HAT-CNN: Harmonious Attention Attribute Convolutional Neural Network

Asli Alpman, Sencer Umut Balkan
Ihsan Dogramaci Bilkent University
Ankara, Turkey

alpman@ee.bilkent.edu.tr, umut.balkan@ug.bilkent.edu.tr

https://github.com/aslialp/HATCNN

Abstract

With the increasing popularity of the video surveillance systems, matching different images of the same person coming from different cameras and time-frames, which is the person re-id task- and performing this task in a short time duration has become a need. However, it is a rather challenging task due to the pose variance, illumination changes and occlusion due to the dynamic nature. Additional information such as pose information, attention information, additional annotations or saliency information has proved to be useful in handling those challenges. This paper uses a light-weighted soft and hard attention learning convolutional neural network as its baseline and integrates manually annotated attribute information into the solution to further improve the accuracy of the person re-identification. Resulted design learns to predict specific set of attributes, to extract feature representation from person IDs and to learn soft and hard attention harmoniously.

1. Introduction

Person re-identification (re-id) tasks aim to retrieve the identity of the person queried by utilizing existing images at the gallery set. This task can be categorized into two classes which are image-based and video-based person re-identification. Image-based person re-id uses single images captured by various cameras to re-identify the query person while video-based person re-id uses video of the query person [13]. In this paper, we will only address image-based person re-id problems.

Further, it is possible to formulate person re-id problem in two different ways. Firstly, the problem of finding the identity of the given query image can be regarded as an image classification task [9]. With this mindset, the task is to assign each distinct ID to a class and to find the ID class which is mostly activated by the given query image.

Secondly, person re-id problem can be formulated as a discriminative embedding feature representation extraction task. With this mindset, discriminate features are extracted from the given images and the feature vectors form an embedding space where each image is represented as a vector. Then, it is possible to evaluate various distance metrics between those vectors and conclude the nearest ones to the given vector. Since each vector represents a person ID, one can conclude about the ID of the query image by looking at the IDs of the nearest vectors in the embedding feature space. This paper utilizes embedding space representation and thus only embedding space models will be discussed.

One of the main challenges of the person re-id is that the images are captured with non-overlapping cameras. This causes a serious pose variance problem even between the images of the same person. It is important to note that person re-id is mainly used in the multi-camera surveillance systems where cameras are placed in a nonoverlapping fashion and in distance from the subjects to be monitored[A survey of approaches and trends in person re-identification]. Therefore, images of the particular subjects are also low resolution due to the long distance between the monitoring cameras and the subjects monitored.

Note that only a very small portion of the image, coming from the cameras, belongs to a specific person ID. At this point, the issue of cropping the images of individual persons from the raw camera data arises. There are several methods employed to crop ID images from raw camera data [9]. One can either manually crop the raw images or use person detection and tracking algorithm to generate bounding boxes. Since manual cropping is cumbersome and becomes almost impossible with the increasing number of images, bounding boxes created algorithms are generally preferred. However, it is possible that bounding boxes may make cropped images suffer from occlusion. Subjects may already be occluded due to the dynamic nature of the scene captured by the cameras and bounding boxes may increase the suffering from occlusion.

Due to the pose variance, occlusion, low-resolution problem mentioned above, additional information other than the person IDs may significantly increase the performance of the person re-id. Additional information may be attention, saliency, attribute, pose information etc. The contribution of this paper can be summarized as follows.

- We are going to integrate attribute information to a baseline network which learns attention information and performs person re-identification simultaneously. As end result, our network are going to learn to predict specific set of attributes, to extract feature representation from person IDs and to learn soft and hard attention harmoniously.

2. Related Work

We are going to group additional information into 4 groups and address them one by one.

2.1. Attribute Information

Besides giving person IDs, this approach employs various attributes to further increase the performance of the network. Lin et. al. [4] manually labelled some attributes for Market-1501 [12] dataset and the DukeMTMC-reID [14] dataset.

Attributes to be annotated are selected such that they would represent persons in the dataset as complete as possible. To illustrate, most of the people in Market-1501 dataset wear dresses or shorts. However, people generally wear pants in DukeMTMC-reID dataset [4], therefore, it is not reasonable to use dress or short attribution in DukeMTMC-reID dataset.

For Market-1501 dataset, which is the interest of this paper, 27 attributes are annotated [4]. These are male or female, long or short hair, long or short sleeve, short or long lower body clothing, pants or dress, wearing a hat or not, backpack or no backpack, handbag or no handbag, having other types of bags or not, 8 colours of upper-body clothing (which are black, white, red, purple, yellow, grey, blue, green), 9 colours of lower-body clothing (which are black, white, red, purple, yellow, grey, blue, green, brown) and age (child, teenager, adult, old). Below figure shows an example of these attributes for a specific person image.

It is important to note that attributes are annotated considering person identity rather than the individual images [4]. Although some of the attributes which are known to be of that person are occluded in one image, the image will still be annotated with that attribute if previous images of the same person conclude the existence of that attribute. For example, a particular image of an ID may not include hat due to occlusion but it is annotated with ”hat” if other images of the same person have a hat.

It is also possible to use attributes to extract heatmaps and attention maps to determine where to look in the image. Class Activation Maps (CAM) [15] is one way to extract heatmaps and attention maps. The output of CAM for an attribute is the discriminative region which most activates this attribute probability in the network.

By utilizing both attributes themselves and the heatmaps and attention maps extracted from those attributes, the benefit of the attribute information can be increased. After combining ID information with the attribute and attribute heatmaps, the performance of the network can further be improved [7].

2.2. Saliency Information

Saliency means uniqueness which would make one element stands out among specific set. Using saliency information can make person re-id algorithm more robust to the pose variance, illumination or occlusion problems. Once the salient features or patches are learnt, ranking of the embedding space representations can be adjusted to give more importance to the matching of the salient features compared to the other ones. Saliency information can be learned in image patch level or feature level. Support Vector Machines (SVM) or k-Nearest Neighbours (kNN) algorithms may be used to determine outliers and saliency in our context [11].

2.3. Pose Information

Pose variance is one of the biggest challenges for the person re-id problems and it may even make intra-person variance bigger than the inter-person variance. Thus, pose information can greatly increase the performance of the re-identification. There are several ways to integrate pose information into the re-id problem. One of them is that image parts can be transformed and modified by using pose information to obtain one common pose for all images. Network fed with uniform pose images would not suffer from pose variance and thus perform better [5].

Another approach is to integrate pose and angle information of the cameras rather than pose information of the individuals [8]. This approach only requires calibration of the camera pairs and thus require less computation compared to the first approach. However, it can only resolve pose variance due to the camera replacement. Rotating subjects will also cause pose variance which can not be solved by the camera calibration.

2.4. Attention Information

Attention information, as the name implies, tells the algorithm where to look. Features with more attention attached to them play a bigger role in determining embedding representation. Attention is twofold which are soft attention and hard attention. Soft attention gives the fine-grained, mostly pixel-wise, regions which should be treated with more care. Hard attention gives the bigger regions which are rich in discriminative features. Attention information can be found in unsupervised ways [3]. It is also possible to regularize attention information learning by enforcing various constraints such as orthogonality [1]. Orthogonality regularization improves the overall performance since the orthogonalized attention is distributed all over the image. Having features from all over the image, excluding junk features with the help of attention, makes re-id task more robust to the challenges mentioned.

3. Proposed Approach

3.1. Baseline Architecture

Our design uses Harmonious Attention Convolutional Neural Network HA-CNN) as its baseline [3]. This network does not stack deep CNNs to avoid slow training -which would be a disadvantage in surveillance problems- and overfitting -which is common in re-id tasks due to the small number of labelled train data. HA-CNN consists of two main branches which are global branch and local branch. The global branch takes the whole person image and tries to extract global feature representation from that image. The local branch consists of 4 identical subbranches (streams). Each stream takes a part of the input image and tries to extract feature representation of that part. In other words, different streams extract local feature representations coming from different image regions. Both branches use InceptionA-InceptionB as building blocks which will be explained in next section.

3.1.1 Harmonious Attention Learning

This network simultaneously learns complementary soft and hard attention. Soft attention maps are used in the global branch while hard attention information is used to determine the image regions fed into the local streams.

For this purpose, Harmonious Attention (HA) modules are placed in different levels of the network. HA modules composed soft spatial-channel attention learning part and hard attention learning part.

3.1.2 Soft spatial-channel attention learning

This part takes input with size $(hxwxc)$ where h is the number of pixels in height, w is the number of pixels in weight and c is the channel dimensions. It aims to extract channel attention and soft attention to combine both in one saliency map. Learning of the channel and soft attention are done in a factorized way. Eventually, soft and channel attention is blended to obtain the saliency map by using learnt parameters and tensor multiplication. Saliency map has the same dimensions as the input image and weights the input vector according to the learnt attention by simple multiplication.

Spatial Attention Branch is a tiny 4 layers network composed of the followings.

- *Global cross-channel averaging pooling layer*: this layer averages input with respect to the channels to remove channel information. By this way, all channels share the same spatial vector representation
- *Convolutional layer*: It has one kernel with the size of 3x3 and stride as 2.
- *Resizing Bilinear Filter*: This filter undoes the effect of the previous convolutional layer in size.
- *Scaling convolutional layer*: This layer simply does scalar multiplication of its input. Here, the multiplier is learnt by the network to optimally combine the output of this branch with the channel attention branch.

Channel Attention Branch is a 3 layers sub-network.

- *Global averaging pooling layer*: This layer averages input with respect to the spatial space to remove spatial information. By this way, all spatially different pixels share the same channel
- *2 convolutional layers*:Two consecutive convolutional layers may be combined into one, however, the number of parameters to be learnt is reduced with this two-folded design

The outputs of the spatial and channel attention branches are multiplied at the end. Since spatial and channel attention is not exclusive but complementary, another convolution layer is used to further blend the outputs of soft and channel attention. Finally, a sigmoid layer is employed for normalization purposes.

3.1.3 Hard Attention Learning

This part aims to find the location of the 4 different image parts which are more discriminative. Location image parts are done with learning a transformation matrix T.

$$T = \begin{bmatrix} s_h & 0 & t_x \\ 0 & s_w & t_y \end{bmatrix}$$

s_h and s_w are the scaling factors in height and weight respectively. t_x and t_y are the translation parameters in x dimension and y dimension. While scaling factors are predefined in the design, translation factors are learnt by the network. Learnt translation factors are used to sample the image around (t_x, t_y) with the size determined by the predefined scale factors (s_h, s_w)

For this purpose, HA modules have a sub-network with 2 layers. It takes its input as the first layer output of the channel attention part. Hard attention learning sub-network is composed of the followings.

- *Fully connected neural network*: The purpose of this network is to learn translational parameters from the given input.
- *Tanh scaling*: It is possible to have discriminative regions with some part outside of the image boundary due to the occlusion or wrong cropping. In those cases, it is reasonable to use percentage parameters instead of positional parameters. The purpose of the tanh scaling is to convert position information to some kind of ratio or percentage.

3.1.4 Cross Attention Interaction Learning

We previously mentioned that HA-CNN has two main branches which are global and local branches. Cross attention interaction learning provides harmony between those two branches. Local feature representation is enhanced by adding global branch representation of the same level.

$$\tilde{X}_L^{(L,k)} = X_L^{(L,k)} + X_G^{(L,k)}$$

where $\tilde{X}_L^{(L,k)}$ is the modified local feature representation, $X_L^{(L,k)}$ is the local feature representation of region k from level L, $X_G^{(L,k)}$ is the global feature representation of region k from layer L and equivalently it is the hard regional attention for region k at level $(L+1)$.

It means that the global branch is also learning from the local branches.

$$\Delta W_G^{(l)} = \frac{\partial Loss_G}{\partial X_G^{(l)}} \frac{\partial X_G^{(l)}}{\partial W_G^{(l)}} + \sum_{h=1}^{4} \frac{\partial Loss_L}{\partial \tilde{X}_L^{(l,k)}} \frac{\partial \tilde{X}_L^{(l,k)}}{\partial W_G^{(l)}}$$

3.2. Design and Architecture

Our design uses HA-CNN as its baseline. Different than HA-CNN, our network does not only use attention as additional information. We are also using attribute information to further improve the performance of the person re-identification.

We tried several ways to integrate attribute information into the HA-CNN design. One of them was to train the local branch with only attribute labels. Another one was to train the local branch with both ID labels and attribute labels. Training a separate network to only predict attribute labels was the last approach that we employed and it turned out to be the most successful integration scheme among our trials. In this paper, we will mainly focus on the last integration scheme which is training a separate network for attribute learning however, results for the other approaches will also be shown in the next sections.

The attribute learning problem is modelled as a multi-class labelling problem. In other words, each given input may belong to more than one classes which is a requirement for attribute learning since it is almost certain that one person will have more than one attribution among the labelled ones. Since our main purpose is to extract features to predict attributes, we took a similar approach with a feature extractor part of the HA-CNN.

To decrease the training time, raw images are resized into $(160x64x3)$. Resized images are first fed into the convolutional layer to further decrease the computational complexity. After that, consecutive InceptionA and InceptionB layers are used for feature extraction. The reason we employed Inception blocks is that they already proved to be useful in feature extraction part of the HA-CNN. Composition of the InceptionA and InceptionB blocks are disccussed in the following subsection.

3.2.1 Inception Blocks

Each InceptionA block is composed of 4 streams. Three of them are identical and uses 2 consecutive convolutional layers. Batch normalization and reLU are applied to the output of those networks. The fourth layer applies average pooling to the input before feeding it to a convolutional layer. Output vectors of 4 streams are concatenated and given as the output of InceptionA block.

InceptionB blocks are composed of 3 streams. One of them has 2 convolutional layers, batch normalization and reLU steps. Another one has a similar design but has 3 convolutional layers instead of 2. The third one applies max pooling to the input before feeding it to a convolutional layer. Output vectors of these 3 streams are concatenated and given as the output of InceptionB block.

3.2.2 Loss Function

Baseline HA-CNN uses cross entropy as loss function for both local and global branches.

$$Loss_L = \sum_{n=1}^{M} \sum_{o=1}^{N_n} y_{(o,n)} log(p_{local}(o,n))$$

$$Loss_G = \sum_{n=1}^{M} \sum_{o=1}^{N_n} y_{(o,n)} log(p_{global}(o,n))$$

where M is the number of classes, N_n is the number of instances of class n, $y_{(o,n)}$ is the label of the observation o for class n, $p_{local}(o,n)$ is the probability of observation o being in the class n predicted by local branch, $p_{global}(o,n)$ is the probability of observation o being in the class n predicted by global branch.

Attribute learning part uses also binary cross entropy loss in training process.

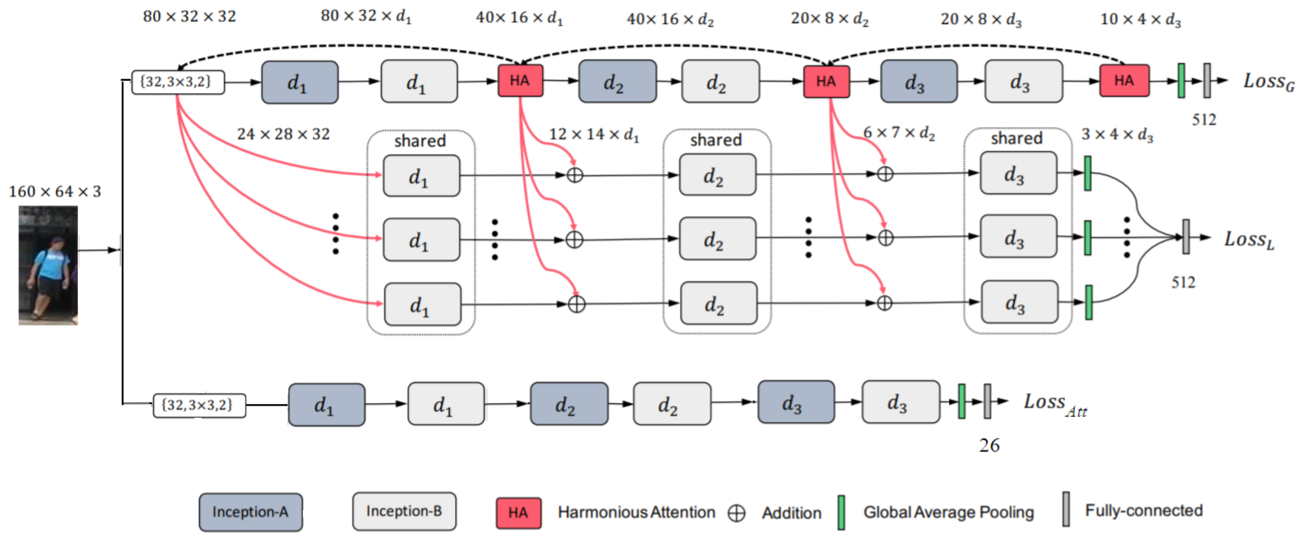

Figure 1. Architecture of HAT-CNN.

$$Loss_{att} = \sum_{n=1}^{M_{att}} \sum_{o=1}^{N_att} y_{(o,n)} log(p_{att}(o,n))$$

where M_att is the number of annotated attributes, N_n is the number of instances for attribute n,$y_{(o,n)}$ is the label of the observation o for attribute n $p_{att}(o,n)$ is the probability of observation o having the attribute n predicted by the attribution learning branch.

4. Results

4.1. Dataset

Experiments are conducted on the Market1501 Dataset which is one of the most popular image-based person re-id [12]. It is also one of the two datasets which have manually annotated attributes [4]. It is composed of coloured images with size of $(64x128x3)$ in .jpg format. Dataset has 1501 different person IDs and 32.668 images in total. The train set has 751 IDs and 12.936 images while test (query) set has 750 IDs and 3368 images total. Left 15.913 images are not used in either training or test but they are in the gallery set which forms the embedding space.

We used 26 out of 27 attributes which are male or female, long or short hair, long or short sleeve, short or long lower body clothing, pants or dress, wearing a hat or not, backpack or no backpack, handbag or no handbag, having other types of bags or not, 8 colours of upper-body clothing (which are black, white, red, purple, yellow, grey, blue, green) and 9 colours of lower-body clothing (which are black, white, red, purple, yellow, grey, blue, green, brown).

4.2. Metrics

4.2.1 Rank-1 Accuracy

Rank-1 accuracy in embedding re-id problems equals the probability of having the correct label as the nearest neighbour in the feature vector space. Similarly, rank-k would mean the probability of having the correct label in the nearest k neighbours in the feature vector space.

4.2.2 Mean Average Precision(mAP)

Mean average precision calculates the precision of the retrievals from gallery set by considering the ranking in the set. maP calculation procedure is as follows. First, the query feature vector representation is placed into the embedding space. Then, gallery vectors are ranked in terms of their distance to the given query vector. There are various distance metrics such as Euclidian norm, infinity norm, Manhattan norm etc. We are using Euclidian norm as the distance metric.

Starting from the nearest feature vector, we are retrieving gallery vectors and checking if the ID of the query match with the ID of the gallery vector. For each retrieved vector whose ID matches with the ID in the query, we calculate the ratio of the number of true matches until that point to the total retrieved items. At the end of this procedure, we sum all those ratios up to find mean of them which is the mean average precision. Formulas for the average precision and mean average precision calculations are given below.

$$AP_L^{(i)} = \frac{1}{M}\sum_{k=1}^{i} p(k)I(k, L)$$

where $AP_L^{(i)}$ is the average precision for class L at rank i, M is the number of instances belonging to the class of interest(L), $p(k)$ is the precision at the rank k which is the ratio between number of retrieved instances from correct class L and the number of all retrieved instances, $I(k, L)$ is the indicator function which is 1 if the k-th element in the ranking belongs to the class L.

$$mAP = \frac{1}{NK}\sum_{j=1}^{K}\sum_{i=1}^{N} AP_j^{(i)}$$

where K is the number of classes and N is the number of the total gallery elements.

4.3. Results

Our network is trained with stochastic gradient descent. We employed multi step learning rate scheduler which starts from 0.3 and decreases to one tenth at epoch 150 and epoch 225.

The proposed approach is tested on Market1501 dataset and following results are obtained. Figure 2 shows the change in the mean average precision with the increasing number of epochs for two different scenarios. In fact, those two results belong to the same network with the identical trained weights however, one of them uses its attribute predictions for ranking while other does not. Similarly, Figure 3 shows the change in the rank-1 accuracy with the increasing number of epochs for the same network.

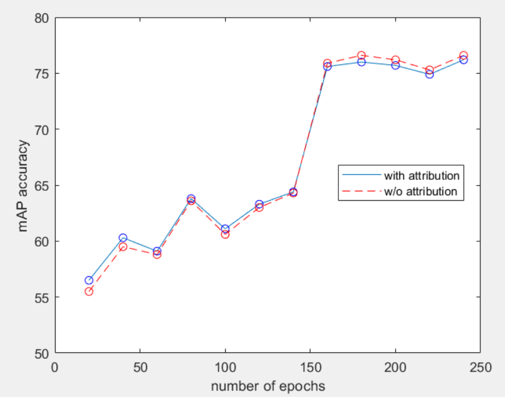

Figure 2. Effect of the attribution use in retrieval process on mAP metric.

One can conclude from figure 2 and 3 that the attribution information is useful especially in the earlier epochs.

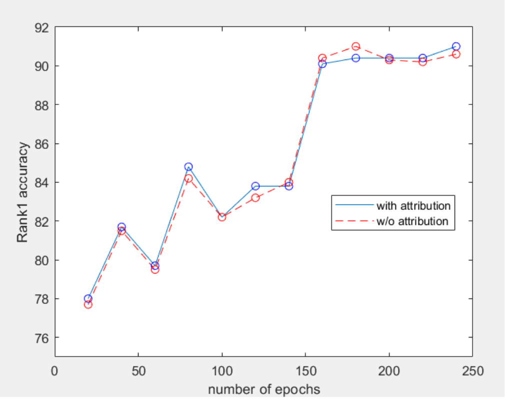

Figure 3. Effect of the attribution use in retrieval process on Rank1 metric.

We see that the results using attribution information is relatively better until epoch 150. After that point, we observe that performance of the non-attribute scheme catches and even passes the one using attribute information. The reason behind this may be that the network which extracts global and local features is not well trained in the earlier epochs and thus attribute information enhances the overall performance. After global and local feature extractors are trained well enough then additional attributions are not useful anymore since other extractors were already extracting correlated or maybe even the same information. Further, this may be the reason why the results with attributions are little behind from the non-attribute results in the later epochs. It is because same attribute is found both by global-local extractors and by the attribute extractor and thus those attributes become weighted in the ranking process with greater importance. Focusing to those attributes more than required may be the reason why the performance deteriorates with the increasing number of epochs.

We also tried another scheme for integration attribute information into HA-CNN. In this design, attributes are also learned from the local branch extractor. In other words, local branch predicts both IDs and attributes. Figure 4 shows the mAP and rank-1 results for this scheme.

Results in Figure 4 concludes that an independent network for attribute extraction performs better compared to the integrated attribute and local feature extraction. Especially, mAP results are lower about 5% in the integrated design. This may be due to the fact that the attribute annotations for Market1501 dataset are done in the identity level. For example, even if the image of the person does not contain a hat (it may be the case that s/he takes his/her hat off and then put it back), it is labeled with hat if we know that most of the images of that person is with hat. This identity level labelling may be causing confusion in learning of the local features either since the weights are common at the

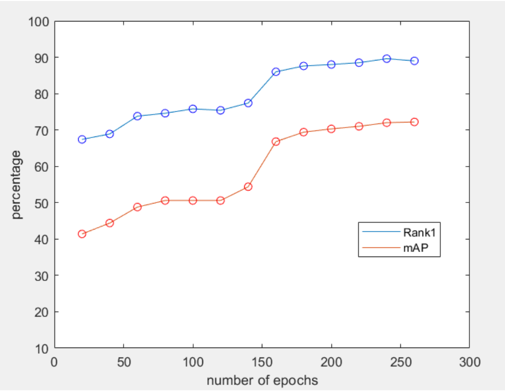

Figure 4. Rank1 and mAP results for the case where attributions are learnt in the local branch.

Method	Rank-1	mAP
MSCAN [2]	80.3	57.5
DLPA [10]	81.0	63.4
PDC [6]	84.1	63.4
HA-CNN [3]	91.2	75.7
HAT-CNN	91.0	76.7

Table 1. Market1501 evaluation for various networks.

previous layers.

Further, our design improves the HA-CNN in terms of metric mAP as shown in below table.

5. Conclusion

In this paper, we tried to integrate attribution information with the attention information by using HA-CNN as our baseline model. The results for two different integration scheme were given in the paper. First of them was to utilize local feature extractor branch of the HA-CNN as attribute extractor. We simply fed local branch with the combination of person IDs and attributes. Second integration method was to construct an independent network for merely attribute extraction task.

Results showed that using independent network for feature extraction performed better compared to the other approach. We attributed this result to the fact that the attributes are annotated in identity level which may confuse the local feature extraction learning process and decrease the re-id performance.

Further, we analyzed the effect of having attribute prediction for the overall performance of the person re-identification. We concluded that additional attributes are useful in earlier training stages however, they are not quite useful for well-trained networks. The reason may be that learning the same features and having them in vector representation multiple times can cause to overweight that feature.

Asli Alpman implemented the project and prepared the paper, slides and gitHub page. Sencer Umut Balkan did research about the topic of the project and the chosen project topic.

References

[1] T. Chen, S. Ding, J. Xie, Y. Yuan, W. Chen, Y. Yang, Z. Ren, and Z. Wang. Abd-net: Attentive but diverse person re-identification. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 3

[2] X. C. D. Li, Z. Zhang, and K. Huang. Learning deep context-aware features over body and latent parts for person. *Conference on Computer Vision and Pattern Recognition*, 2017. 7

[3] W. Li, X. Zhu, and S. Gong. Harmonious attention network for person re-identification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3, 7

[4] Y. Lin, L. Zheng, Z. Zheng, and C. Y. Y. Y. Y. Wu, Z. Hu. Improving person re-identification by attribute and identity learning. *Pattern Recognition*, 2019. 2, 5

[5] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian. Pose-driven deep convolutional model for person re-identification. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 2

[6] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian. Posedriven deep convolutional model for person re-identification. *International Conference on Computer Vision*, 2017. 7

[7] C.-P. Tay, S. Roy, and K.-H. Yap. Aanet: Attribute attention network for person re-identifications. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 2

[8] W. L. Yang Shen, unchi Yan, M. Xu, J. Wu, and J. Wang. Person re-identification with correspondence structure learning. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[9] M. Ye, J. Shen, G. Lin, L. S. T. Xiang, and S. C. H. Hoi. Deep learning for person re-identification: A survey and outlook. *arXiv*, 2020. 1

[10] L. Zhao, X. Li, J. Wang, and Y. Zhuang. Deeply-learned part-aligned representations for person re-identification. *International Conference on Computer Vision*, 2017. 7

[11] R. Zhao, W. Ouyang, and X. Wang. Unsupervised salience learning for person re-identification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[12] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. *IEEE International Conference on Computer Vision*, 2015. 2, 5

[13] L. Zheng, Y. Yang, and A. G. Hauptmann. Person re-identification: Past, present and future. *arXiv*, 2016. 1

[14] Z. ZHENG and L. ZHENG. A discriminatively learned cnn embedding for person reidentification. *ACM transactions on multimedia computing communications and applications*, 2017. 2

[15] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2