

# Amur Tiger Re-identification in the Wild

Shuyuan Li<sup>1</sup>, Jianguo Li<sup>2\*</sup>, Weiyao Lin<sup>1</sup>, Hanlin Tang<sup>3</sup>  
<sup>1</sup>SJTU, <sup>2</sup>Intel Labs, <sup>3</sup>Intel AIPG

## Abstract

*Monitoring the population and movements of endangered species is an important task to wildlife conversation. Traditional tagging methods do not scale to large populations, while applying computer vision methods to camera sensor data requires re-identification (re-ID) algorithms to obtain accurate counts and moving trajectory of wildlife. However, existing re-ID methods are largely targeted at persons and cars, which have limited pose variations and constrained capture environments. This paper tries to fill the gap by introducing a novel large-scale dataset, the Amur Tiger Re-identification in the Wild (ATRW) dataset. ATRW contains over 8,000 video clips from 92 Amur tigers, with bounding box, pose keypoint, and tiger identity annotations. In contrast to typical re-ID datasets, the tigers are captured in a diverse set of unconstrained poses and lighting conditions. We demonstrate with a set of baseline algorithms that ATRW is a challenging dataset for re-ID. Lastly, we propose a novel method for tiger re-identification, which introduces precise pose parts modeling in deep neural networks to handle large pose variation of tigers, and reaches notable performance improvement over existing re-ID methods. The dataset will be public available at <https://cvwc2019.github.io/>.*

## 1. Introduction

Wildlife conservation is critical for maintaining species biodiversity. Failure to protect endangered species on Earth may lead to imbalance ecosystems [3] and affect environmental health [36]. This mission is increasingly depended on accurate monitoring of the geospatial distribution and population health of these endangered species [27], especially in the face of poaching and loss of habitats. Traditional methods of attaching transmitters to wildlife are prone to sensor failure, difficult to scale to large populations, and cannot measure how the wildlife interacts with its environment.

Computer vision techniques are a promising approach to wildlife monitoring, especially with the use of unmanned

aerial vehicles or camera traps to collect visual data [16]. In particular, re-identification (re-ID) is a core vision method required to obtain accurate population counts and track wildlife trajectory [32]. Figure 1 illustrates one such system which tracks the movement trajectory of individual Amur tigers through an edge-to-cloud re-identification framework. Amur Tigers are classified as an endangered species, with a remaining population fewer than 600.

However, deployment of such systems is hampered by several challenges. *First*, resource constraints on the edge camera require low-power and accurate tiger detection to trigger the image capture [23, 29, 31] and thus avoid that massive irrelevant image capturing consumes space of storage card and battery life. *Second*, recent re-ID methods [14, 33] typically use pedestrians and cars as target objects, which usually have limited pose variations in a relatively constrained environment. In contrast, wildlife data have a wide range of pose variations due to unrestricted four-limbed movement, complex natural backgrounds, and unconstrained lighting conditions. *Third*, research into these open challenges is slow due to the lack of large-scale datasets and benchmarks beyond object types such as pedestrian and cars [7, 44, 46] that have the aforementioned weaknesses. Importantly, most existing datasets lack a systematic benchmark protocol to evaluate end-to-end re-ID performance [32]. This incurs the strong requirement to build a new dataset and benchmark to systematically study wildlife re-identification.

To address the above challenges, we introduce a novel large-scale dataset named Amur Tiger Re-identification in the Wild (ATRW). Identification of individual Amur Tigers are mostly based on the body stripe patterns [15], which can be easily deformed due to the posture and movement of the tiger. Manual corrections as in [15] are labor intensive and not suited for wild environments. Automatic correction requires identifying pose keypoints, and potentially additional advanced modeling methods to account for non-rigid body deformation due to tiger movement. Novel methods developed to address these issues can extend to the monitoring of other wildlife species or objects that rely on deformable body patterns for re-identification. In summary, our major contributions are:

\*Corresponding author.

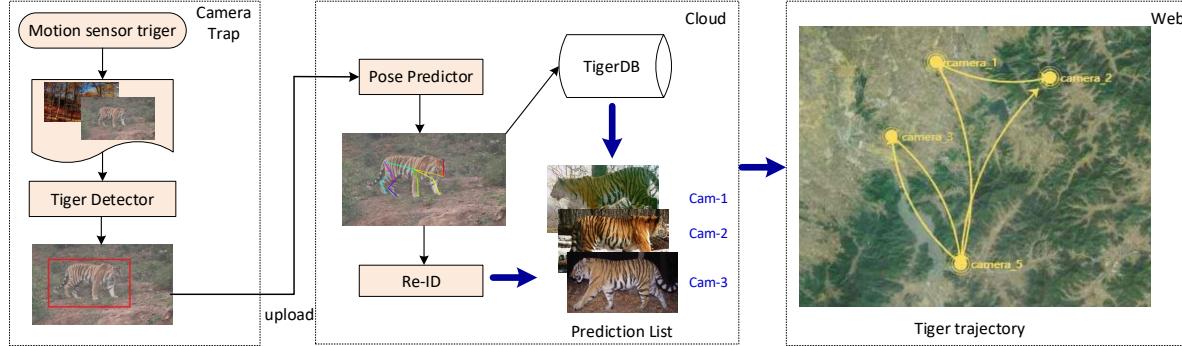


Figure 1: Framework of Amur tiger re-ID system. Motion sensor triggers lightweight detector on smart camera to further filter non-tiger images. Tiger related images are fetched or uploaded to cloud services, which will run pose estimation and re-ID algorithm to produce tiger to camera association. As cameras are discrete placed in a conservation park, we could visualize the tiger moving trajectory on the park map based on camera position information.

- We introduce a new large-scale wildlife dataset ATRW, which contains 92 Amur tiger identities from multiple wild zoos, 8,076 high-resolution video clips from multiple cameras in which tigers are annotated with bounding box positions, pose keypoints and identities on sampled frames.
- We create a systematic benchmark and comprehensive baseline on ATRW for the full tiger recognition pipeline, including: Amur tiger detection, pose estimation, and re-identification.
- We propose a novel solution to tiger re-identification based on precise pose parts modeling with deep neural networks to handle the large pose variation of tigers, which demonstrates noticeable performance improvement over traditional re-ID approaches.

## 2. Related Work

**Re-ID datasets.** Several large scale person re-ID datasets have been released in recent years [22, 30, 44] to support researches to improve algorithms performance and robustness. Vehicles are also an important object for re-ID due to its wide applications in video surveillance [7]. Most re-ID datasets only contain cropped images, with the exception of PRW [46], which provided raw frames along with annotated bounding box for evaluation of the full re-ID pipeline.

Besides the plethora of person or vehicle re-ID datasets, there are also a few datasets on animal re-ID, which are well summarized in the review [32], including primates [6, 11], tigers [20, 19], elephants [21], and whales [28]. However, these animal re-ID datasets have various weaknesses, such as small data sizes, limited annotations, and captured in non-wild settings. Because of these limitations, these datasets are not widely used in re-ID research. Our contributed dataset, ATRW, fills this gap to provide a large-scale, well-annotated, full pipeline re-ID dataset, to challenge existing approaches. For a comparison with existing

animal re-ID datasets, see Table 1.

The term of **Re-identification** was first proposed in 2005 [40] to support multi-camera person tracking with explicit “re-identification” based on appearance features. With large scale datasets, deep learning approaches become dominant in this field. Many approaches learn identity-related representations for re-ID purposes [12, 25]. Others formulate re-ID as a ranking problem, and feed a pair of images into a convolutional neural network to learn a ranking function [1, 22, 39]. Another type of approaches formulates re-ID as a metric learning problem, and combine CNNs with novel loss functions to learn similarity metrics [5, 33]. Local modeling with fusion of global and local representations is also used to enhance re-ID [37, 42, 43, 34].

Successful deployment of re-ID in the wild also requires object detection and pose estimation methods to normalize the image for accurate matching. **Efficient object detection** on the edge client is an active area of research. Typical methods include MobileNet-SSD [17, 31], SqueezeDet [38], and Tiny-DSOD [23]. **Pose estimation** is useful for precise target modeling especially in datasets with rich variations in pose. Human pose estimation is relatively well studied with datasets such as COCO [24] and MPII[2]. To our knowledge, there are no re-ID datasets that include annotated ground truth for pose estimation, and also no datasets that permit studying how pose estimation impacts the re-ID performance. Our ATRW dataset will provide ground truth annotations for tiger pose, as well as studies of the impact of generic pose estimation methods such as OpenPose [4], AlphaPose [9] and HRNet [35].

## 3. The ATRW Dataset

### 3.1. Annotation Description

The Amur tiger (also known as Siberian tiger, Northeast-China tiger) is a tiger population in the far east region (particularly the Russian Far East and Northeast China), which

Table 1: Comparison of animal re-ID datasets. ‘\*’ denotes number of video clips, and ‘-’ denotes no public data available. Our dataset is significantly larger, captured in the wild, with rich and dense bounding box annotations, as well as pose keypoint annotations.

Datasets	ATRW	[20, 19]	C-Zoo[11]	C-Tai[11]	TELP[21]	$\alpha$ -whale[28]
Target	Tiger	Tiger	Chimpanzees	Chimpanzees	Elephant	Whale
Wild	✓	✓	✗	✗	✗	✓
Pose annotation	✓	✗	✗	✗	✗	✗
#Images or #Clips	8,076*	-	2,109	5,078	2,078	924
#BBoxes	9,496	-	2,109	5,078	2,078	924
#BBoxes with ID	3,649	-	2,109	5,078	2,078	924
#identities	92	298	24	78	276	38
#BBoxes/ID	39.7	-	19.9	9.7	20.5	24.3

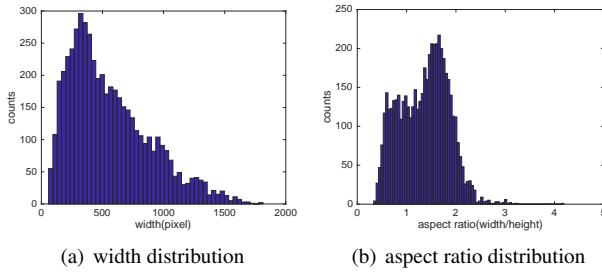


Figure 2: Distribution of width and aspect ratio (width/height) of bounding boxes in the dataset.

currently has less than 600 wild individuals in the world. Capturing enough image data for free-roaming Amur tigers is infeasible as these tigers have an activity range over hundreds of kilometers. Instead, with the help of the World Wildlife Foundation (WWF), a third-party company (MakerCollider) captures Amur tigers imagery data from multiple wild zoos in a unconstrained setting with time-synchronized surveillance cameras and tripod fixed SLR cameras. In total, 8,076 high resolution ( $1920 \times 1080$ ) video clips are captured with at least one Amur tiger inside. The raw data is donated to us for research purpose usages. We process the raw data by uniformly sampling video clips (one out of ten) into frames, while some frames are discarded due to motion artifacts, lack of tigers, or other noise.

The annotation contains three steps. *First*, we annotate a bounding box for each tiger as well as the view orientation of the tiger in the image (frontal, left, right, back). *Second*, professionals determine the tiger identity based on temporal and appearance cues. Tigers from different zoos are physically separated during annotation, so that the id annotation is fairly accurate. If the tiger cannot be clearly identified, an unknown id will be assigned to that tiger, which will not be used in the re-ID procedure. *Third*, as the tiger movement causes large pose variations, we further annotated skeleton keypoints for each tiger for downstream pose normalization or precise pose modeling. Details of the keypoint definition will be discussed in the following section.

Similar to many other re-ID datasets, we also provide a cropped dataset uncoupled from the wild environment for isolated tests of re-ID algorithms. The annotated images are

Table 2: Definition of key-points in our dataset

key-point	definition	key-point	definition
1	left ear	9	right knee
2	right ear	10	right back paw
3	nose	11	left hip
4	right shoulder	12	left knee
5	right front paw	13	left back paw
6	left shoulder	14	root of tail
7	left front paw	15	center, mid point of 3 & 14
8	right hip		

cropped according to the bounding boxes, and renamed with camera id, shot id, frame number and entity id. Here we use entity to refer to a combination of the tiger identity and its view information. Based on professional consultations [15], the tiger stripe pattern is the most informative marker of tiger identity. Since the left and right side of Amur Tigers have different stripe patterns, and it is rare to capture both sides of the tiger in the wild environment, we treat different sides of the same tiger as a different entity.

The full pipeline of tiger re-identification contains three modules as shown in Figure 1. We describe data information of each module in detail below.

## Detection Data

The object detection module allows the system to select only frames that include an Amur tiger, thus reducing storage, power, and networking consumption. Our tiger detection dataset includes 4,434 images with 9,496 bounding boxes. Some of the bounding boxes may have the ‘unknown’ tiger identity, as described above, but the annotations can still be used for training and testing object detector models. We provide annotations in the same format as that of PASCAL VOC [8]. See Figure 5 for some sample bounding boxes. The distribution of bounding box width and aspect ratio are shown in Figure 2.

## Pose Keypoint Data

This module performs pose estimation to locate tiger skeleton keypoints, which are important to the re-ID task due to the significant non-rigid movement of the tigers. The extracted pose information could be used to align and normalize the tiger, or provide precise modeling of the tiger to improve the accuracy of re-ID algorithms. We defined a set

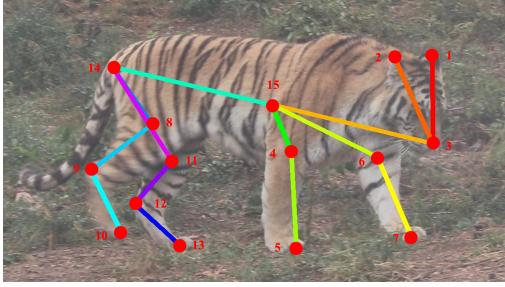


Figure 3: Definition of tiger keypoints.

Table 3: Individual distribution: single-camera vs cross-camera

	#Entity	#Tiger	#BBox
Single-Cam	132	53	1927
Cross-Cam	50	39	1722
Total	182	92	3649

Table 4: Variance of keypoints annotations.

keypoint	$\sigma^2(10^{-4})$	keypoint	$\sigma^2(10^{-4})$	keypoint	$\sigma^2(10^{-4})$
1	5.8	6	5.2	11	8.4
2	51.2	7	31.5	12	22.6
3	52.2	8	6.9	13	5.2
4	3.1	9	14.7	14	35.3
5	38.8	10	7.6	15	21.9

of tiger skeleton keypoints (Table 2), as illustrated in Figure 3 and Figure 6. If there are more than two annotations per tiger, the keypoint positions are averaged when the annotations are not too far apart. If there are significant differences, we manually validate the keypoints. The annotation is given in the COCO format.

## Re-ID Data

We define two settings to evaluate re-ID algorithms. First, in ‘plain re-ID’ setting, both the query tigers and database tigers are cropped and normalized with manually annotated bounding boxes and poses. Second, ‘wild re-ID’ requires automatic tiger detection and pose estimation to provide tiger normalization for the following re-ID procedure.

The re-ID dataset contains 92 tigers with 182 entities (different left/right sides as different entities), and a total of 3,649 bounding boxes. Most of the entities appear at least 10 times in the subset, as shown in the histogram of the number of occurrences in Figure 4. Unlike the popular person re-ID dataset Market-1501 [45], not all entities appear cross camera due to capturing restrictions in some wild zoos. In this case, we ensure that significantly different frames from the same camera are selected into our dataset. Table 3 lists the detailed data distribution among cross-camera and single-camera, showing that each entity has an average of 28.3 bboxes.

## 3.2. Evaluation Protocol

### The detection task

To conform with the following wild re-ID task and avoid possible ground truth bounding box annotation leakage, we

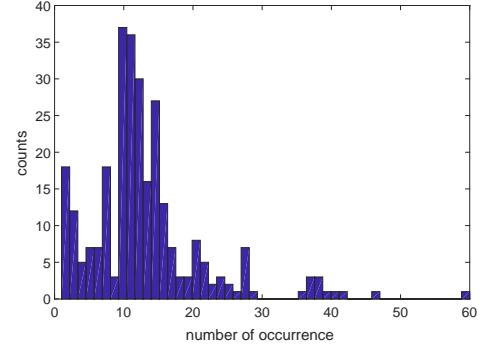


Figure 4: Distribution of occurrence times.

Table 5: Statistics of training set and test set for re-ID task

Datasets	#images	#entities	#tigers
Train	1887	107	75
Test	1762	75	58

Table 6: Statistics of query from cross-camera and single-camera cases in the re-ID task.

Queries	#images	#entities	#tigers
Single-Cam.	701	47	42
Cross-Cam.	1061	28	20

split 1,161 images (about 33%) as testing set for the detection task, which contains images from all tiger entities in the testing set of the wild re-ID task, while keep the other images as the training set, in which a portion of training subset (10%) are used for validation purposes. Similar to the COCO object detection challenge [24], we use the average precision (AP) at different Intersection-over-Union (IoU) thresholds to evaluate the performance. Since these algorithms need to be deployed on resource-constrained edge devices, we also use FLOPs and parameter count to compare detectors.

### The pose estimation task

The pose estimation task randomly splits the dataset into training, validation and testing subsets, containing 70%, 10% and 20% bboxes with keypoint annotations respectively. The evaluation metric is AP measured by Object Keypoint Similarity (OKS) as defined in COCO [24]. Table 4 lists the annotation statistics for each keypoint due to inhomogeneous annotations from different annotators, where  $\sigma_i^2 = E[d_i^2/s^2]$  denotes the variance of keypoint position normalized by object scale,  $d_i$  represents the deviation of the  $i$ -th keypoint among different annotators, and  $s$  represents the scale of the object. Then OKS is calculated as

$$OKS = \frac{\sum_i [\delta(v_i > 0) \exp(-\frac{d_i^2}{2s^2 k_i^2})]}{\sum_i [\delta(v_i > 0)]}, \quad (1)$$

where  $k_i = 2\sigma_i$  and  $v_i > 0$  if the  $i$ -th keypoint is visible.



Figure 5: Example images of detection subset.



Figure 6: Example images of pose subset.

## The re-ID task

As described previously, we have defined two tracks (plain re-ID and wild re-ID) based on whether images are cropped manually or automatically. Similar to Market-1501 [44], we choose the mean average precision (mAP) as the primary metric to evaluate the performance and top- $k$  accuracy as the secondary metric. For each query, the re-ID algorithm produces a list of predictions from which we measure the AP. The mean value of AP scores over all queries is the mAP, which is based on the evaluation code from Market-1501 [44], but with following difference. In Market-1501, if the algorithm returns a result image that is from the same cameras as the query image, a false positive is counted. In our dataset, we sometimes have tiger identities that are only captured from a single camera. Therefore, we modified the handling of returning images from the same camera slightly differently. We exclude temporal adjacent images within 1 second (forward and backward) to the query image from the query results for computing the AP score. To be more precise, we separate each query image into two categories: ‘single camera’, where the identity only appears in one camera, or ‘cross-camera’, where the target appears in multiple cameras. We report performance for both cases.

We then constructed the training and testing sets with the following procedure. First, we randomly choose 60% entities from the single-camera category and 40% entities from cross-camera category. Collectively, this formed the training set. The remaining images comprise the test set. Each image in testing set will be queried once with respect to the whole testing set. Metrics are calculated separately for single-camera and cross-camera part in testing set. Detailed statistics about dataset split can be found in [Table 5](#) and [Table 6](#).

As aforementioned, the re-ID task consists of both the ‘plain re-ID’ case and the ‘wild re-ID’ case. The plain case uses manual tiger bounding-box and pose annotations for the re-ID purpose, while the wild case aims to evaluate full-pipeline performance based on automatic tiger detec-

tion and pose estimation results. More strictly, bounding-boxes of both the gallery and query are generated by detection module. For those query tigers not found by the detectors, the corresponding query AP is counted as 0.

## 4. Baseline Methods

Here we describe the baseline methods we used for re-ID, especially including our innovative pose part-based CNN modeling framework. We use common baselines for object detection and pose estimation, which we will introduce with the results.

### 4.1. Classification based Baseline

Large-scale trained classification networks are believed to produce a rich representation that could be generalized to new tasks, particularly under the same dataset. Hence, we use a classification based re-id method as proposed in [12] as a baseline. That model uses an ImageNet pre-trained ResNet-50 [13] backbone, followed by two fully-connected (FC) layers with 1024 and 107 neurons respectively ([Figure 7](#)). The input image resolution is  $256 \times 128$  to accommodate the horizontal aspect ratio of tigers. The output feature length of the network ( $n = 107$ ) is set to the number of entities in the training set. To avoid overfitting, the ResNet-50 backbone parameters are frozen, and we only train the added classification layers. As classification tasks usually use a cross entropy (CE) loss, we refer this baseline as CE.

### 4.2. Triplet Loss Baseline

Metric learning is also widely used for re-ID, including methods such as Triplet loss[33], Quadruplet loss[5], or Tri-Hard loss [14]. Triplet loss aims to pull semantically similar points on the data manifold close in the embedding space and push dissimilar points farther apart. In this study, we choose TriHard, a triplet loss variant with batch hard mining, as one of our baseline method. TriHard loss examines the hardest pairs between positive pairs and negative pairs

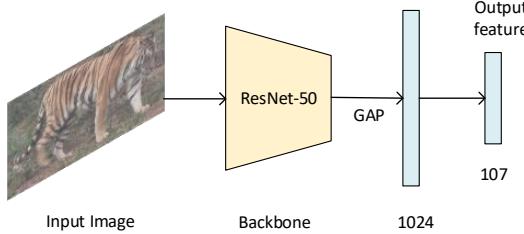


Figure 7: Architecture for classification/triplet based baseline.

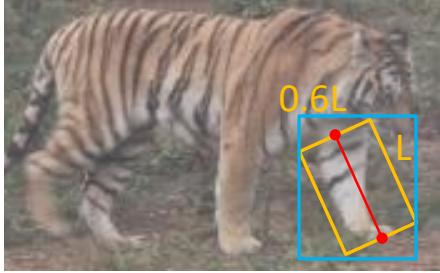


Figure 9: Extract AABB from skeleton. Two red points are keypoints defining a part, while the orange rectangle is the corresponding bounding box with height of  $L$  and width of  $0.6L$ . The blue rectangle is AABB which circumscribes the orange bbox.

in a mini-batch. Formally, the triplet loss is

$$\mathcal{L}_{tri} = [\|f(x_a) - f(x_p)\|_2 - \|f(x_a) - f(x_n)\|_2 + m]_+, \quad (2)$$

where  $x_a, x_p, x_n$  represents the anchor, positive and negative input respectively,  $f(\cdot)$  represents the network,  $m$  represents a margin. And the TriHard loss is defined as

$$\begin{aligned} \mathcal{L}_{TH}(X) = \sum_{i=1}^P \sum_{a=1}^K & \left[ m + \max_{p=1 \dots K} \|f(x_a^i) - f(x_p^i)\|_2 \right. \\ & \left. - \min_{\substack{j=1 \dots P \\ n=1 \dots K \\ j \neq i}} \|f(x_a^i) - f(x_n^j)\|_2 \right]_+, \end{aligned} \quad (3)$$

where  $X$  is a mini-batch of  $PK$  samples, which has  $P$  identities and  $K$  samples from each identity. The max-term represents the farthest positive pairs and the min-term represents the nearest negative pairs. For simplicity and fair comparison, we used the same network architecture as used in the classification baseline as shown in Figure 7.

### 4.3. Aligned re-ID Baseline

The above methods use the global representation, which may not be effective for targets with large pose variations. In the aligned re-ID method [42], the authors propose a local distance concept to enhance the feature representation. They use dynamic programming to compute the shortest mapping path between two images, and define local distance as the length of shortest path. The final distance between two images is the sum of global distance and this

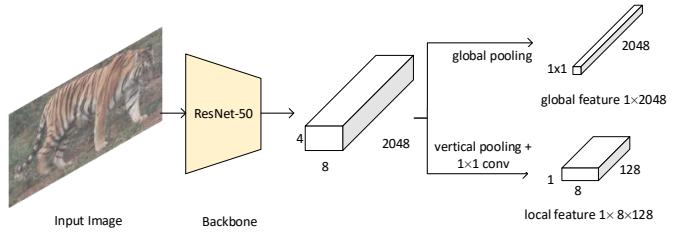


Figure 8: Architecture for Aligned-ReID baseline.

local distance. In our implementation, we only introduce the metric loss to combine local distance and global distance, without using other tricks in the original Aligned Re-ID work [42].

Due to the addition of local distance, the network architecture is changed as shown in Figure 8. For the last pooling layer, we pool along each column to obtain local features  $F(x) = \{f_1(x), \dots, f_W(x)\}$  for input image  $x$ , where  $W$  is the pooling feature map width ( $= 8$  as shown in Figure 8). We define some intermediate variables as

$$d_{i,j}(x, y) = \frac{e^{\|f_i(x) - f_j(y)\|_2} - 1}{e^{\|f_i(x) - f_j(y)\|_2} + 1} \quad i, j \in 1, 2, 3 \dots, W, \quad (4)$$

$$S_{i,j}(x, y) = \begin{cases} d_{i,j} & i = j = 1 \\ S_{i-1,j} + d_{i,j} & i \neq 1, j = 1 \\ S_{i,j-1} + d_{i,j} & i = 1, j \neq 1 \\ \min(S_{i-1,j}, S_{i,j-1}) + d_{i,j} & i, j \neq 1, \end{cases} \quad (5)$$

and the local distance as

$$D_{local}(x_1, x_2) = S_{W,W}(x_1, x_2). \quad (6)$$

A local distance based loss is defined as

$$\mathcal{L}_{local} = [D_{local}(x_a, x_p) - D_{local}(x_a, x_n) + m]_+, \quad (7)$$

where  $x_a, x_p, x_n$  are examined by batch hard mining in TriHard using global distance. The total loss  $\mathcal{L} = \mathcal{L}_{TH} + \mathcal{L}_{local}$  is used for this baseline.

### 4.4. Pose Part based Model

While the Aligned-ReID demonstrates strong performances for pedestrian re-ID, the method performs worse than the triplet-loss based baseline for tiger re-ID. Please see experimental section for detailed comparison. Because tigers have much larger pose variation due to non-rigid motion, the local feature representations created by pooling do not provide an invariant representation or precise modeling of the tiger body. Part-based models have shown great success for objects that are composed of deformable parts in tasks such as object detection [10] or fine-grained object recognition [41]. This kind of methods represent local parts by a rectangular patch, and adopts structured SVM to learn part structures. Recently, pose keypoint estimation techniques such as OpenPose [4] or AlphaPose [9] provide

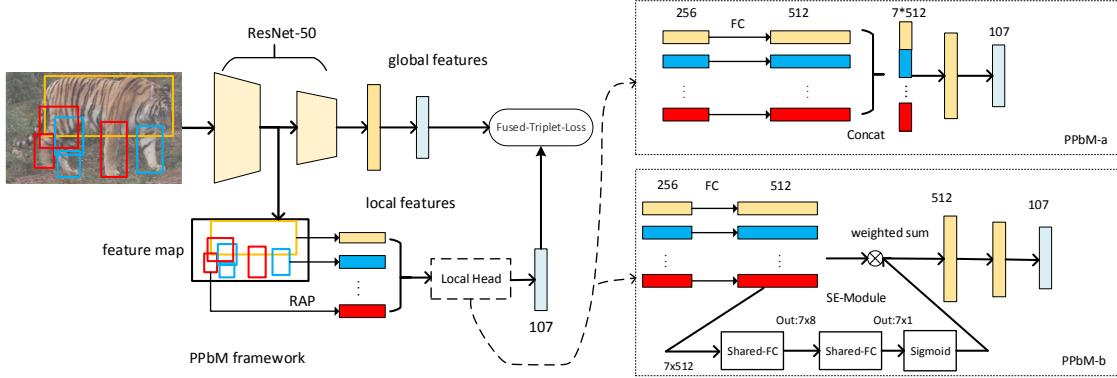


Figure 10: Architecture of our proposed pose part based model (PPbM). Left is the overall framework. Right are two “local-head” structures with PPbM-a at top-right, PPbM-b at bottom-right.

even precise body parts and skeletons modeling, which offers new opportunities for part-based model.

We propose pose part based model (PPbM) for tiger re-ID, which seamlessly integrates the result of pose keypoint estimation into deep neural networks. Figure 10 illustrates the network architecture. We characterize a tiger with a 7-part star model, including trunk, left and right of front legs, hind thighs, and hind shanks. For each part, we compute Axes Aligned Bounding Box (AABB) according to the pose skeleton, as shown in Figure 9. In fact, non-AABB region representation is more accurate, but less efficient to compute on feature-map during training phase, so that we have to resort to the AABB approximation. For each AABB area, we apply the ResNet-50 backbone, and extract the local feature representation with regional average pooling (RAP) on the *res3d* feature map. We use the intermediate layer *res3d* instead of the final residual layer because relative higher feature-map resolution ( $32 \times 16$  vs  $8 \times 4$ ) can provide more accurate RAP for each part. Nevertheless, most backbone network layers are shared between the global features and the local part-based features. Suppose  $\{\mathbf{x}_i\}_{i=1}^7$  are the RAP representations for the 7-parts, the local model is also trained with the TriHard loss defined as below

$$\mathcal{L}_{part} = \mathcal{L}_{TH}(\mathbf{g}\{F_{i=1:7}[\mathbf{f}_i(\mathbf{x}_i)]\}), \quad (8)$$

where  $\mathbf{f}_i(\cdot)$  is local transformation (i.e., FC-layer) for each part,  $F_{i=1:7}[\cdot]$  is a function to aggregate 7-parts information together,  $\mathbf{g}\{\cdot\}$  is global transformation (i.e., FC-layer), and  $\mathcal{L}_{TH}$  is TriHard loss defined in Equation 3. Note the global transformation  $\mathbf{g}\{\cdot\}$  outputs a 107-dimensional feature vector as final representation, similar to the global representation as in previous two baselines. There are two variants on aggregating local features from 7-parts: PPbM-a and PPbM-b. (1) PPbM-a adopts a concatenating function  $F_{i=1:7}[\cdot]$  to concatenate features from 7-parts together, as shown in top-right of Figure 10. (2) PPbM-b adopts a soft-

attention strategy to combine 7-parts together:

$$F_{i=1:7}[\mathbf{y}_i] = \sum_{i=1}^7 \alpha_i \mathbf{y}_i, \quad (9)$$

where  $\mathbf{y}_i$  is local transformation result for part- $i$ ,  $\alpha_i$  is soft-attention coefficient obtained similar as Squeeze-excitation network (SENet) [18] with shared FC layers as shown in bottom-right of Figure 10.

Both global representation and pose-part based representation could be trained either with cross-entropy loss or triplet loss. In our implementation, we defined a combined triplet loss to train the whole network together as

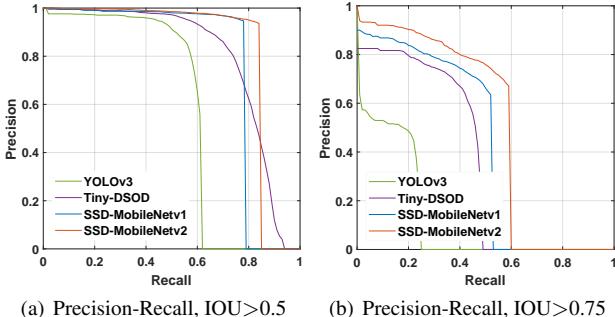
$$\mathcal{L} = \mathcal{L}_{TH} + \lambda \mathcal{L}_{part}, \quad (10)$$

where  $\lambda$  is a hyper-parameter to control contribution of global and part based representation, with default value  $\lambda = 1$ .

## 5. Experiments

### 5.1. Training Settings

For each of the three modules (object detection, pose estimation, and tiger re-ID), we choose the most widely used methods either with available open-source code or through re-implementation. Hyper-parameters for these methods are kept as default or recommended except when explicitly noted. We modified the number of training epochs according to dataset size. Specifically, we continued training until the accuracy on the validation set converged. We then re-trained the model on the entire training set, including the validation set, using the same number of epochs. Since the bounding boxes of tiger are usually horizontal major, which is different from the vertical major case like pedestrian re-ID, we exchange all the corresponding hyper parameters about width and height.



(a) Precision-Recall, IOU>0.5      (b) Precision-Recall, IOU>0.75

Figure 11: Precision-Recall curve under different IOU thresholds.

Table 7: Performance of detectors. mAP is averaged among IOU=0.50:0.05:0.95.

Detector	mAP	FLOPs	parameters
SSD-MobileNetv1	0.419	1.2B	6.8M
SSD-MobileNetv2	0.473	1.25B	14.8M
Tiny-DSOD	0.400	1.1B	0.95M
YOLOv3	0.237	18.7B	41.3M

## 5.2. Benchmark Results

### Tiger Detection

Since the object detector will be deployed on a battery-powered edge camera, we only tested lightweight object detectors to benchmark. In particular, MobileNetv1 [17] and MobileNetv2 [31] are widely used image classification models that also serve as backbones for object detector models such as Single Shot Detection (SSD) [26]. This paper benchmarks performance for both SSD-MobileNet-v1 and SSD-MobileNet-v2 models on the detection dataset. We used ImageNet pre-trained backbones for the MobileNet-SSD models. We also benchmarked other efficient object detectors Tiny-DSOD [23] and YOLOv3 [29], in which Tiny-DSOD is trained from scratch on the training set, while YOLOv3 pre-trains its backbone DarkNet on ImageNet and then strictly followed the official training process.

Table 7 lists the mAP results on the dataset as well as the FLOPs number and parameter size of these detectors. Note the input resolution of  $300 \times 300$  is the default resolution for the SSD model, and we expect that higher resolutions would provide better results. Based on these results, SSD-MobileNet-v2 achieves better accuracy but with higher computing cost than SSD-MobileNet-v1, while both SSD-MobileNet-v2 and SSD-MobileNet-v1 performs better than Tiny-DSOD and YOLOv3. The relatively lower performance of Tiny-DSOD may be due to the fact that we disable the FPN module in the original Tiny-DSOD network structure. The poor result of YOLOv3 may be due to prior boxes definition unsuitable for wild tigers, insufficient training or potential bugs, and warrants further investigation.

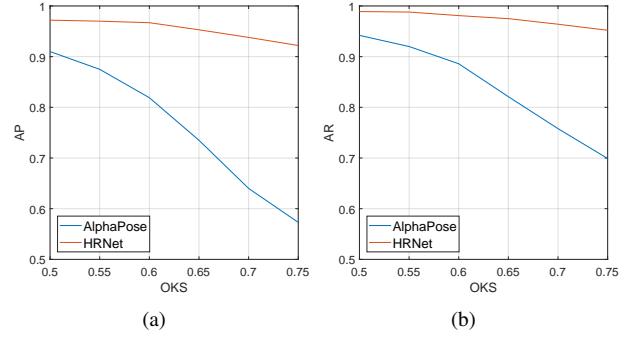


Figure 12: Curve of AP, AR w.r.t different OKS thresholds.

Table 8: Quantitative result of pose estimators, OKS=0.5.

Method	AP(%)	AR(%)
OpenPose	Fail	Fail
AlphaPose	0.574	0.671
HRNet	0.869	0.905

### Tiger Pose Estimation

For the pose estimation, we adopt OpenPose [4], AlphaPose [9] and HRNet [35] as they are open-sourced with state of the art results. Tigers have a very different skeleton definition from humans so we have to modify the original code for human pose accordingly to fit the skeleton definition of tigers. The modification of OpenPose code failed, yielding a non-convergent training procedure. Fortunately, we successively modified the code of AlphaPose and HRNet, and fine-tuned those pose estimators from provided checkpoint to accommodate the tiger pose configuration. Table 8 lists the quantitative results by AlphaPose and HRNet, and Figure 12 shows the Average-Precision (AP) and Average-Recall (AR) curve w.r.t different OKS thresholds. This benchmark reveals that state-of-the-art pose estimator by HRNet can provide fairly accurate (86.9% for tiger pose versus 77.0% for human pose [35]) pose results for the wild re-ID purpose. The results can be further improved with more training data.

### Plain and Wild Re-ID

We evaluated both wild case and plain case of tiger re-ID on baseline methods as described in section 4. In this study, we use ResNet-50 [13] pre-trained on ImageNet as the backbone network. Our experiment is based on the open-source implementation<sup>1</sup> with some modification of loss functions and data processing module. For the plain case, tigers are normalized to  $256 \times 128$  with manual annotated bounding-box, while for the wild case, tigers are normalized from automatic bounding-box and pose keypoints. Note that as the final detection result is not high enough, we adopt the trick proposed in [46], which utilizes the top- $K$  ( $K = 10$  at

<sup>1</sup><https://github.com/VisualComputingInstitute/triplet-re-ID>

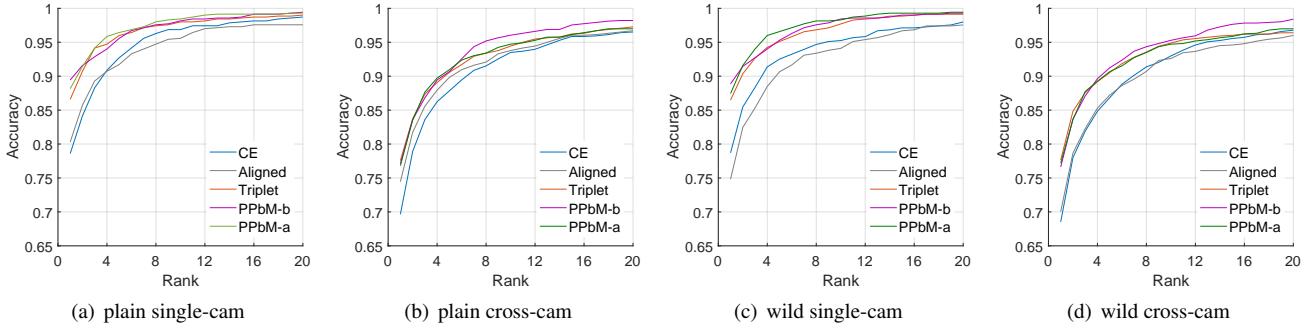


Figure 13: CMC curves for plain and wild tiger re-ID.

Table 9: Benchmark results of baseline re-ID methods on plain and wild re-ID tracks. PPbM fails in the wild case due to pose estimator failed in many images.

Setting	Method	Single-Cam			Cross-Cam		
		mAP	top-1	top-5	mAP	top-1	top-5
Plain	CE	59.1	78.6	92.7	38.1	69.7	87.8
	Triplet loss	71.3	86.6	96.0	47.2	77.6	90.6
	Aligned-reID	64.8	81.2	92.4	44.2	73.8	90.5
	PPbM-a (ours)	<b>74.1</b>	88.2	<b>96.4</b>	<b>51.7</b>	76.8	<b>91.0</b>
	PPbM-b (ours)	72.8	<b>89.4</b>	95.6	47.8	<b>77.1</b>	90.7
Wild	CE	58.8	78.7	92.5	34.5	68.5	86.8
	Triplet loss	70.7	86.5	95.1	45.2	<b>77.6</b>	90.5
	Aligned-reID	58.7	74.8	90.7	41.0	70.1	87.2
	PPbM-a (ours)	<b>71.0</b>	87.4	<b>96.6</b>	<b>50.3</b>	77.2	90.7
	PPbM-b (ours)	69.2	<b>88.9</b>	95.3	46.2	76.6	<b>91.2</b>

most) scored anchor boxes in the analysis.

Table 9 lists the mAP and top- $k$  ( $k = 1, 5$ ) results for all the compared baseline methods. For the wild case, we evaluate PPbM with bounding boxes provided by SSD-MobileNet-v2, and pose provided by the HRNet. Figure 13 further illustrates the recognition rate vs rank through the Cumulative Match Curve (CMC).

We have several observations from the results. *First*, each method clearly performs much better in the plain case than that in the wild case, especially on the cross-camera scenario. This indicates that there is still large improving space for detection and pose estimation modules. *Second*, PPbM model clearly outperforms other baseline models on both the plain case and the wild case. This indicates that precise pose modeling is important to target with large pose variations like tigers. *Third*, PPbM-a performs better than PPbM-b in terms of the mAP metric, but performs worse on the top-1 metric mostly. And more interestingly, PPbM-b outperforms PPbM-a on the CMC curve when  $rank > 5$  for the cross-camera case (Figure 13b). We believe that PPbM-b is still powerful and may have potential improving space. *Forth*, PPbM-a drops 1.4% in mAP from plain case to wild case, while the Triplet loss baseline drops more than 2.0%. This verifies the generalization power for PPbM to some extend. *Fifth*, there is large improved space for the cross-camera setting, which ensures great value of our ATRW dataset for re-ID research.

## 6. Conclusion

We present a new large-scale wildlife re-ID dataset named ATRW, which contains bounding box, pose keypoint and ID annotations of Amur tigers from multiple wild zoos. Compared to person or vehicle re-ID datasets, wildlife re-ID has a number of novel challenges for re-ID, such as varied pose, lighting, and background environments. In particular, large pose variations due to non-rigid motion require precise target modeling, which are less studied in current re-ID datasets and research. Through systematic benchmarking, we demonstrate that state-of-the-art algorithms are challenged by this dataset, compared to performance on pedestrian or vehicle datasets. The dataset also expands both the application area and the research challenges for computer vision techniques like re-ID into an important application domain of wildlife conservation.

## Acknowledgement

We would like to thank MakerCollider for capturing the raw data and donating it to us for academic usages, and thank WWF Amur tiger and leopard conservation programme team for great support during the whole project. Thanks Paul Lu, Kacie Zhang, Honggang Li, Yunpeng Song, Jiawei Li for numerous helps.

## References

- [1] E. Ahmed, M. Jones, and T. K. Marks. An improved deep learning architecture for person re-identification. In *CVPR*, 2015. [2](#)
- [2] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014. [2](#)
- [3] G. P. Asner, R. E. Martin, D. E. Knapp, et al. Airborne laser-guided imaging spectroscopy to map forest trait diversity and guide conservation. *Science*, 355(6323):385–389, 2017. [1](#)
- [4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. [2](#), [6](#), [8](#)

- [5] W. Chen, X. Chen, J. Zhang, and K. Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *CVPR*, 2017. 2, 5
- [6] D. Deb, S. Wiper, A. Russo, S. Gong, Y. Shi, C. Tymoszek, and A. Jain. Face recognition: Primates in the wild. *arXiv preprint arXiv:1804.08790*, 2018. 2
- [7] Z. Dong, Y. Wu, M. Pei, and Y. Jia. Vehicle type classification using a semisupervised convolutional neural network. 2015. 1, 2
- [8] M. Everingham, S. M. A. Eslami, L. V. Gool, et al. The pascal visual object classes challenge: A retrospective. *IJCV*, 111(1):98–136, 2015. 3
- [9] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017. 2, 6, 8
- [10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, et al. Object detection with discriminatively trained part-based models. *IEEE Trans on PAMI*, 32(9):1627–1645, 2010. 6
- [11] A. Freytag, E. Rodner, M. Simon, A. Loos, et al. Chimpanzee faces in the wild: Log-euclidean cnns for predicting identities and attributes of primates. In *German Conference on Pattern Recognition*, 2016. 2, 3
- [12] M. Geng, Y. Wang, T. Xiang, and Y. Tian. Deep transfer learning for person re-identification. *arXiv preprint arXiv:1611.05244*, 2016. 2, 5
- [13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5, 8
- [14] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017. 1, 5
- [15] L. Hiby, P. Lovell, N. Patil, et al. A tiger cannot change its stripes: using a three-dimensional model to match images of living tigers and tiger skins. *Biology letters*, 5(3):383–386, 2009. 1, 3
- [16] J. C. Hodgson, S. M. Baylis, R. Mott, A. Herrod, and R. H. Clarke. Precision wildlife monitoring using unmanned aerial vehicles. *Sci Rep*, 6:22574, Mar 2016. 1
- [17] A. G. Howard, M. Zhu, B. Chen, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 2, 8
- [18] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 7
- [19] K. U. Karanth, J. D. Nichols, N. S. Kumar, et al. Tigers and their prey: predicting carnivore densities from prey abundance. *PNAS*, 101(14):4854–4858, 2004. 2, 3
- [20] K. U. Karanth, J. D. Nichols, N. S. Kumar, and J. E. Hines. Assessing tiger population dynamics using photographic capture–recapture sampling. *Ecology*, 87(11):2925–2937, 2006. 2, 3
- [21] M. Körschens, B. Barz, and J. Denzler. Towards automatic identification of elephants in the wild. *arXiv preprint arXiv:1812.04418*, 2018. 2, 3
- [22] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014. 2
- [23] Y. Li, J. Li, W. Lin, and J. Li. Tiny-DSOD: Lightweight Object Detection for Resource-Restricted Usages. In *BMVC*, 2018. 1, 2, 8
- [24] T.-Y. Lin, M. Maire, S. Belongie, et al. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2, 4
- [25] Y. Lin, L. Zheng, Z. Zheng, Y. Wu, and Y. Yang. Improving person re-identification by attribute and identity learning. *arXiv preprint arXiv:1703.07220*, 2017. 2
- [26] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37, 2016. 8
- [27] J. Martin, W. M. Kitchens, and J. E. Hines. Importance of well-designed monitoring programs for the conservation of endangered species: case study of the snail kite. *Conserv. Biol.*, 21(2):472–481, Apr 2007. 1
- [28] A. Polzounov, I. Terpugova, D. Skiparis, and A. Mihai. Right whale recognition using convolutional neural networks. *arXiv preprint arXiv:1604.05605*, 2016. 2, 3
- [29] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 1, 8
- [30] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV workshop*, 2016. 2
- [31] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, pages 4510–4520, 2018. 1, 2, 8
- [32] S. Schneider, G. W. Taylor, S. Linquist, and S. C. Kremer. Past, present and future approaches using computer vision for animal re-identification from camera trap data. *Methods in Ecology and Evolution*, 2018. 1, 2
- [33] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 1, 2, 5
- [34] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian. Pose-driven deep convolutional model for person re-identification. In *ICCV*, 2017. 2
- [35] K. Sun, B. Xiao, D. Liu, and J. Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 2, 8
- [36] D. Tilman and D. John. Biodiversity and stability in grasslands. *Nature*, 1994. 1
- [37] L. Wei, S. Zhang, H. Yao, W. Gao, and Q. Tian. Glad: Global-local-alignment descriptor for pedestrian retrieval. In *ACM Multimedia*, pages 420–428, 2017. 2
- [38] B. Wu, F. Iandola, et al. Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. In *CVPR Workshops*, 2017. 2
- [39] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Deep metric learning for person re-identification. In *ICPR*, 2014. 2
- [40] W. Zajdel, Z. Zivkovic, and B. Krose. Keeping track of humans: Have i seen this person before? In *ICRA*, 2005. 2
- [41] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*, pages 834–849, 2014. 6
- [42] X. Zhang, H. Luo, X. Fan, et al. Alignedreid: Surpassing human-level performance in person re-identification. *arXiv preprint arXiv:1711.08184*, 2017. 2, 6
- [43] L. Zheng, Y. Huang, H. Lu, and Y. Yang. Pose invariant embedding for deep person re-identification. *arXiv preprint arXiv:1701.07732*, 2017. 2

- [44] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. [1](#), [2](#), [5](#)
- [45] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. [4](#)
- [46] L. Zheng, H. Zhang, S. Sun, M. Chandraker, and Q. Tian. Person re-identification in the wild. In *CVPR*, 2017. [1](#), [2](#), [8](#)