Re-identification of vessels with convolutional neural networks

Kong, Y.

Award date:
2018

Re-identification of Vessels with Convolutional Neural Networks

Yitian Kong*

***Abstract*— Vessel re-identification is an important task in maritime surveillance. Similar to pedestrian re-identification problems, vessel re-identification also has challenges due to illumination, occlusion, viewpoints and complicated background. To explore the vessel re-identification, in this work, first we classify the detected vessels into 10 vessel types and 5 orientation classes. Then, we propose a new vessel re-identification approach based on the original triplet method. To support our research, we also present three datasets for multi-class vessel detection, vessel orientation recognition, and vessel re-identification. Moreover, we explore several conditions which can influence the proposed re-identification model performance. Our experimental results reveal that our proposed approach achieves 81.46% of the mean average precision accuracy in $3.8ms$ for a single image to query the correct match in the database.**

***Keywords*— vessel re-identification, vessel detection, vessel orientation recognition, Convolutional Neural Networks**

I. INTRODUCTION

In maritime surveillance, it is essential to keep track of vessels to monitor safety, unreported fishing, drugs smuggling, etc [28]. There are many works which have been developed based on satellite images to tackle these kinds of problems. However, camera-based surveillance is also a vital part, as it can be deployed on the shoreline to monitor from different points of views. One of the important components in maritime surveillance is vessel re-identification, which should discover whether a vessel is captured on another location or time by different cameras. In other words, the vessel re-identification model should automatically find the query ship in different cameras, as presented in Fig. 1. Vessel re-identification task usually consists of vessel detection and vessel retrieval. The camera images contain not only vessels but also irrelevant objects, as shown in Fig. 1. Therefore, we first need to detect the vessel bounding box. Then we can use the detected bounding box in re-identification. Furthermore, it is also crucial to classify the vessel types. Unlike the pedestrian re-identification where all humans have the similar appearance, vessels have different classes, such as passenger ship, river cargo ship, etc. Specifically, for vessel re-identification task, we first detect the ship bounding box and ship type, and then we re-identify the ships by searching for the best match among the database samples. Besides, detecting vessel orientation is also a critical approach in the process of vessel re-identification, since orientation of a vessel provides the auxiliary information about the vessel.

Research Question Explore a vessel re-identification approach and training settings which would enable real-time vessel re-identification with at least 80% of the mean average precision score.

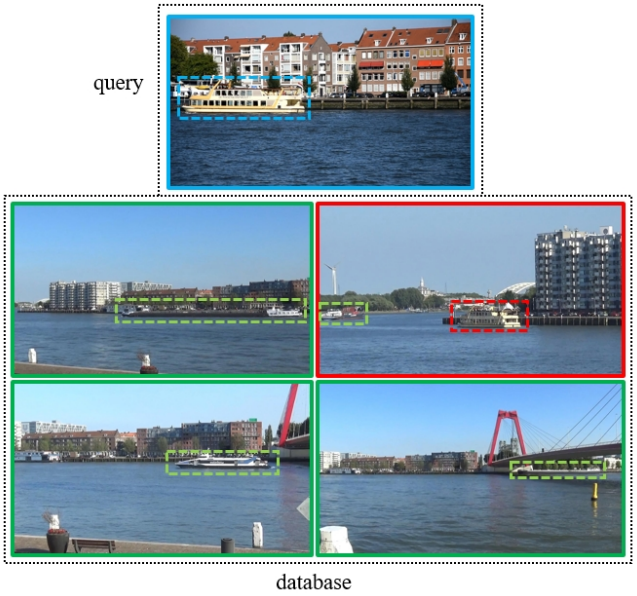


Fig. 1: The vessel re-identification is to find the image with same vessel in the database given a query vessel image. The blue box means the query image, while green box represents other vessels in the database and the red box is the correct matched one in the database.

Nevertheless, to the best of our knowledge, there is no existing work on vessel re-identification nor vessel re-identification dataset. Here, we first propose a multi-class vessel detection and orientation recognition dataset, which is collected from several cities in the Netherlands and Turkey including 11,000 images with over 30,000 vessels. All images are labeled by bounding boxes for all vessels, as well as categories and orientations for each vessel. The second dataset we propose is captured in several cities in the Netherlands by two cameras. It contains over 4,600 images of 733 vessel identities. All unique vessels are labeled by the bounding boxes, types and orientations.

Despite there is no vessel re-identification research, pedestrian re-identification has been developed dramatically in the recent years. Especially deep-learning based pedestrian re-identification has achieved significant improvement. Therefore, we refer to pedestrian re-identification, since it is similar to vessel re-identification from several aspects. First, human surveillance in the city is based on visual cameras, which is the same data type as we use. Second, the goal of human re-identification is to re-identify the same person in the database

*Yitian Kong is with Video Coding and Architectures Research Group, Department of Electrical Engineering, Eindhoven University of Technology, 5612 AZ Eindhoven, the Netherlands. Email: y.kong@student.tue.nl

Fig. 2: Slight unique differences of different vessel identities but from same vessel model.

given a query image, which is the same task as we pursue for vessels. Third, the challenges in pedestrian re-identification are similar to our application. These challenges for vessel re-identification are, a) the cameras are placed on the shorelines in different locations, which leads to different viewpoints of the same vessel, b) vessels are affected by occlusion, illumination change, and other environmental noises, c) different vessel identities have similar appearance because they belong to the same ship companies/models, which makes the vessel re-identification even more challenging. Fig. 2 illustrates few samples of this challenge.

Inspired by the triplet model of pedestrian re-identification [10], we propose a vessel re-identification approach to alleviate the challenges of vessel re-identification. The goal of this approach is to learn how to extract more discriminative feature representations. With this, the same vessel identities will be similar to each other, while being different from other vessel identities. In general, our contribution of the work is as following:

- We propose a multi-class vessel detection and orientation recognition dataset which is collected in multiple cities in the Netherlands and Turkey. Furthermore, we propose a vessel re-identification dataset which is captured in several places in the Netherlands from two non-overlapping viewpoints.
- We implement a multi-class vessel detection and orientation recognition approach which can detect vessel bounding box and classify the vessel categories and orientations.
- We propose a vessel re-identification approach which constrains the original triplet loss [10] more. Furthermore, we explore several conditions which can influence the re-identification performance.

This work is structured as follows. Related works are surveyed in section II. In section III, we present the multi-class vessel detection and orientation recognition method. Section IV presents our proposed vessel re-identification approach. Section V evaluates the proposed methods.

II. Related work

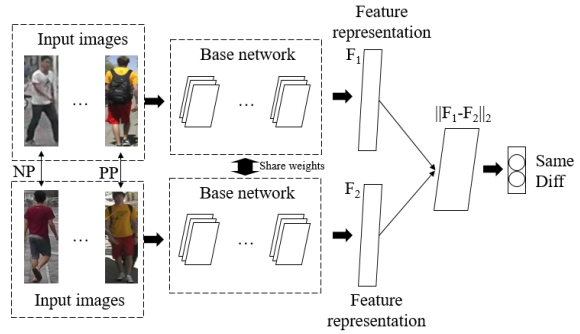
In this section, first, we survey the state-of-the-art works of vessel detection and classification. Then we review the re-identification methods.

1) Multi-class vessel detection and vessel orientation recognition: With recent developments in Convolutional Neural Networks (CNN), most of the state-of-the-art works in object detection use CNNs. Generally, these works focus on either proposing a new CNN or improving an existing CNN such as Faster RCNN [18], or SSD [16]. Some of these works [2,13,28,29,34] focus on detecting vessels in the synthetic aperture radar (SAR) images. In [12], the proposed method improves the Faster RCNN by combining the traditional constant false alarm rate method to select better region proposals generated by Faster RCNN to improve the accuracy of the predicted vessel locations. The work in [34] propose SVDNet which jointly utilizes the CNN and the singular value decompensation algorithm to learn more discriminative features from the SAR images with the interference of clouds and different sizes of vessels. Moreover, [29] propose a new model called S-CNN which embeds an improved saliency detection method improving accuracy, especially for the offshore small sized vessels.

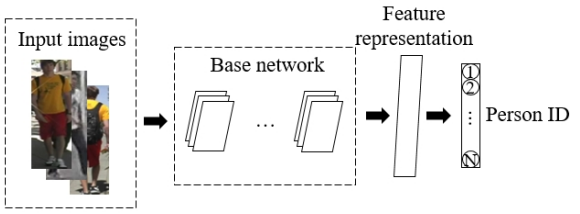
We did not find any works of vessel orientation detection, but there are some vehicle orientation estimation approaches. Similar to the methods of vessel detection, vehicle orientation recognition is also based on object detection algorithms. [8] proposed a CNN architecture based on Faster RCNN which can detect the location of vehicles and estimate the orientation simultaneously. The main idea is that it adds another classification layer with softmax loss function to predict the orientation in the ground plane.

2) Pedestrian re-identification: To the best of our knowledge, there is no work on vessel re-identification. However, pedestrian re-identification are widely explored in the literature. These methods attempt to re-identify the same person over different locations by matching a query image to the previously captured database images. Existing pedestrian re-identification works approach the problem by improving the feature representation to better discriminate images and calculating a distance metric which can find the similarity between two feature embeddings. Most of the recent works [1,4,5,7,8,12,16,18,20,24-28,32] focus on obtaining more distinguishable feature representations. Generally, they are based on either verification model or identification model [33].

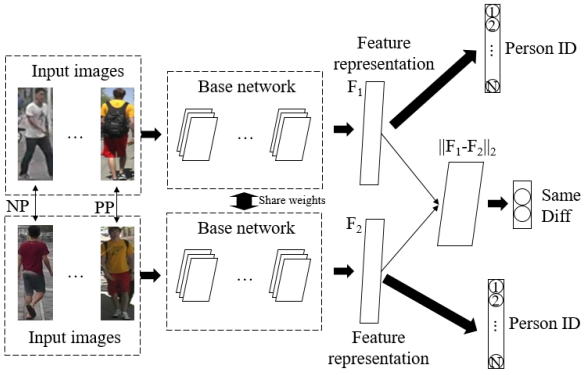
The verification models usually adopt siamese network as base architecture which takes pairwise images as input and has two same branches. The general architecture of the verification model is presented in Fig. 3a. The goal of the verification model is to output a similarity score between two input images by utilizing the Euclidean distance of extracted features to decide whether the input images belong to the same identity. The work in [23] improve the siamese network by integrating matching gates. These gates investigate the low-level features to distinguish the critical point in higher layers. Instead of low-level features, [1] improved the siamese network by appending a patch-matching layer which is used to find the similarity of mid-level features of two input images. Furthermore, [5] proposed the triplet architecture on pedestrian re-identification. The triplet method takes three images as input while two input images have the same identity and the third input image belongs to another identity. And the triplet loss pulls the features of same identities close while pushing the features



(a) The verification model architecture.



(b) The identification model architecture.



(c) The combination model architecture.

Fig. 3: Architectures of three basic pedestrian re-id models. The numbers in the circle represent different identities. The N in the circle is the total number of person identities in the dataset. NP and PP represent negative pairs and positive pairs.

of different identities away. According to [3], this condition is not restricted enough such that it may result in a situation where the images belonging to the same identity could be clustered in a large distance. Therefore, [3] adds another condition that the distance between similar pairs should be smaller than a predefined margin.

The generic architecture of the identification model is presented in Fig. 3b. The input is a single image while the output of the network is the predicted person identity. In [27], the proposed method utilizes a typical classification model which generates the identity of each input person. This method combines several pedestrian re-identification datasets as a whole large dataset since the conventional datasets contain a low amount of samples (e.g., the VIPeR [7] which only contains 1264 images of 632 people). A robust model can be obtained from the mixed datasets using the convolutional neural network. [26] improves the identification model by combining the hand-crafted features with CNN features to fine-tune the network. Moreover, the attribute is also utilized as auxiliary information for pedestrian re-identification. The work in [17] considers the enormous data disparity between ImageNet [4] and pedestrian re-identification datasets since such datasets are usually captured by surveillance cameras which have relatively lower quality. This work utilizes the pedestrian attribute dataset as the auxiliary dataset to fine-tune the pretrained network. Furthermore, [15] improves the identification model by utilizing attribute labels and person identities as final classification labels. With the single input image, the network can not only recognize the attributes appeared in the input person but also re-identify the pedestrian identity.

A typical combination model is presented in Fig. 3c. It is similar to the verification model that it also has two branches. But there are two extra fully connected layers which are used to predict the person identities. The identification loss and verification loss work together to optimize the base network. The work in [6] has the similar architecture as the presented. This work adopts a siamese network as base architecture and adds two identification subnets and a verification subnet. Besides, it introduces a dropout unit to drop the same neurons for verification subnet regarding two feature vectors generated by the siamese network. The work in [33] improves the siamese architecture by adding a square layer which calculates the squared difference between two feature representations generated by the base network. Moreover, the work in [2] combines the triplet model and verification model to improve the performance further. It first uses two convolution layers to transform the input three images to feature vectors. Then, these three feature vectors are fed into two subnets. The idea of the triplet subnet is same as the triplet loss function used in the work [5]. The verification subnet contains three convolutional layers to compare the difference between the positive pair (two images belong to the same person) and between the negative pair (two images belong to different persons).

In this work, we present a multi-class vessel detection and orientation recognition dataset, and a vessel re-identification dataset. We refer to these datasets as VesselDetection and VesselReID. The VesselDetetion contains 31,078 vessels captured in multiple cities in the Netherlands and Turkey under different weather conditions. Each vessel is annotated by bounding box, vessel type, and vessel orientation. We use this dataset to train the multi-class vessel detection and the vessel orientation recognition models. The VesselReID includes 4,616 images which are captured by two cameras in several places in the Netherlands with different backgrounds like natural scene and buildings. Each vessel is labeled by a unique id and appears in more than two images. Moreover, we also annotate the bounding box, vessel type and vessel orientation of each vessel for the potential further process. Additionally, we propose a new vessel re-identification model which uses triplet model as base architecture. This

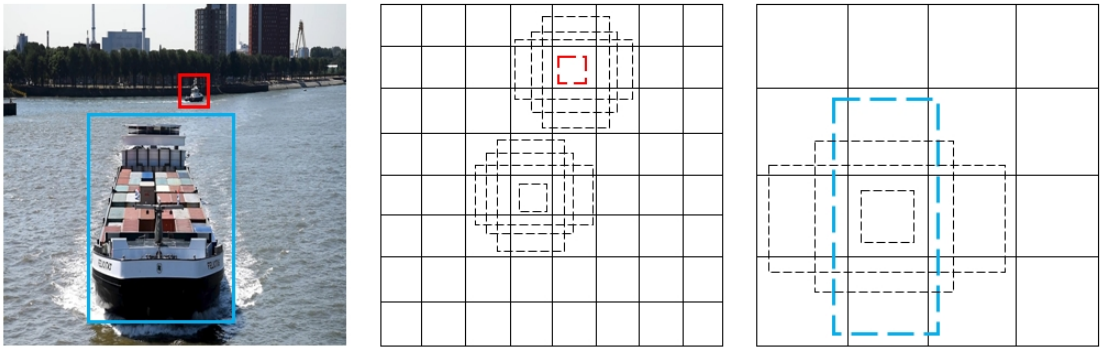


Fig. 4: The SSD feature maps in multi-scales.

Fig. 5: Five orientations of vessels. First row is front, back and side while the second row is front-side and back-side.

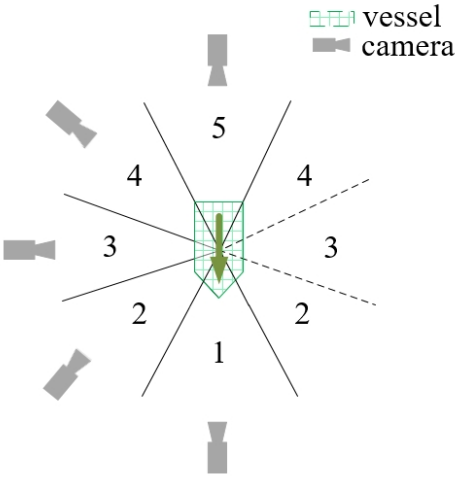


Fig. 6: Vessel orientations. The numbers 1 to 5 represent five orientations as front, front side, side, back side, back.

model learns both the distance metric (to pull the positive pairs close and push negative pairs away in the feature space) and a new hyperplane decision making metric. The second metric improves the feature similarity of the samples belonging to the same vessel identity.

III. The Vessel Detection and Classification Method

A. Multi-class vessel detection

We use single shot detection (SSD) [16] as our base method to detect the location of vessels in an image and predict the vessel type. SSD takes single image labeled by ground-truth bounding box and object class as input, and generates the predicted bounding box and predicted object class. It appends a set of convolutional feature layers by a base network which can be an image classification network like VGG [21]. These feature layers can generate multiple feature maps in different scales. As presented in Fig. 4, the leftmost image is the raw image labeled by ground-truth bounding boxes. The middle one is the 8×8 feature map while the rightmost is the 4×4 feature map. The network will evaluate four default boxes with different aspect ratio represented by dash lines in Fig. 4 for each feature map cell. For each default box, the network will predict the shape offset with the ground-truth boxes as well as the vessel types. The network concatenates six different scales of features to provide the final detection.

B. Vessel orientation recognition

Since the visual cameras are located in different places, they provide different views to the vessels. As illustrated in Fig. 5, we divide the vessel orientation into five classes, which are front, back, side, front side, and back side. In Fig. 6, the given arrow represents the vessel positioning direction. Since the views from the left side and right side are equivalent, we represent the viewpoints just by dividing the left area into five equal parts. The viewpoints $1, 2, 3, 4, 5$ represent the labels, *front, front side, side, back side, back*, respectively. Similar to the multi-class vessel detection, we adopt the VGG-SSD [16] network to detect the location of vessels and predict the five different orientation classes at the same time.

IV. The Proposed Vessel Re-ID Method

The architecture of the proposed method is illustrated in Fig. 7. It has three parts which are presented in different colors. The black part includes the three base networks which share weights between each other. The base network is used to transform the input image into a feature vector. To reduce the risk of error in computing feature distance in the triplet subnet, we use the same CNN for the three branches. In this work, we use ResNet50 [9] as our base network, and we take the average pooling layer as the base

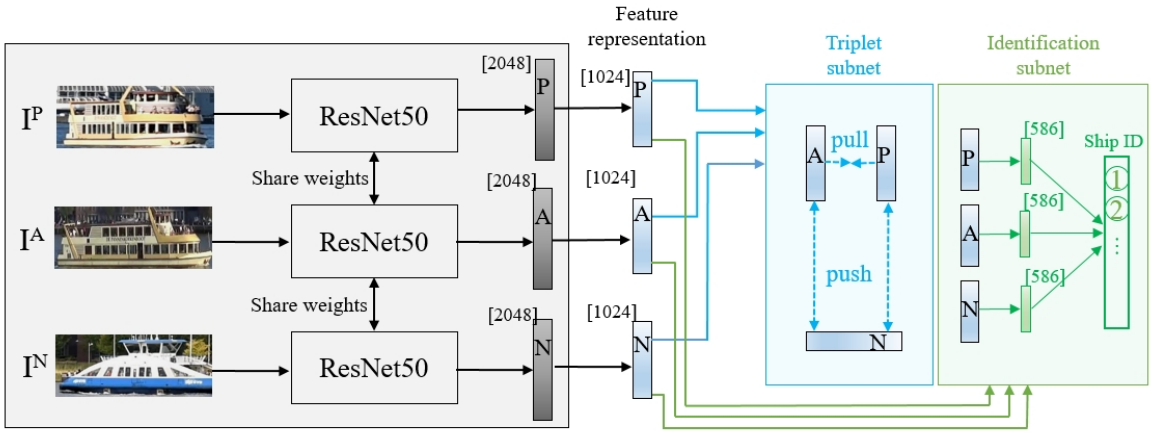


Fig. 7: The proposed vessel re-id model architecture. The letter A and P represent same vessel while N is another different vessel.

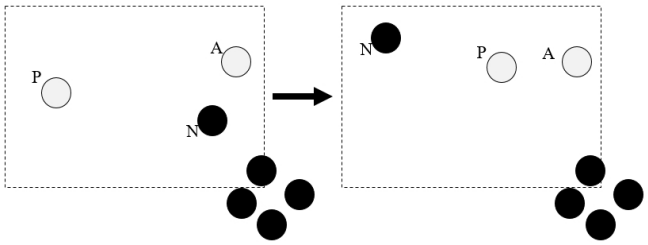


Fig. 8: The triplet loss will pull samples from same identities close and push different identities away. But it may also mislead negative to the wrong direction against with its cluster. The letter A, P, and N represent three images anchor, positive and negative, while A and P are the same identity and N is from another class.

network output which is a 2048-dims feature vector. Then we append a batch normalization layer which could speed up the convergence and be beneficial for deep convolution networks [11] and obtain a 1048-dims feature representation. As presented in Fig. 7, the blue box is the triplet subnet while the green box is the identification subnet. The triplet subnet pulls the features of same vessel identities close and pushes the features of different vessel identities away. The identification subnet improves the similarity of features for the same vessel identities. The base network receives three images as input denoted by I^A, I^P, and I^N, while I^A, I^P belong to the same vessel (positive pair) and I^N is another vessel (negative sample). The objective of our method is to cluster the features of same vessel identity to a single point and at the same time to increase the feature similarity between the same vessel samples.

A. Triplet subnet

We first briefly review the traditional triplet model and its limitations. FaceNet [19] first proposed the triplet loss and applied it on face re-identification. The goal of triplet loss is to pull image features from the same class closer to each other than other samples from different classes, which is illustrated in Fig. 8. Suppose we have three images as A (Anchor), P (Positive) and N (Negative) while A and P belong to the same identity and N is another identity. This process can be expressed by

$$D_{AP} - D_{AN} \geq \alpha \tag{1}$$

where D_{AP} is the distance between features of images from the same class and D_{AN} is the distance between features of images from different classes in the feature space. α is a margin to determine how far should two distances be.

By optimizing this process over the whole dataset iteratively, positive pairs converge into a single cluster while getting more distance with negative samples. As a limitation, the triplet architecture only considers two different identities at a time. This can push the negative pair against its cluster [30], as presented in Fig. 8. This drawback increases the convergence time. Another limitation of triplet architecture appears when we only choose *easy negative* samples (the negative pair with a very different appearance compared to the anchor). The explicit difference between the anchor image and its easy negative pair disables the network to learn how to perform differentiation between a positive and a negative sample with similar appearance. Therefore, it is crucial to select *hard negatives* to improve the performance. An instance of hard negative can be a person with similar clothes with anchor person. Accordingly, it is also important to use *hard positive* pairs (e.g., the same person with different appearance due to pose or viewpoint). On the other hand, if we choose the "hardest negative" or "hardest positive", the network can only learn some outliers of the dataset.

Evidently, the proper sample-set selection is of vital importance for triplet learning. The work in [10] alleviates the hard mining problem. It first picks P person and K images per person in a batch. After feeding the $P \times K$ images into the CNN, it obtains feature representations of these images. Then, it calculates all pairwise Euclidean distances of all features. For each image, the positive pair is selected as the

image from the same identity but with the largest distance from the anchor. The negative pair is picked as the image from a different identity which has the smallest distance compared to the anchor. In this case, the triplet pairs consist of the hardest negative and the hardest positive in this mini-batch. In order to increase the convergence speed, the works in [19] and [20] consider the *moderate negative* and the *moderate positive*. According to these papers, this technique also improves the re-identification accuracy.

B. Identification subnet

As discussed above, one of the triplet limitations is the negative misleading problem, which is shown in Fig. 8. In order to solve this problem, we propose the multi-task learning architecture, as illustrated in Fig. 7. In the identification subnet, we consider all the samples belonging to the same identity as a unique label and perform as multi-class detection learning. With the output of the base network, we feed the feature representations into a new fully-connected layer to generate 586-dims feature vectors, as there are 586 different identities in our VesselReID training set. Then, we use the softmax function to normalize the feature vectors. By adding this subnet, the final loss function of our network can be formulated as follows:

$$L = \lambda L^{triplet} + (1-\lambda) L^{identification} \tag{2}$$

where $L^{triplet} = \alpha + D_{AP} - D_{AN}$ according to equation (1) and $L^{identification}$ is the softmax loss function. The trade-off parameter $\lambda \in (0, 1)$. And when $\lambda = 0$ the final loss becomes identification loss function. When $\lambda = 1$ it turns to pure triplet loss function.

This proposed loss function restricts the CNN such that the feature representations of the same vessel should be similar to each other while being different from other vessels.

V. Empirical validation

A. Datasets

*1) **VesselDetection dataset**:* This dataset is captured in several places in the Netherlands and Turkey. It contains 11,000 images with 31,078 vessels. Each image is annotated by three labels: bounding box, vessel type, and vessel orientation. In this dataset, we annotate ten vessel types based on the captured data in all maritime backgrounds like river, harbor, sea. As illustrated in Fig. 9, the ten vessel types are sailing ship, container ship, passenger ship, fishing ship, tanker ship, river cargo ship, boat, yacht, tug ship, and taxi ship. Additionally, the orientation labels include front, front side, side, back, and back side. We divide the dataset into training set and testing set with 10,000 images (28,260 labels) and 1,000 images (2,818 labels), respectively.

*2) **VesselReID dataset**:* This dataset is captured by two different cameras in multiple cities in the Netherlands. It contains 4616 images with 733 different vessels. Each vessel is represented by more than two images. Also, we annotated three states of vessels which are normal, truncated and occlusion. These status are explained in Fig. 10. For example, the up-left image of the first vessel in Fig. 10 is occluded by other vessels. The first image of the second ship loses part of the body which is the truncated. Similar to our VesselDetection, this dataset is also labeled by bounding box, vessel model and orientation. Besides, we give a unique id to each vessel. To fit in our vessel re-identification method, we crop each vessel from the whole image according to the annotated bounding box, as shown in Fig. 10.

We split this dataset into two parts for training and testing. The training set contains 586 identities with 3,651 images, while testing set includes 147 identities with 965 images.

B. Training

For multi-class vessel detection and orientation recognition task, we use VGG [21] as our base network and SSD [16] as our detection approach to localize the vessels in the images and recognize the vessel categories and orientations. The image size in the dataset is 1080×1920, and we reduce it to 512×512 as the input size. We train the models for 240,000 iterations.

For vessel re-identification task, the base network is ResNet50 [9] which is pretrained on ImageNet [4] and take the global average pooling layer as the output. The optimizer is Adam [13] with default hyper-parameters. We set the initial learning rate to 0.0003 and exponentially decayed after 35,000 iterations with the total iteration of 50,000, while the computation of weight decay is followed by the strategy used in [10]. We select 18 vessel identities and 4 images per identity to form a mini-batch of size 72. Furthermore, we insert a dropout layer [22] after the batch normalization layer [11] for identification subnet to reduce the risk of overfitting. The trade-off parameter in equation (2) is 0.6 in our model.

C. Testing

To test the vessel detection and classification models, we use the 1,000 images as the test dataset and predict the vessel bounding box, vessel type, and vessel orientation. The confidence score threshold for both vessel type and vessel orientation is 0.5.

For the vessel re-identification testing phase, we want to get a feature extractor after training the network. So we feed a 224×224 input image into the network and obtain the result of batch normalization layer which is a 1024-dim feature embedding. After we get feature embeddings of all database images offline, we first collect the query image feature embedding online and then calculate the distance between the query image with all database images using Euclidean distance and rank the result from the smallest to the largest.

D. Evaluation metrics

There are usually two metrics adopted for pedestrian re-identification [32]. The first is cumulative matching characteristics (CMC). This metric is calculated by the first matched appearance position in the ranking list. However, CMC is accurate only when there is a single image per identity in the database. Therefore, the second metric mean average precision (mAP) was proposed by [31]. If there are multiple

Fig. 9: Ten models of vessels. The most left image is sailing ship. The four images in the first row are container, passenger, fishing and tanker. The five images in the second row are river cargo, boat, yacht, tug and taxi.

Fig. 10: Three vessel samples of VesselReID dataset.

	Classification	Orientation
TP	2,071	1,979
FP	83	140
FN	747	839
mAP	0.96	0.93
F1 score	0.83	0.80

TABLE I: Results of multi-class vessel detection and orientation.

Models	mAP	Rank1	Rank5	Rank10
Identification	35.32	55.78	73.47	79.59
Triplet	78.36	88.44	97.28	98.64
MTLnet	**81.46**	**91.16**	**98.64**	**99.32**

TABLE II: Results of vessel re-identification.

images for an identity in the database, the re-id model should return all the true matched images. In this case, if CMC values are the same for two re-id models, the recall is more significant to evaluate the performance. For multi-class vessel detection and orientation recognition, we use two metrics to evaluate our approach, which are mAP and F1-score.

E. Result and Analysis

1) ***Multi-class vessel detection and orientation recognition****:* As presented in Table I, in our 1,000 images with 2,818 labels, our multi-class vessel detection approach detects 2,071 vessel correctly and only gives 83 wrong detections. The mAP and F1 measurement are 0.96 and 0.83 respectively. The vessel orientation recognition method gives 1,979 correct predictions and presents the mAP and F1 measurement by 0.93 and 0.80.

2) ***Vessel re-identification****:* To present the actual performance improvement performed by our combination models, we train an identification model with pretrained ResNet50. Similar to the testing period, we take the output of batch normalization layer as the feature embedding and calculate the Euclidean distance to find the matched result in the database. The performance is presented in Table II. Evidently, the accuracy of mAP or CMC for identification model is very worse than our proposed model (MTLnet). This is reasonable because we split 586 classes according to the identities of vessels in the training set, while each class only has less than 20 images, especially with different orientations. It is very tough for softmax function to distinguish so many different vessels. We also train a triplet model which is based on the TriNet [10]. We can see there is a huge improvement compared with identification model. The result of the mAP for triplet model is two times larger compared to the identification model. The rank1 accuracy increases by 45% while rank5 and rank10 also increase by 63%. We can conclude that the triplet model is very suitable for the retrieval task on the small-sized dataset. Generally, the verification model achieves higher accuracy for vessel re-identification task with hundreds of identities and dozens of images per identity. This is because the triplet model takes three images at a time and compare their difference directly. But the identification model can only make the difference in the hyperplane decision.

Evidently, our MTLnet model improves the mAP,

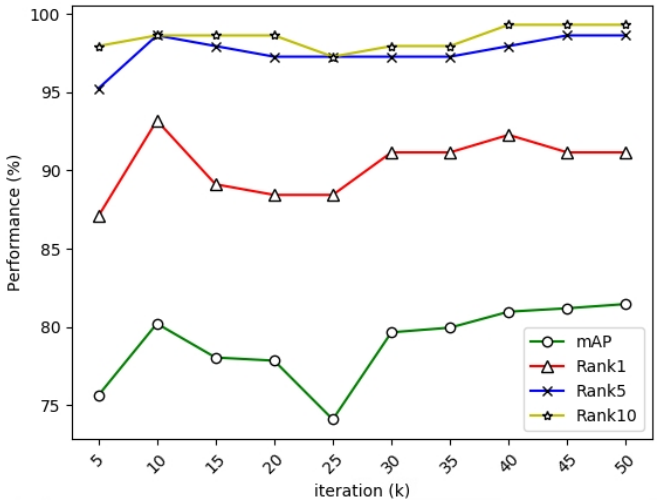


Fig. 11: Performance metrics in different training iterations.

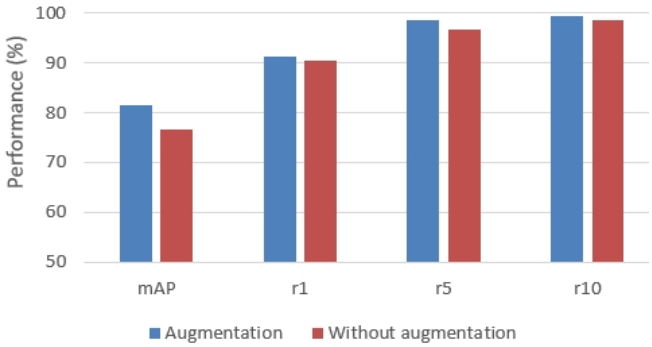


Fig. 12: Results on deployment of data augmentation.

Rank1, Rank5, and Rank10 to $81.46\%, 91.16\%, 98.64\%$ and 99.32%, respectively. This proves our expectation that the combination model strengthens the compensatory advantages and alleviate the complementary disadvantages of identification model and triplet model. As we see, the identification model takes strong labels like ID number into consideration but does not estimate the similarity between triplet pairs.

Triplet model directly compares the similarity between triplet images but only uses the weak label of positive pair or negative pair. Our MTLnet model builds the explicit relationship by triplet loss using the Euclidean distance between triplet embeddings. Meanwhile, it also constructs the implicit relationship by identification loss between all identities in the dataset. More specifically, our model constrains the final loss in two aspects. It proposes the feature embeddings of the same identity should be close to each other and far away from other identities in the feature space by triplet loss function. Simultaneously, it restricts the triplet condition in hyperplane decision that feature representations of the same identity should be similar to each other and dissimilar to other identities.

We train our model for 50,000 iterations with an initial learning rate of 0.0003 with exponential decay after 35,000 iterations. The scores of mAP and CMC are presented

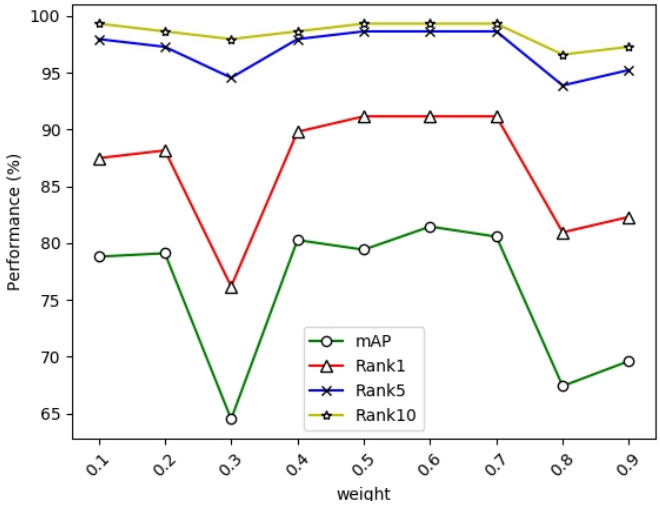


Fig. 13: The results of different trade-off parameter.

in Fig. 11. We can see the results are not stable before 35,000 iterations, which happens because the learning rate is relatively large and the model skips some local optimizations. After 35,000 iterations, the scores tend to be stable and slightly improve.

We augment our vessel images with randomly cropping and horizontal flipping in the training phase, like other pedestrian re-identification works [1,11] did. We first increase the image size by $\frac{9}{8}$ with the same aspect ratio. And then we crop the image randomly with the original size. Then we randomly choose to flip the image horizontally. The results of using data augmentation and without data augmentation are given in Fig. 12. According to this figure, if we do not use data augmentation, the mAP result is over 80%. However, if we augment the vessel images, the performance goes down to about 76%. This behavior is similar to CMC scores that the performance is poorer if we use data augmentation. Nevertheless, in pedestrian re-identification task, the model improved after using data augmentation [1]. This happens because training random parts of a vessel does not improve the model.

We explore the sensitivity of mAP scores and rank1 to the trade-off parameter λ in equation (2). As presented in Fig. 13, our proposed model achieves the best Rank1 performance when $\lambda = 0.5$, 0.6 and 0.7. But the mAP is the highest when $\lambda = 0.6$. Then we choose this value as our trade-off parameter.

To pursue the real-time vessel re-identification, we calculate the average time of discovering the same vessel in the database for a single query image. There are 147 images with 147 identities in our query set and 818 images in the database set. The total time for all 147 query images is $558.7ms$. Therefore, for a single query image, it only takes $3.8ms$ to inquiry the Euclidean distance with all database images and returns the ranking list which promises the real-time requirement.

VI. Conclusion

In this work, we propose a vessel re-identification approach to explore the problem of finding the same vessel in the database with a given vessel image. We first present a multi-class vessel detection and orientation recognition model which is based on SSD framework to find the location of vessels and predict the vessel model and orientation. Then we propose an improved vessel re-identification model based on a triplet architecture. This model combines the advantage of triplet model and identification model that it can not only learn the explicit similarity between triplet images with Euclidean distance but also learn the implicit relationship using the annotated ID labels. The empirical results demonstrate our approaches outperform the original triplet model on our vessel re-identification dataset. We also propose a vessel dataset which is annotated by the bounding box, vessel category, and vessel orientation. This dataset is captured in multiple cities in the Netherlands and Turkey with 11,000 images of 31,078 vessels under different weather conditions and different time. Besides, we present another vessel re-identification dataset which is captured in several places in the Netherlands with two non-overlapped cameras. Except for the bounding box, vessel category, and orientation, we also assign a unique id number to each vessel, while each vessel has at least two images. With the carefully-annotated 4,616 images of 733 vessels with identity number, bounding box, vessel category, and orientation, this dataset can be utilized for further research on vessel re-identification. For example, we can combine orientation recognition with the re-identification model to further improve performance.

References

[1] Ejaz Ahmed, Michael Jones, and Tim K Marks. An improved deep learning architecture for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3908–3916, 2015.

[2] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. A multi-task deep network for person re-identification. In *AAAI*, volume 1, page 3, 2017.

[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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1335–1344, 2016.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[5] Shengyong Ding, Liang Lin, Guangrun Wang, and Hongyang Chao. Deep feature learning with relative distance comparison for person re-identification. *Pattern Recognition*, 48(10):2993–3003, 2015.

[6] Mengyue Geng, Yaowei Wang, Tao Xiang, and Yonghong Tian. Deep transfer learning for person re-identification. *arXiv preprint arXiv:1611.05244*, 2016.

[7] Douglas Gray and Hai Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *European conference on computer vision*, pages 262–275. Springer, 2008.

[8] Carlos Guindel, David Martín, and José María Armingol. Joint object detection and viewpoint estimation using cnn features. In *Vehicular electronics and safety (ICVES), 2017 IEEE international conference on*, pages 145–150. IEEE, 2017.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[10] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[11] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[12] Miao Kang, Xiangguang Leng, Zhao Lin, and Kefeng Ji. A modified faster r-cnn based on cfar algorithm for sar ship detection. In *Remote Sensing with Intelligent Processing (RSIP), 2017 International Workshop on*, pages 1–4. IEEE, 2017.

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[15] Yutian Lin, Liang Zheng, Zhedong Zheng, Yu Wu, and Yi Yang. Improving person re-identification by attribute and identity learning. *arXiv preprint arXiv:1703.07220*, 2017.

[16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[17] Tetsu Matsukawa and Einoshin Suzuki. Person re-identification using cnn features learned from combination of attributes. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 2428–2433. IEEE, 2016.

[18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[19] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[20] Hailin Shi, Yang Yang, Xiangyu Zhu, Shengcai Liao, Zhen Lei, Weishi Zheng, and Stan Z Li. Embedding deep metric for person re-identification: A study against large variations. In *European Conference on Computer Vision*, pages 732–748. Springer, 2016.

[21] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[22] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[23] Rahul Rama Varior, Mrinal Haloi, and Gang Wang. Gated siamese convolutional neural network architecture for human re-identification. In *European Conference on Computer Vision*, pages 791–808. Springer, 2016.

[24] Faqiang Wang, Wangmeng Zuo, Liang Lin, David Zhang, and Lei Zhang. Joint learning of single-image and cross-image representations for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1288–1296, 2016.

[25] Lin Wu, Chunhua Shen, and Anton van den Hengel. Personnet: Person re-identification with deep convolutional neural networks. *arXiv preprint arXiv:1601.07255*, 2016.

[26] Shangxuan Wu, Ying-Cong Chen, Xiang Li, An-Cong Wu, Jin-Jie You, and Wei-Shi Zheng. An enhanced deep feature representation for person re-identification. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–8. IEEE, 2016.

[27] Tong Xiao, Hongsheng Li, Wanli Ouyang, and Xiaogang Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 1249–1258. IEEE, 2016.

[28] Guang Yang, Bo Li, Shufan Ji, Feng Gao, and Qizhi Xu. Ship detection from optical satellite images based on sea surface analysis. *IEEE Geosci. Remote Sensing Lett.*, 11(3):641–645, 2014.

[29] Ruiqian Zhang, Jian Yao, Kao Zhang, Chen Feng, and Jiadong Zhang. S-cnn-based ship detection from high-resolution remote sensing images. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 41, 2016.

[30] Yiheng Zhang, Dong Liu, and Zheng-Jun Zha. Improving triplet-wise training of convolutional neural network for vehicle re-identification. In *Multimedia and Expo (ICME), 2017 IEEE International Conference on*, pages 1386–1391. IEEE, 2017.

[31] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1116–1124, 2015.
[32] Liang Zheng, Yi Yang, and Alexander G Hauptmann. Person re-identification: Past, present and future. *arXiv preprint arXiv:1610.02984*, 2016.
[33] Zhedong Zheng, Liang Zheng, and Yi Yang. A discriminatively learned cnn embedding for person reidentification. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(1):13, 2017.
[34] Zhengxia Zou and Zhenwei Shi. Ship detection in spaceborne optical image with svd networks. *IEEE Transactions on Geoscience and Remote Sensing*, 54(10):5832–5845, 2016.

APPENDICES

A. Hard positives and hard negatives

In section IV-A, we discuss hard positives and hard negatives of triplet model. Fig. 14 illustrates some examples of hard positives and hard negatives. Images with blue border and red border represent true matched and false predicted images respectively. The first row is the query vessel images. Following three images with blue border represent the top3 matched images in the database. Our MTLnet model recognizes these true images from the database and gives a high ranking. The images with the red border are the first false samples appears in the ranking list, which represents the *hard negatives*. We can see the appearances of these vessels are very close to the query images. In addition, the hard negative of the left query image has the different orientation with the query. However, if we observe the two images, the color of forecastle is the same and the bridge of the vessel in the two images is also similar. This also proves that our MTLnet learns the metric that orientation change has no strong influence on the re-identification, despite this is a negative image. We can improve this performance in the future work. The last row is the true matched images but with the largest distance to the query images, which represents *hard positive*. Some of the hard samples are easy to distinguish for human, but not for the machine, such as the hard negative image and hard positive image of the middle query.

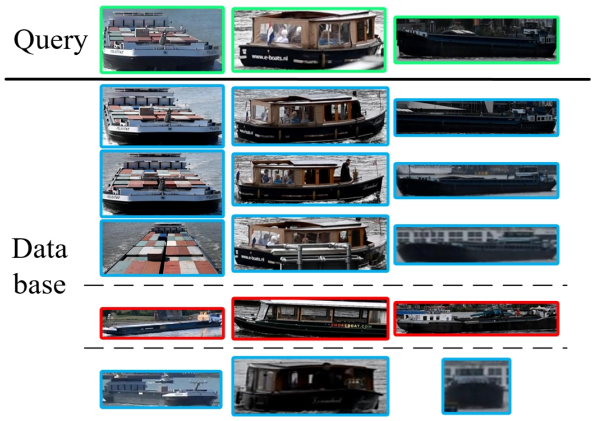


Fig. 14: Some of the hardest samples. The top row represents query images. Followed by the top3 database images discovered by our MTLnet model. The images with the red border are the retrieved mistakes (hard negative). The last row represents the correct matches in the database with the largest distance to the query images (hard positive).

layer name	VGG16	ResNet50
conv1_x	$3\times3, 64$ $3\times3, 64$	$7\times7, 64$
maxpool	2×2	3×3
conv2_x	$3\times3, 128$ $3\times3, 128$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$
maxpool	2×2	—
conv3_x	$3\times3, 256$ $3\times3, 256$ $3\times3, 256$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times4$
maxpool	2×2	—
conv4_x	$3\times3, 512$ $3\times3, 512$ $3\times3, 512$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times6$
maxpool	2×2	—
conv5_x	$3\times3, 512$ $3\times3, 512$ $3\times3, 512$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$
pool	maxpool	averagepool
FC	4096	—
FC	4096	—
FC	1000	1000
softmax		

TABLE III: Network definition of VGG16 and ResNet50.

B. Image classification network definition

In this work, we use two image classification network as our base networks, which are VGG16 [21] and ResNet50 [9]. The network definitions of two models are listed in Table III. In addition, we only take the convolutional layer and fully connected layer into account. So the layer amount does not include pool or activation layer. For VGG16, there are 13 convolutional layers and three fully connected layers. For ResNet50, there are $3+4+6+3=16$ building blocks. Each building block contains three convolutional layers. As these two networks are trained on ImageNet [14] which has 1000 object classes, the output of the final fully connected layer is a 1000-dims vector.

C. Parameters during training process

Fig. 15 - Fig. 18 illustrates four parameters of our MTLnet model during the training process. Fig. 15 presents the top3 accuracy in a mini-batch. We can see our network learns the similarity metric rapidly since the accuracy achieves to 100% at around 3000 iterations. The final loss curve in Fig. 16 also illustrates this since it drops rapidly at the beginning. The triplet loss tends to reduce to 0 and has no enormous variations, while the identification loss fluctuates but has a downward trend.

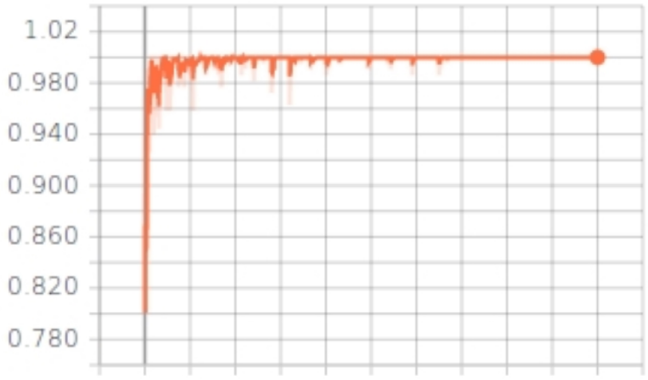


Fig. 15: The batch top3 predicted accuracy.

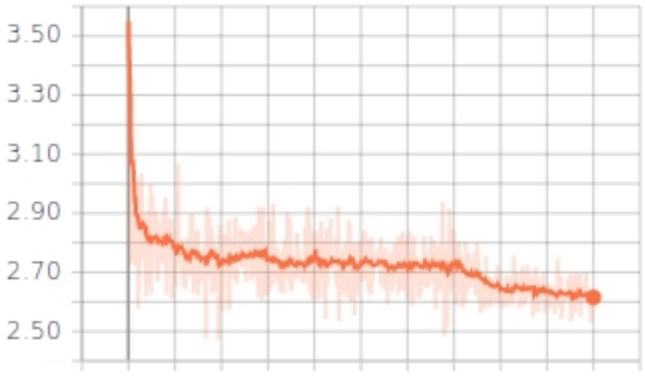


Fig. 16: The final loss.

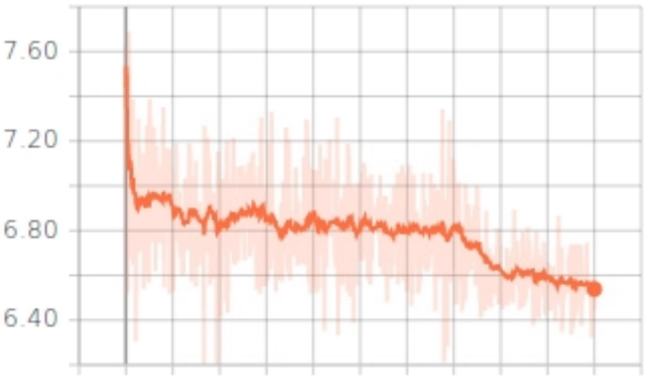


Fig. 17: The identification loss.

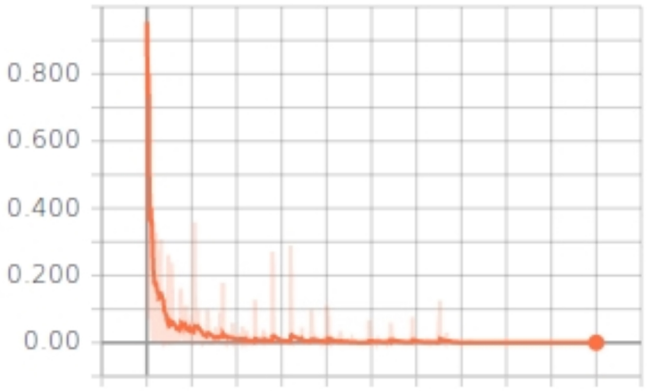


Fig. 18: The triplet loss.