

$$S_{i,j} = \begin{cases} d_{i,j}, & i = 1, j = 1, \\ S_{i-1,j} + d_{i,j}, & i \neq 1, j = 1, \\ S_{i,j-1} + d_{i,j}, & i = 1, j \neq 1, \\ \min(S_{i-1,j}, S_{i,j-1}) + d_{i,j}, & i \neq 1, j \neq 1, \end{cases} \tag{2}$$

where $S_{i,j}$ is the distance of the shortest path when walking from (1,1) to (*i,j*) in the distance matrix D. $S_{8,8}$ is the local distance between two images. The whole procedure is shown in Figure 3.

As shown in Figure 4, they are samples of the same tiger. The top and the middle photos are original photos, and the bottom photo is flipped from the first picture. It should be noted that the flipped images doesn't mean we really flip this image in our training phase, this flipped images is just for demonstration because, in practice, we flip the local features to achieve the same effect.

For the local loss, it is a triplet loss computed with the local distance. We will show its excellent performance in 4.3.

3.4. Metric Learning and ID Classification

For id classification, we define an C-classes problem in which C is the number of IDs in train set and choose cross-entropy loss, which is employed as follows:

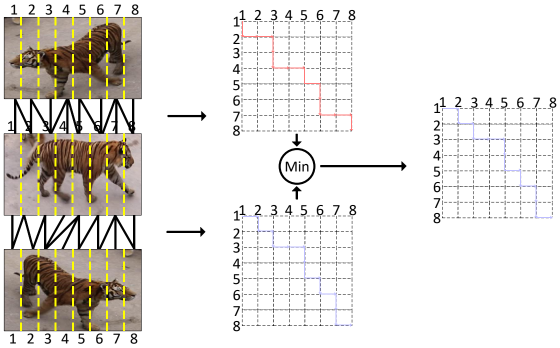


Figure 4: The grid shows the corresponding alignment of photos. It is clear that if the tigers' heads towards different directions, the alignment will fail and we can choose the images with the same head direction by comparing the image distances.

$$L_{ce}(x, gt) = -\log(\frac{\exp(x[gt])}{\sum_i(x[i])}), \tag{3}$$

where x is the prediction vector and gt is the index of the ground truth.

In order to deal with the similarity of different features, we construct the distance function manually by selecting appropriate features in a specific task. Metric learning can autonomously learn the metric distance function for a specific task according to different tasks.

The number of IDs in training set is relatively small, using only ID loss may cause the model to overfit the training set, so we combined id loss and triplet loss together to train the ReID models. We use Imagenet pre-trained models, such as Resnet50, to extract the feature maps, and then use global average pooling to get feature vectors, which will be used to find the nearest neighbor category according to metric Learning as the matching item.

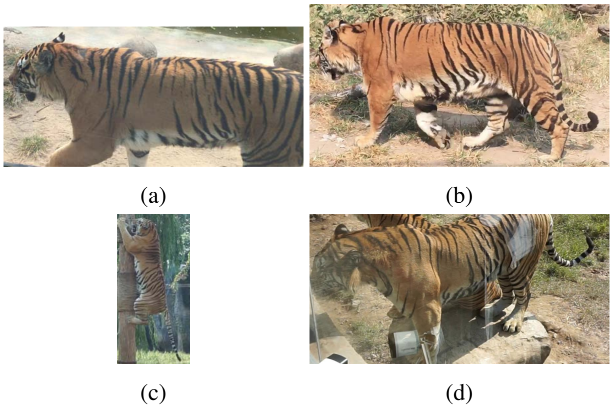
3.5. Data Augmentation

In plain Re-ID task, it has been challenging as various backgrounds in the images from different cameras while same backgrounds in same cameras, which may mislead our model to learn the background feature rather than the stripe pattern information. Besides, the distributions of these identities in the training set are extremely unbalanced which may worsen the performance of id classification. For this reasons, we propose several suitable methods for data augmentation.

Flip as New ID. Discrimination among tigers can be considered as a problem of identifying stripe patterns. It will be another tiger if flip the image, so we apply the flip as new id method to our data augmentation. Using this method, we not only enlarge the size of training set, but also increase the number of images of different IDs in the same camera case. As flip not change the background information such as lightness and environment, the model trained with the new data seems to have a better ability to extract local features from images. Compared with global feature, the local feature represent the stripe pattern better in tiger Re-ID task.

Random Whitening and Crop. As mentioned above, the key to perform well in tiger Re-ID task is learning local feature particularly the stripe pattern. In consideration of practicability, we choose random whitening and random crop as the main data augmentation method to make the model learn how to extract local feature more.

The reason why we choose these methods is due to two considerations: First, the main background information contains the environment and the pose of tiger. For the pose, common methods such as random rotation and introduction of noise cannot change it too much and the latter even may change the pattern information which is not we want to see. For environment information, the random whitening method seems to be really helpful to change the images greatly while retaining the pattern information just as the Figure 7 shows. Second, after looking through all images in training set, we find that the number of different IDs is unbalanced and the number of identical id from different cameras is also unbalanced. In order to enlarge our data set and reduce the degree of unbalance, we apply the random crop to the data augmentation procedure after using random whitening. IDs which have relatively few images may have higher probability to crop while others have lower probability.


(a) (b)

(c) (d)

Figure 5: Example images of hard samples. (a) and (b) are the same ID, while (c) and (d) are the same ID.

3.6. Sampling Strategy

In practice, we have applied a sampling strategy to our train and validation stage to improve our model's accuracy especially in cross camera case.

The specific sampling strategy is using hard negative mining strategy. We use global distance to find the hard samples and make the majority of train set comprised of hard samples. Most of IDs which have hard samples are from cross camera, the examples are showed in Figure 5. The advantage of this method is making our model more inclined to learn the unique feature extracted from cross camera images. These features, in a way, seems to be the pattern information we encourage the model to learn. The implementation details about our sampling strategy will be discussed in 4.1.

3.7. Ensemble Strategy

In the test phase, each query or gallery image is extracted to get the global feature using the trained model. Based on these features, we calculate the Euclidean distance between each of the query images and each of the gallery images, so that we get a distance matrix ($M * N$, where M is the number of query images, and N is the number of gallery images), which represents the similarity between the query or gallery images.

As shown in Figure 6. We get the distance matrices under several different models and use these matrices to get the final distance matrix. The final distance is defined as

$$D_{i,j} = \frac{d_{i,j}^{M_1} + d_{i,j}^{M_2} + ... + d_{i,j}^{M_n}}{n}, \tag{4}$$

where n is the number of models. After sorting the final matrix, we can get the gallery images list for each query image.

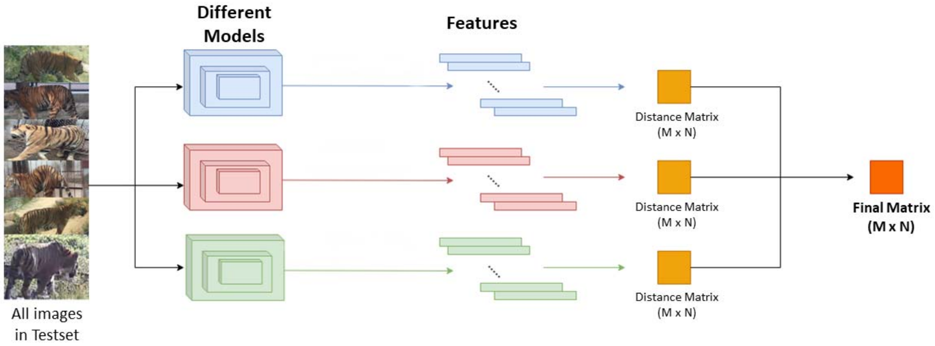


Figure 6: Framework of the model ensemble. Various models are trained by the different models, and each model generates a distance matrix. The final matrix is calculated from these matrices

4. Experiment

4.1. Dataset

In the ATRW dataset, 107 identities with 1877 images are for training. We choose 60 IDs to constitute the train set and choose the rest IDs to make the validation set where we split $1/3$ images as query images and rest as gallery images for each id. The experiment results shown in Tables 1, 2 and 3 are all based on this train set and validation set.

For details about sampling strategy, we first split the whole train set (60 IDs in total) into IDs with hard samples (30 IDs in total) and IDs without hard samples (easy samples, 30 IDs in total). In order to make the proportion of hard samples larger, we choose all 30 hard IDs and only 10 easy IDs to constitute our train set. However, this method makes the train data less which makes against the training process. In order to solve it, we split the 30 easy IDs into three 10-id groups. For each training, we select one group and all 30 hard IDs. After training, we can get 3 models trained with different training set and then we ensemble them and take the ensemble result as the last result for our sampling strategy.

Two obvious advantages of this method are: first, we ensure a high proportion of hard samples in train set; second, by using ensemble, we ensure that all images in train set have been considered.

4.2. Implementation Details

Our proposed model is implemented in Pytorch with CUDA. All experiments are performed on NVIDIA GTX2080Ti graphics processing units. We set the dimensionality of the final global representation to 2048 at first. We use Adam as the optimizer. The batch size is set to 30 and the total epochs is 20. We set the learning rate to 0.001 initially, which will exponential decay after half epochs.

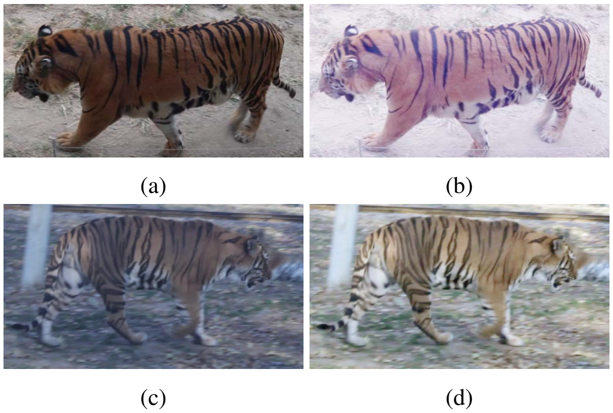


Figure 7: Example images of random whitening. original images: (a,c), processed images: (b,d).

In order to evaluate our model, we calculate mAP (Mean Average Precision) and top-k accuracies for each image. For a given query image, we calculate Euclidean distances between it and all gallery images. Then, we rank these distances in an ascending order. For top-k results, the retrieval will be considered as a success if the groundtruth gallery image is found in the top-k results. We take the maximum precision as the precision corresponding to this retrieval. So that we can obtain mAP by calculating the average value of all the retrievals' precisions.

4.3. Ablation Study

Through the ablation study in this section, we show the contributions of different tricks in our model. The following models are trained on the ATRW dataset.

Data augmentation. For validating the effectiveness of the

Table 1: The performance of all useful tricks on ResNet-50. All results are based on the dataset discussed in 4.1.

method	mAP	top-1	top-5
baseline	80.20	94.36	97.30
+flip as new id	84.13	96.42	98.48
+random whitening&crop	85.25	97.17	99.28
+sampling strategy	87.97	97.72	99.28
+local loss	**88.23**	**97.97**	**98.48**

Table 2: The performance of flip as new id method (flip-new) and flip as identical id method (flip). All results are based on the dataset discussed in 4.1.

method	mAP	top-1	top-5
baseline	80.20	94.36	97.30
baseline+flip-new	84.13	96.42	98.48
baseline+flip	78.65	94.04	97.30

flip as new id, we conduct three experiments in which the models are trained with different flip methods, *i.e.*, flip as new id, no flip (baseline), flip as identical id. Table 2 lists the mAP and top-k (k = 1,5) results of these methods. The flip as new id (flip-new) performs best as we expected while the flip as identical id (flip) is even worse than no flip which may be an evidence to show that the key for tiger Re-ID task is making the model capable of learning how to recognize stripe pattern of tigers rather than any background information.

For methods of random whitening and random crop, the results are shown in Table 1 and this trick also achieves an improvement on the basis of flip as new id method.

Sampling strategy. For validating the effectiveness of the sampling strategy, we employ the method discussed in 4.1. After ensemble of different models, we can realize our sampling strategy. The results shown in Table 1 are excellent on the basis of data augmentation methods.

Different losses. As shown in Table 1, we use global loss and local loss for the first 4 lines. Then we add the local loss (defined in 3.2) and achieve a little progress which may prove that local loss can make our model learn different perspectives of the images and it may be the key to improve the performance in the cross camera case.

4.4. Comparison with different models

In this subsection, we compare the results of different models based on the same train and validation set in Table 1. The results are shown in Table 3.

We proposed to use the models from ResNet series and DenseNet series. After submission, we also find that although models from ResNet series have good performance in single camera case, they performs badly in cross camera case for which we only use models from DenseNet series in our final ensemble and achieve better results in both single and cross camera case.

Table 3: Performance of different models. For all models, all usefull tricks have been considered. All results are based on the dataset discussed in 4.1.

Model	mAP	top-1	top-5
ResNet-50	88.23	97.97	98.48
ResNet-152	87.50	97.30	98.65
DenseNet-121	88.78	98.60	99.30
DenseNet-161	89.37	98.60	98.30
DenseNet-169	89.77	98.60	99.30
DenseNet-201	89.59	98.75	99.60
Ensemble with all	**90.64**	**99.60**	**100.00**

5. Conclusion

In this paper, we have introduced our work on the plain Re-ID track in CVWC2019 challenge. For the ATRW dataset, we chose some methods such as flip as new id, random whitening, dynamic programming, combining metric loss and classification loss and so on. All of these methods have made a certain degree of progress in the tiger re-ID task and finally we got the third place in the challenge of plain Re-ID phase.

However, the relatively bad performance on cross camera case and limitation of the dataset also makes the task still have a lot of work to do.

References

[1] Song Bai, Xiang Bai, and Qi Tian. Scalable person re-identification on supervised smoothed manifold. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2530–2539, 2017. 2

[2] De Cheng, Yihong Gong, Zhihui Li, Dingwen Zhang, Weiwei Shi, and Xingjun Zhang. Cross-scenario transfer metric learning for person re-identification. *Pattern Recognition Letters*, page S0167865518301430, 2018. 2

[3] De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *Computer Vision Pattern Recognition*, 2016. 2

[4] Su Chi, Jianing Li, Shiliang Zhang, Junliang Xing, Gao Wen, and Tian Qi. Pose-driven deep convolutional model for person re-identification. 2017. 2

[5] Yang Fu, Yunchao Wei, Yuqian Zhou, Honghui Shi, Gao Huang, Xinchao Wang, Zhiqiang Yao, and Thomas S. Huang. Horizontal pyramid matching for person re-identification. *ArXiv*, abs/1804.05275, 2018. 2

[6] Bing He, Jia Li, Yifan Zhao, and Yonghong Tian. Part-regularized near-duplicate vehicle re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3997–4005, 2019. 1

[7] Shuyuan Li, Jianguo Li, Weiyao Lin, and Hanlin Tang. Amur tiger re-identification in the wild. *arXiv preprint arXiv:1906.05586*, 2019. 2

[8] Wei Li, Xiatian Zhu, and Shaogang Gong. Person re-identification by deep joint learning of multi-loss classification. 2017. 2

[9] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z. Li. Person re-identification by local maximal occurrence representation and metric learning. 2014. 2

[10] Wu Lin, Wang Yang, Junbin Gao, and Li Xue. Where-and-when to look: Deep siamese attention networks for video-based person re-identification. *IEEE Transactions on Multimedia*, pages 1–1, 2018. 2

[11] Jiawei Liu, Zheng-Jun Zha, Di Chen, Richang Hong, and Meng Wang. Adaptive transfer network for cross-domain person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7202–7211, 2019. 1

[12] Yihang Lou, Yan Bai, Jun Liu, Shiqi Wang, and Lingyu Duan. Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3235–3243, 2019. 1

[13] Sakrapee Paisitkriangkrai, Chunhua Shen, and Anton Van Den Hengel. Learning to rank in person re-identification with metric ensembles. In *Computer Vision Pattern Recognition*, 2015. 2

[14] Zhao Rui, Wanli Ouyang, and Xiaogang Wang. Unsupervised salience learning for person re-identification. In *Computer Vision Pattern Recognition*, 2013. 2

[15] Zheng Tang, Milind Naphade, Ming-Yu Liu, Xiaodong Yang, Stan Birchfield, Shuo Wang, Ratnesh Kumar, David Anastasiu, and Jenq-Neng Hwang. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8797–8806, 2019. 1

[16] Rahul Rama Varior, Mrinal Haloi, and Wang Gang. Gated siamese convolutional neural network architecture for human re-identification. 2016. 2

[17] Cheng Wang, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. *Mancs: A Multi-task Attentional Network with Curriculum Sampling for Person Re-Identification*. 2018. 2

[18] Guangcong Wang, Jianhuang Lai, Zhenyu Xie, and Xiaohua Xie. Discovering underlying person structure pattern with relative local distance for person re-identification. *ArXiv*, abs/1901.10100, 2019. 2

[19] Guanshuo Wang, Yufeng Yuan, Chen Xiong, Jiwei Li, and Zhou Xi. Learning discriminative features with multiple granularities for person re-identification. 2018. 2

[20] Jiayun Wang, Sanping Zhou, Jinjun Wang, and Qiqi Hou. Deep ranking model by large adaptive margin learning for person re-identification. *Pattern Recognition*, 74:241–252, 2018. 2

[21] Peng Wang, Bingliang Jiao, Lu Yang, Yifei Yang, Shizhou Zhang, Wei Wei, and Yanning Zhang. Vehicle re-identification in aerial imagery: Dataset and approach. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2019. 1

[22] Li Wei, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. 2018. 2

[23] Feng Yang, Ke Yan, Shijian Lu, Huizhu Jia, Xiaodong Xie, and Wen Gao. Attention driven person re-identification. *Pattern Recognition*, 86:143–155, 2018. 2

[24] Xuan Zhang, Hao Luo, Xing Fan, Weilai Xiang, Yixiao Sun, Qiqi Xiao, Wei Jiang, Chi Zhang, and Jian Sun. Alignedreid: Surpassing human-level performance in person re-identification. *arXiv preprint arXiv:1711.08184*, 2017. 1, 2, 3

[25] W. S. Zheng, S. Gong, and T. Xiang. Towards open-world person re-identification by one-shot group-based verification. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 38(3):591–606, 2016. 2

[26] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2138–2147, 2019. 1

[27] Zhedong Zheng, Liang Zheng, and Yi Yang. A discriminatively learned cnn embedding for person re-identification. *Acm Transactions on Multimedia Computing Communications Applications*, 14(1), 2016. 2

[28] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1318–1327, 2017. 2