VI Semester Minor Project Report
ON

Person Re-Identification via Deep Learning

Submitted by

Komal Krishna Panigrahi- 191020430
Basharat Khalid Harris- 191020417
Adarsh Ranjan Gupta- 191000004

Under the guidance of

Dr. Vivek Tiwari

Department of COMPUTER SCIENCE AND ENGINEERING

Dr. Shyama Prasad Mukherjee

International Institute of Information Technology, Naya Raipur

(A Joint Initiative of Govt. of Chhattisgarh and NTPC)

Email: iiitnr@iiitnr.ac.in, Tel: (0771) 2474040, Web: www.iiitnr.ac.in

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Dated: May 12, 2022

Komal Krishna Panigrahi (191020430)
Basharat Khalid Harris (191020417)
Adarsh Ranjan Gupta (191020455)

Certificate

This is to certify that the project titled "PERSON RE-IDENTIFICATION VIA DEEP LEARNING" by *Komal Krishna Panigrahi, Basharat Khalid Harris and Adarsh Ranjan Gupta* has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree/diploma.

Dated: May 12, 2022

Dr. Vivek Tiwari
Assistant Professor Department of CSE
Dr. SPM IIIT Naya Raipur- 493661
May 2022

Approval Sheet

This project report entitled “PERSON RE IDENTIFICATION VIA DEEP LEARNING” by *Komal Krishna Panigrahi, Basharat Khalid Harris and Adarsh Ranjan Gupta* is approved for 6th Semester Minor Project.

(Signature of Examiner - I)

____

Name of Examiner -I

(Signature of Examiner - II)

____

Name of Examiner -II

(Signature of Chair)

____

Name of Chair

Date: _____________ Place: _____________

Acknowledgments

At the outset, I would like to express my wholehearted and deep sense of gratitude to my guide **Prof. Dr. Vivek Tiwari** for his guidance, help and encouragement throughout my research work. I greatly admire his attitude towards research, creative thinking, hard work and dedication in work. I am highly grateful to him for patiently checking all my manuscripts and thesis. This thesis would not have been possible without his bounteous efforts. More than a guide, he is my mentor for shaping my personal and professional life, without whom I would not have been where I am today. I owe my profound gratitude to Prof. Dr. Vivek Tiwari for his support in all respects.

Komal Krishna Panigrahi (191020430)
Basharat Khalid Harris (191020417)
Adarsh Ranjan Gupta (191000004)

Abstract

Person re-identification (re-ID) is the technique of correlating photos or videos of the same person captured from various perspectives and cameras. It has immense societal applications in our day-to-day life like video surveillance, culprit identification etc. Despite this, it has always remained a difficult process even with the recent progress, due to the significant differences in human looks from various capture angles.

In object identification problems, negative training examples are the negative examples we provide our classifier at the start of the training process. Then we go out and run new data through our trained model, only to find that we've picked up on false positives. When these erroneous positives are added to our first training data as negative examples, they are referred to as hard negatives. Existing deep metric learning-based re-ID algorithms frequently suffer from two flaws: a) Most works based on pairwise or triplet constraints have delayed convergence and poor local optima and hard negative sample mining has been frequently used in existing studies. b) Hard positive samples, on the other hand, which contribute to network training, have gotten insufficient attention.

Therefore, we are proposing to develop an end-to-end approach to tackling the problem by learning feature representation and distance metric together, by which similar samples will be kept closer together and dissimilar samples will be pushed farther away. This will result in a low-dimensional embedding space where person images can be nonlinearly mapped. We are also planning to develop the suggested technique using the inception architecture, test it on re-ID data sets, and compare them with our current state of the art methods.

Table of Contents

Title **Page No.**

List of All Figures and Tables

Chapter 1

Introduction

This project considers the task of improvement from existing *Person Re-identification* (ReID) models. Person ReID, also known as person retrieval, aims at solving the problem of matching and identifying people from changing views of cross cameras. As an important intelligent video analysis technology, person reidentification is widely used in the fields of intelligent security, case detection, lost query, intelligent interaction, and so on [2]. There are many challenges involved in the process as well. The huge difference in illumination, occlusion, resolution, background, clothing, and changing posture, in the actual scene, and the amount of data available at instant. There are two most used methods, feature learning, and the other is metric learning, in the field of person ReID. Deeply learned representations provide high discriminative ability, especially when aggregated from deeply learned part features [1].

The development of person re-id methods seeks robust person matching through combining various feature types because it is hard to further improve the performance using only general visual features due to appearance ambiguity. For, example, let's consider two different person such that they have same surroundings and ambience, similar clothing, and other factors. The problems comes when they both have different actions and pose. Our model is focused on how to exploit extra information to get around this bottleneck.

The previous models and studies attempt to exploit person structure information to improve the performance of ReID methods. Thus, they were focused very much on exploiting a person's structure information to improve the performance of ReID methods. Considering this approach, the model can map full body's attributes with local body-part features, and the pose-invariant features by exploiting key point annotations of the poses, creating a pose-based approach. Other methods also include surrounding features, background features, semantic segmentation (labelling of pixels) etc., but always failed to tackle the ambiguity problem.

Considering the challenges, we proposed a framework to develop an end-to-end approach or combining the learning feature representation and distance metric together. Through this we can group the images/data with similar features together and distinguish them from dissimilar data. This will give us a low-dimensional embedding space where the person's images can be nonlinearly mapped. We are also planning to develop the suggested technique using the base architecture, test it on re-ID data sets, and compare them with our current state of the art methods. We proposed the framework that exploits both the visual semantic similarity and the spatial information in the image for person re-ID.

Chapter 2

Literature Survey

Long before deep learning algorithms dominated the re-ID research field, hand-crafted algorithms had created methodologies to learn portion or local characteristics. Gray and Tao [6] divides the pedestrians into horizontal stripes to extract color and texture information. Gheissari et al. [7] separated the pedestrian into triangles for part feature extraction, and many other publications [8, 9, 10, 11] have used similar divisions. Cheng et al. [12] use graphic structure to parse the pedestrian into semantic pieces.

Deep learning for visual feature representation is the focus of recent person Re-ID algorithms. Essentially, these deep models try to create successful convolutional neural networks or use various loss functions, such as classification loss [13], verification loss [14], and triplet loss [15]. These algorithms obtain high performance due to the exceptional ability of CNN representation; however, these methods cannot handle the appearance ambiguity problem. Many studies have attempted to use personal structural information to alleviate this limitation [16]. A multi-scale context-aware network [17] is employed to capture local context information to learn powerful features spanning whole body parts. A pose-driven deep convolutional model [18] is introduced to relieve pose changes and develop robust feature representations from the global pictures and distinct local positions.

Another recent approach is PAR [19], proposed by Zhao et al. It uses a part-classifier to perform a “soft” partition on pedestrian photos by learning the aligned parts directly. PAR trains the part classifier in an unsupervised way using the attention technique. It consists of a baseline network and an expansion for calculating individual component loss. It is trained to reduce both part loss and global classification loss. Because there is just one key parameter to tweak, i.e., the number of produced parts K, PAR is also simple to repeat.

Chapter 3

Insight into our Dataset:

In this project we used the '*Market-1501*' dataset. This data set is preferred because other datasets:

1. are limited in scale.
2. consist of hand-drawn bounding boxes,
3. have only one ground truth and one query image for each identity.

About *Market-1501* dataset:

The Market-1501 dataset is currently best large-scale public benchmark dataset which is available for person ReID. The pictures used in this dataset are taken from the supermarket present in front of Tsinghua University, China, using a total of six cameras (five high resolution and one low resolution camera). The field-of-view overlaps among all different cameras. Overall, this dataset contains 32,668 annotated bboxes (bounding boxes) of 1,501 identities, with a distractor set containing over 500,000 images. We can say that almost each one of the six cameras tracked or captured images of a single entity/individual present on site. Also, for cross camera search, the annotated entities are present in at least two camera views.

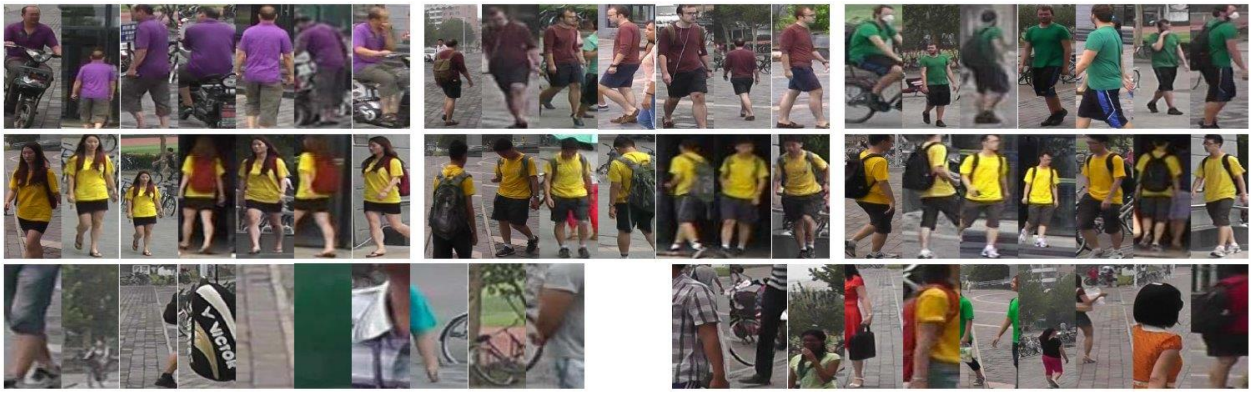

Figure 1: Market 1501 dataset preview

Some features of '*Market-1501*' dataset is:

- Bounding boxes (bboxes) were detected using DPM
- Not only true positive bboxes are there, but false detection results are also there.
- Since, the photos of the entities may be tracked/captured by different cameras, multiple ground truth for everyone is obtained during cross camera search.

Following table gives the comparison of different datasets and Market-1501:

Dataset	Market 1501	VIPeR	CHUK03	RaiD
No of individuals tracked	1,501	632	1,360	43
No of bboxes	32,668	1,264	13,164	6,920
No of Distractors	2,793 + 500,000	0	0	0
No of cameras used	6	2	2	4
DPM or Hand	DPM	Hand	DPM	Hand
Evaluation	mAP	CMC	CMC	CMC

Table1: Comparison of different dataset

Note: The Deformable Parts Model (DPM) recognizes objects with a mixture graphical model (Markov random fields) of deformable parts. The model consists of three major components: A coarse root filter defines a detection window that approximately covers an entire object. In Market-1501, DPM is used for creating bboxes, rest were hand-drawn.

Chapter 4

Software Requirement Specification:

- **PyTorch:** It is one of the preferred platforms for deep learning research. The framework is built to speed up the process between research prototyping and deployment. It is an open-source machine learning framework based on the Torch library. We preferred it over TensorFlow because PyTorch allows quicker prototyping than TensorFlow.

- **PyTorchLightning:** PyTorchLightning is a lightweight PyTorch wrapper for high-performance AI research that aims to abstract Deep Learning boilerplate while providing you full control and flexibility over your code. With Lightning, you scale your models not the boilerplate.

- **TorchVision:** TorchVision is a library for Computer Vision that goes hand in hand with PyTorch. It has utilities for efficient Image and Video transformations, some commonly used pre-trained models, and some datasets.

- **TorchText**: TorchText is a PyTorch package that contains different data processing methods as well as popular NLP datasets. According to the official PyTorch documentation, TorchText has 4 main functionalities: data, datasets, vocab, and utils. Data is mainly used to create custom dataset class, batching samples etc.

- **TensorBoard:** TensorBoard is a tool for providing the measurements and visualizations needed during the machine learning workflow. It enables tracking experiment metrics like loss and accuracy, visualizing the model graph, projecting embeddings to a lower dimensional space, and much more.

- **Test-tube:** Test tube is a python library to track and parallelize hyperparameter search for Deep Learning and ML experiments. It's framework agnostic and built on top of the python argparse API for ease of use.

- **Joblib:** Joblib is a set of tools to provide lightweight pipelining in Python. In particular: transparent disk-caching of functions and lazy re-evaluation (memorize pattern) easy simple parallel computing.

- **NumPy:** It is a library used for working with multidimensional arrays and matrices and supports many high-level mathematical functions. We have used NumPy to increase the dimension of images using expand_dims () function.

Chapter 5

5.1 Proposed Solution

Given a query image, conventional methods compute the feature distances between the query image and all the gallery images in order to find the ordered set of probable classes. However, when the size of the database is very large, these approaches fail to obtain a good performance due to appearance ambiguity across different camera views.

Our model attempts to tackle the problem of person re-identification by utilizing both the visual and spatial temporal information of the query image to narrow the gallery database.

We use a Part-based Convolutional Baseline (PCB) for discriminative and robust feature representation in this project. The PCB model employs part-level features for pedestrian image description as it offers fine-grained information extraction and has been verified as beneficial for person retrieval in very recent literature.

The model partitions the conv-layer uniformly for learning part-level features. Images are not partitioned explicitly by the model. PCB takes a whole image as the input and outputs a convolutional feature. The architecture of PCB is simple and concise with slight modifications on the RESNET-50 network. The convolutional descriptor has much higher discriminative ability than the commonly used fully connected descriptor.

The model consists of:

1) A ResNet-50 network

2) An average pooling layers

3) Six 1 × 1 kernel-sized convolutional layers, six fully connected classifiers implemented with Cross-Entropy loss function.

5.2 Model Workflow

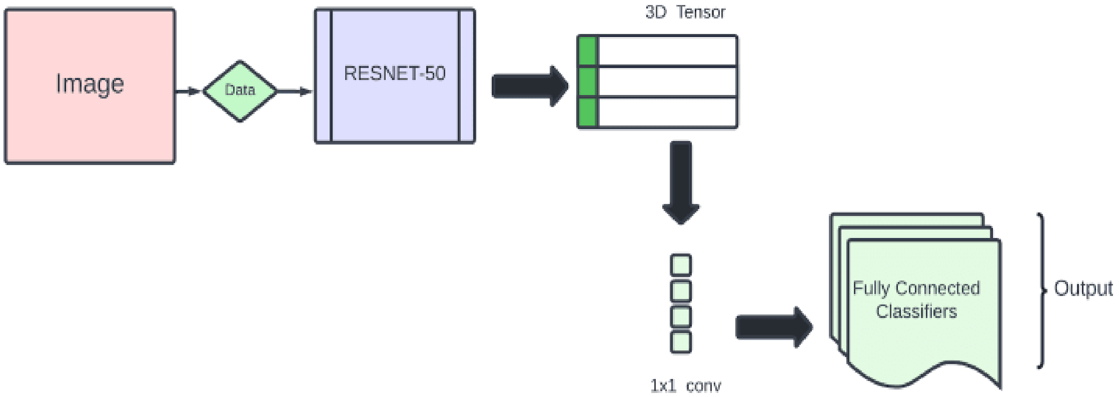

Figure 2: Our model's Architecture

5.3 Structure of the PCB Model

Backbone network

PCB can take any network without hidden fully connected layers designed for image classification as the backbone, e.g., Google Inception and ResNet. This paper mainly employs ResNet50 because of its concise architecture and high performance.

From backbone to PCB

We reshape the backbone network to PCB with slight modifications. The structure before the original global average pooling (GAP) layer is maintained the same as the backbone model. The difference is that the GAP layer and what follows are removed. When an image passes through all the layers on the backbone network, it becomes a 3D tensor T of activations. We define the vector of activations as a column vector.

Then, with a conventional average pooling, PCB partitions T into p horizontal stripes and averages all the column vectors in a same stripe into a single part-level column vector g. Afterwards, the model then deploys a convolutional layer to reduce the dimension of g. The dimension-reduced column vectors h is set to 256. Finally, each h is input into a classifier, which is implemented with a fully connected (FC) layer and a following Softmax function, to predict the identity (ID) of the input.

The input image goes forward through the stacked convolutional layers of the Resnet-50 network to form a 3D tensor. The model uses a conventional pooling layer, to spatially down-sample the 3D Tensor into pieces of column vectors. A following 1 × 1 kernel-sized convolutional layer reduces the dimension of the column vectors. Finally, each dimension-reduced column vector is input into a classifier, respectively. Each classifier is implemented with a fully connected layer using cross entropy loss function. During testing, these column vectors are concatenated to form the final descriptor of the input image.

During the training phase, each classifier is used to predict the class (person identity) of a given image. With the part-level feature representation learning scheme, PCB can learn local discriminative features and thus achieve competitive accuracy. During the test phase, six stripe-based features are concatenated into a column vector for the visual feature representation. We compute a similarity score according to the cosine distance between two feature vectors xi and xj .

$$s(xi\,,\,xj) = \frac{xi \cdot xj}{||xi|| ||xj||}$$

5.4 Extracting the Spatial Temporal component from the images

The filename of any image in this of the format: The X values represent variable numbers

$$XXXX\_CXSX\_XXXXXX\_XX$$

Here the first four characters represent the **label** of the number. The **camera number** is represented by the number after letter C and the **frame number** of the image is represented by the six numerical values after the camera number.

These values form the spatial-temporal value of the image and are used for prediction.

5.5 Implementation

We use a pretrained ResNet-50 as the backbone layer. This is followed by the “pool5” layer, after that we append a fully-connected layer followed by Batch Normalization and ReLU. The output dimension of the appended FC layer is set to 256-dim. We apply dropout on “pool5” layer as it avoids over-fitting and gains considerable improvement . We set the dropout ratio to 0.5.

The training images are augmented with horizontal flip and normalization. We set batch size to 64 and train the model for 34 epochs with base learning rate initialized at 0.1 and decayed to 0.01 after 40 epochs. The backbone model is pre-trained on ImageNet . The learning rate for all the pre-trained layers are set to 0.1× of the base learning rate. When employing refined part pooling for boosting, we append another 10 epochs with learning rate set to 0.01.

Chapter 6

Experiments and Result Analysis:

6.1 Result and Analysis

For determining the performance of our model, we have performed classification of all the images in our Market-1501 dataset and plotted the various evaluation metrics with respect to the number of epochs as shown in the below figures:

- **Training accuracy vs Epoch curve**

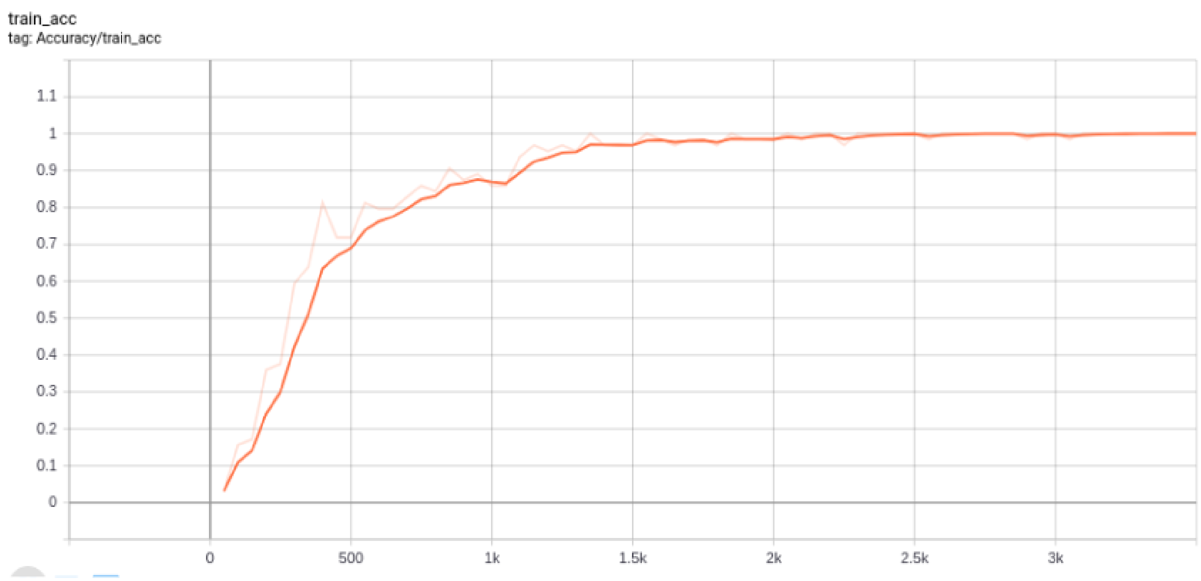

Fig 3: Training accuracy vs Epoch Curve

The above graph plots the variation (increase/decrease) in training accuracy with respect to the increase in number of epochs, where each epoch consists of 65 steps. And as clearly evident from the figure, the training accuracy gradually increases from null and reaches close to 100% at approximately the 34th epoch**.**

- **Loss vs Epoch curve**

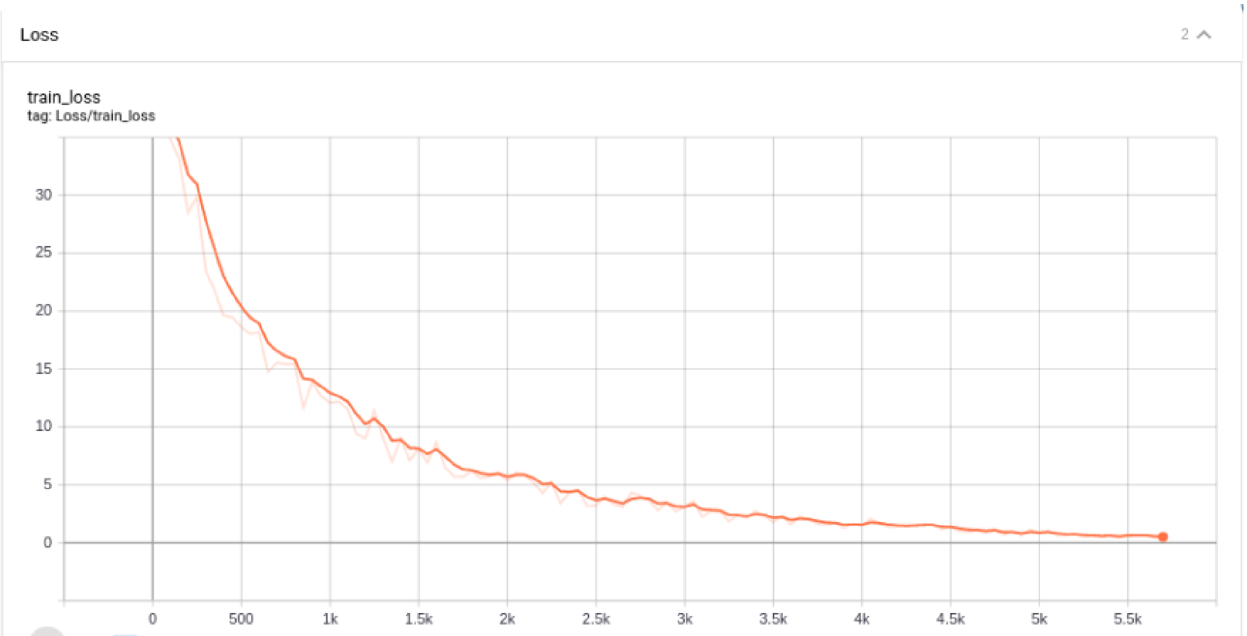

Fig 4: Loss vs Epoch Curve

The above graph plots the variation (increase/decrease) in the total loss of our model with respect to the increase in number of epochs, where each epoch consists of 65 steps. And as clearly evident from the figure, the loss gradually decreases from 100% and reaches close to 0 at approximately the 34th epoch**.**

- **CMC vs Epoch curve**

The **Cumulative matching characteristics** (CMC) curve is used to assess the accuracy of algorithms that produce an ordered list of possible matches. The final CMC score value is represented as Top-N accuracy form, and its value denoted the probability that the correct match lies in the top N position among all the matches returned for a particular query image.

For ex: we have a CMC curve with a 75% rank 10 accuracy. This suggests that 75% of the time, the correct match will be found in the top 10. The higher the rank N CMC-percentage, the better the algorithm (assuming the algorithms have been tested on the same dataset).

The graph as shown below, plots the variation (increase/decrease) in the CMC value of our model with respect to the increase in number of epochs, where each epoch consists of 65 steps. And as clearly evident from the figure, the value fluctuates between 0 and 0.007 for the whole training interval of 34 epochs**.**

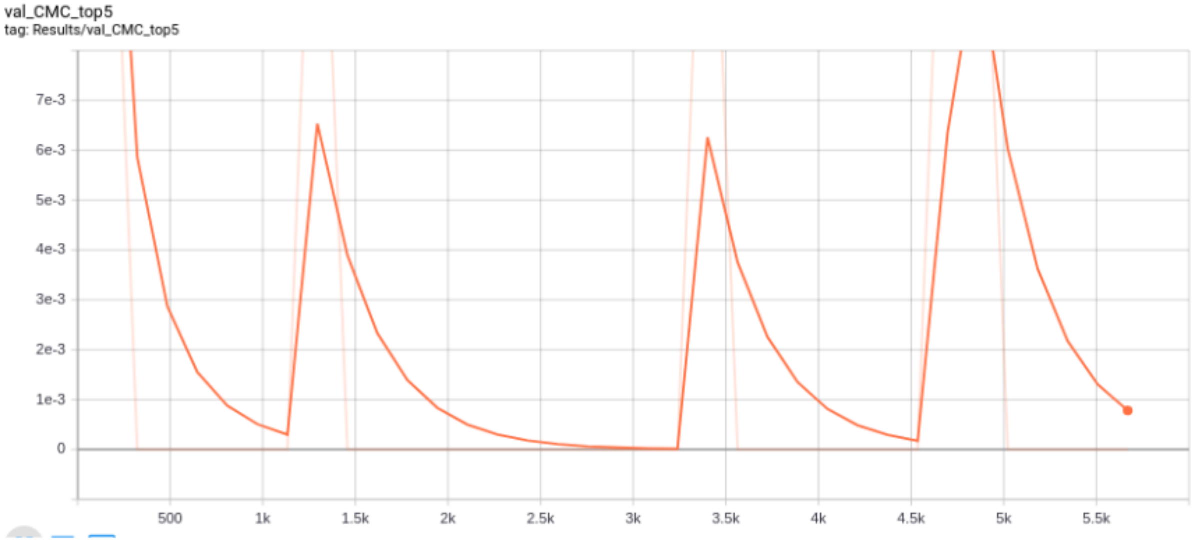

Fig 5: CMC vs Epoch Curve

- **mAP vs Epoch curve**

Mean Average Precision (mAP) is a metric in which the mean of average precision (AP) values are calculated over recall values ranging from 0 to 1. This measure is used to evaluate object identification algorithms.

The mAP is calculated by averaging the Average Precision (AP) of each class over a number of classes. The mathematical expression for calculating MAP is as shown below:

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{AP}_i$$

The mAP considers both false positives (FP) and false negatives (FN), as well as the trade-off between accuracy and recall (FN). Because of this, mAP is a good measure for most detection applications.

The graph as shown below. plots the variation (increase/decrease) in the mAP value of our model with respect to the increase in number of epochs, where each epoch consists of 65 steps. And as clearly evident from the figure, the value fluctuates between 0 and 0.0024 for the whole training interval of 34 epochs.

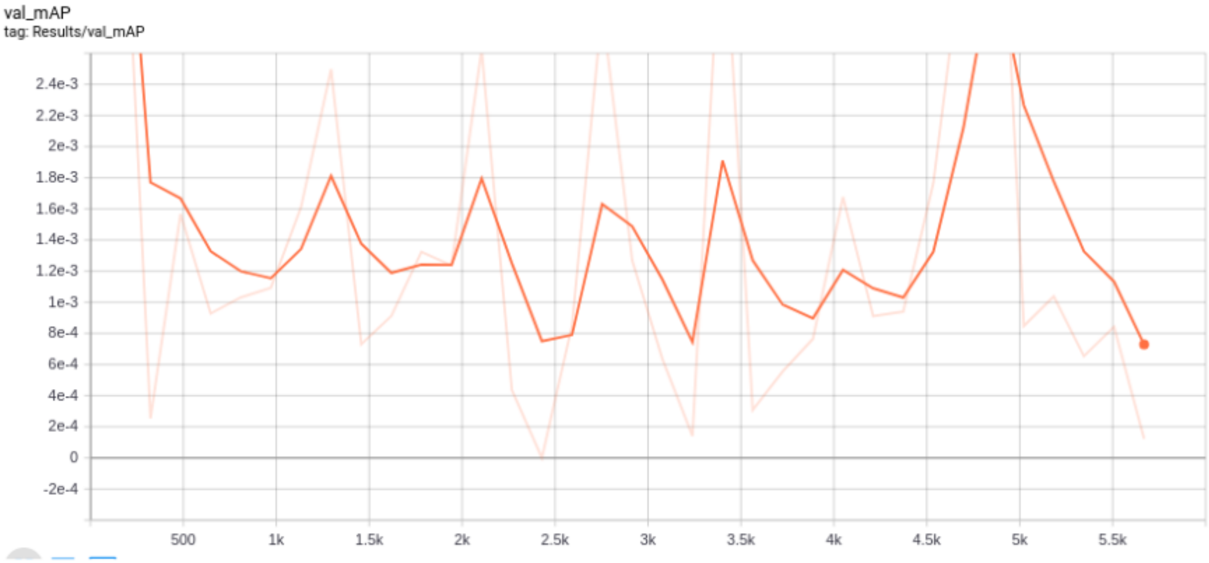

Fig 6: mAP vs Epoch Curve

6.2 Comparison with the existing solutions

As mentioned in literature review part, most of the existing solutions for the task of person re-id suffer from the appearance ambiguity problem because they only consider the visual component of a query image for finding its matches.

To mitigate the above-mentioned shortcoming, we designed a two-stream spatial temporal person ReID (st-ReID) architecture that mines both visual semantic similarity and spatial-temporal data. Our st-ReID approach surpassed all prior state-of-the-art methods by a considerable margin, achieving rank-1 accuracy of 98.1 percent on Market-1501. The below table shows a comparison of some previous models and our proposed model.

Name	CMC Top-1	CMC Top-5	mAP
SVDNet	82.3	92.3	62.1
Triplet Loss	84.9	94.2	69.1
PAR	81.0	92.0	63.4
MultiRegion	66.4	85.0	41.2
PDC	84.4	92.7	63.4
PCB	92.3	97.2	77.4
Resnet-50 + PCB (Our model)	95	98	98.9

Table2: Comparison of other models with ours

Chapter 7

Conclusion and Future Scope:

We were able to successfully perform person re-identification task by utilizing the spatial-temporal constraint along with the visual information of an image using our model. With the help of the Market-1501 database we were able to run our model with 500,000+ images which helped us reach closer to realistic configurations for person ReID. Thus, in this person ReID framework, we were able to bridge the gap between the spatial-temporal constraints and the visual semantics in the image. For increasing the accuracy of our model, we'll need a greater number of images.

In future we intend to increase the amount of data available for training in updated or new datasets, including them in the project to overcome the limitation of not being able to train a deeply learned network. We can also use a GAN for generation of a greater number of samples, including other data augmentation techniques.

Chapter 8

References:

1. Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, Shengjin Wang, “Beyond Part Models: Person Retrieval with Refined Part Pooling (and A Strong Convolutional Baseline)”, 2018, From: https://arxiv.org/pdf/1711.09349.pdf
2. A Person Reidentification Algorithm Based on Improved Siamese Network and Hard Sample. *Guangcai et. al.*
3. X. Yang, P. Zhou and M. Wang, "Person Re-identification via Structural Deep Metric Learning," in IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 10, pp. 2987-2998, Oct. 2019, doi: 10.1109/TNNLS.2018.2861991.
4. D. Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In ECCV, 2008
5. N. Gheissari, T. B. Sebastian, and R. Hartley. Person reidentification using spatiotemporal appearance. In CVPR, 2006
6. C. Engel, P. Baumgartner, M. Holzmann, and J. F. Nutzel. Person re-identification by support vector ranking. In BMVC, 2010
7. W. Zheng, S. Gong, and T. Xiang. Reidentification by relative distance comparison. TPAMI, 2013
8. A. J. Ma, P. C. Yuen, and J. Li. Domain transfer support vector ranking for person re-identification without target camera label information. In ICCV, 2013
9. S. Liao, Y. Hu, X. Zhu, and S. Z. Li. Person re-identification by local maximal occurrence representation and metric learning. In CVPR, 2015
10. D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino. Custom pictorial structures for re-identification. In BMVC, 2011
11. Zheng et al. 2016; Feng, Lai, and Xie 2018; Liang et al. 2018
12. Li et al. 2014; Chen, Guo, and Lai 2015
13. Ding et al. 2015; Wang, Lai, and Xie 2017; Hermans, Beyer, and Leibe 2017; Wang et al. 2016
14. Li et al. 2017; Zhao et al. 2017b; Su et al. 2017; Zhao et al. 2017a; Kalayeh et al. 2018; Song et al. 2018

15. Su, C.; Li, J.; Zhang, S.; Xing, J.; Gao, W.; and Tian, Q. 2017. Pose-driven deep convolutional model for person reidentification. In ICCV, 3960–3969

16. Identification. In CVPR, 1179–1188. Su, C.; Zhang, S.; Xing, J.; Gao, W.; and Tian, Q. 2016. Deep attributes driven multi-camera person re-identification. arXiv:1605.03259

17. H. Yao, S. Zhang, Y. Zhang, J. Li, and Q. Tian. Deep representation learning with part loss for person re-identification. arXiv preprint arXiv:1707.00798, 2017.

18. http://zheng-lab.cecs.anu.edu.au/Project/project_reid.html

19. https://lilianweng.github.io/posts/2017-12-15-object-recognition-part-2/

20. https://devblog.pytorchlightning.ai/why-should-i-use-pytorch-lightning-488760847b8b

21. https://ryanong.co.uk/2020/07/01/day-183-learning-pytorch-torchtext-introduction/