Pruning Methods for Person Re-identification: A Survey

- Aditya Tushar Wadnerkar

(Student at San Jose State University)

Original Paper : https://arxiv.org/abs/1907.02547

Introduction

- tremendous increase in deep learning architectures
- visual based recognition such as person re-identification
- computational complexity of CNNs hinders the deployment of Deep Siamese networks
- pruning can drastically reduce the complexities in network
- pruning reduces the number of FLOPS required by ResNet feature extractor

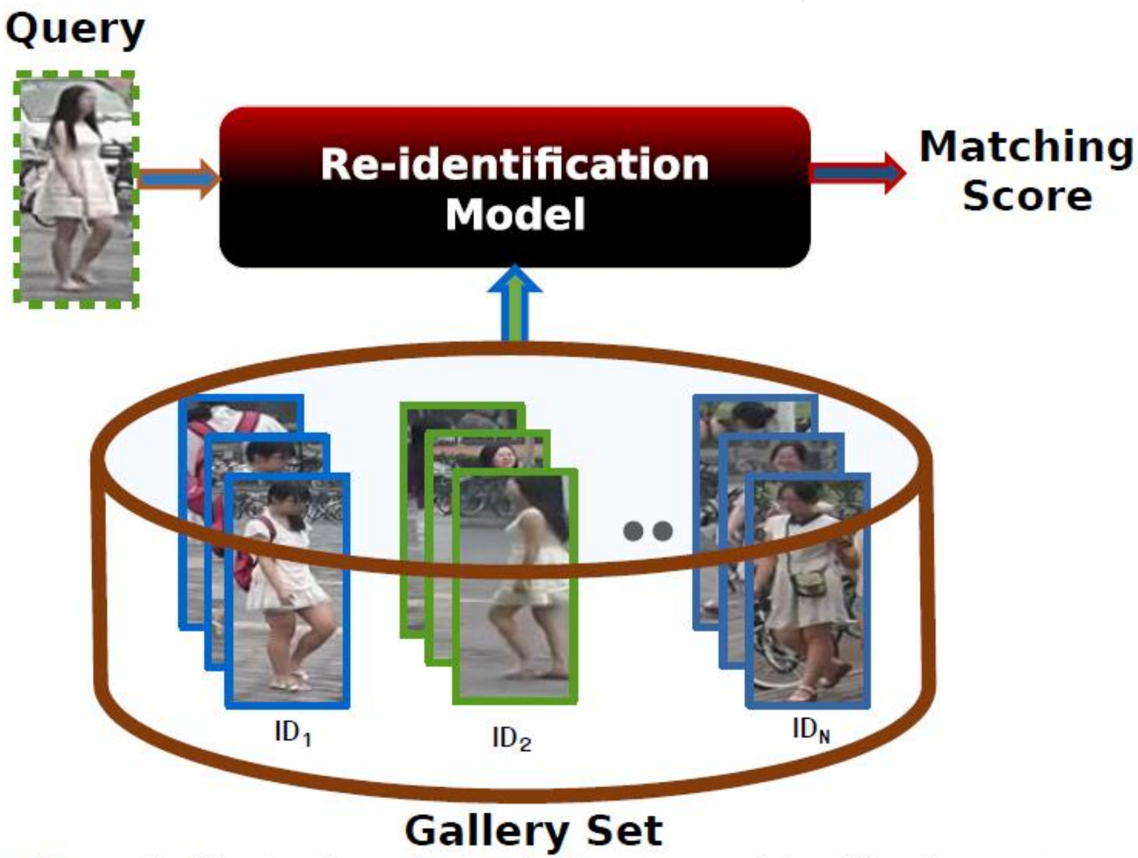

Figure 1: Illustration of a typical person re-identification system.

CNNs have achieved state of art accuracy at cost of high complexity

Siamese Networks

- used for biometric authentication where two sub networks share weights
- trained with labelled data extracting features from input images and performing pairwise matching

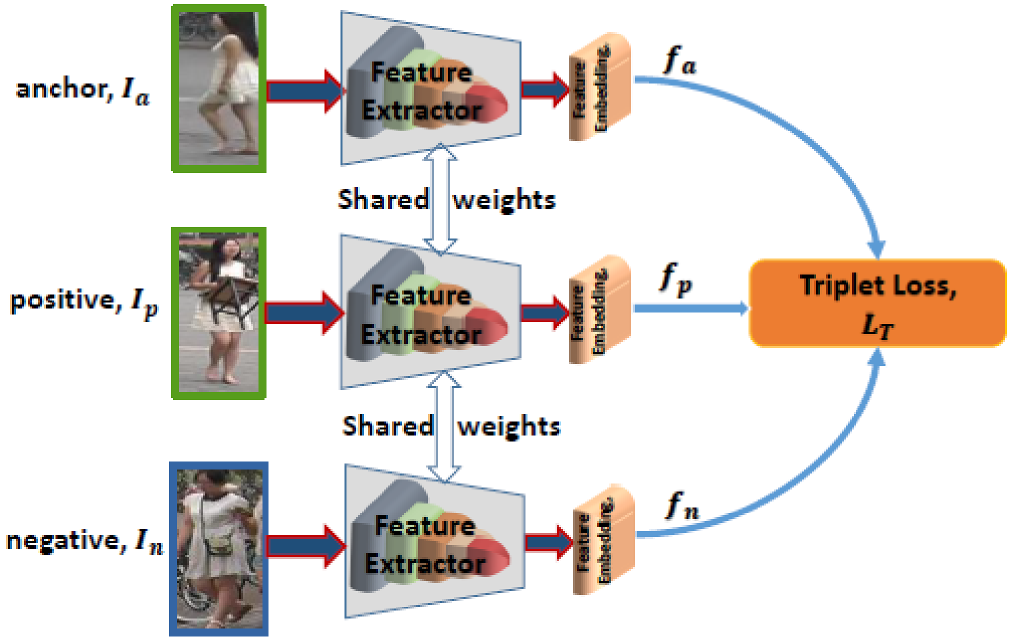

Figure 2: Triplet Training Architecture. Anchor and positive samples are same individual, whereas negative sample is different individual. These triplet set is fed through three identical networks. The triplet loss function optimizes the network parameters in such a way that minimizes the intra-class distances while maximizing the inter-class distance.

- VGG, Inception, ResNet and DenseNet can be used as feature extractor
- ResNet18 and ResNet34 being shallow CNNs provide lower re-identification accuracy

$$\mathcal{L}_{\mathrm{T}} = \frac{1}{N_T} \sum_{\substack{\mathrm{a,p,n} \\ y_a = y_p \neq y_n}} \left[m + d\left(\mathrm{f}_a, \mathrm{f}_p\right) - d\left(\mathrm{f}_a, \mathrm{f}_n\right) \right]_+$$

$$\mathcal{L}_{\mathrm{TBH}} = \frac{1}{N_s} \sum_{a=1}^{N_s} \left[m + \max_{y_p = y_a} d\left(\mathrm{f}_a, \mathrm{f}_p\right) - \min_{y_p \neq y_a} d\left(\mathrm{f}_a, \mathrm{f}_n\right) \right]_+$$

- we randomly sample a triplet {Ia, Ip, In}, where, (Ia, Ip) is a pair of images of the same individual, and (Ia, In) is that of different individual
- corresponding features from the backbone networks are fa, fp and fn
- to form batches by randomly sampling a person, and then sampling number of images of each person
- selects the hardest positive and the hardest negative samples within the batch
- forming the triplets for computing the loss

CNN Pruning Techniques

Table 1: A Taxonomy of techniques according to pruning strategyto reduce chanels.

Pruning Strategy	**Methods**
Prune Once	Hao Li[31] Redundant Channels[40] Entropy[32]
Iterative Pruning	Molchanov[30] Play and Prune[41] FPGM[42]
Pruning using regularization	Auto-Balance[43] Play and Prune[41]
Pruning by minimizing reconstruction error	ThiNet[44] Channel Pruning[33]
Progressive Pruning	PSFP[34]

Comparing Algorithms

PSFP pruning scheme

Algorithm 1 Algorithm Description of PSFP

Input: training data: $\mathbf{X}$
Input: pruning rate: P_i, pruning rate decay D
Input: the model with parameters $\mathbf{M} = \{\mathbf{M}^{(i)}, 0 \leq i \leq L\}$
  Initialize the model parameter $\mathbf{M}$
  for $epoch = 1; epoch \leq epoch_{max}; epoch++$ **do**
    Update the model paramters $\mathbf{M}$ based on X
    for $i = 1; i \leq L; i++$ **do**
      Calculate the l_2-norm for each channel
      Calculate the pruning rate P' at this epoch using P_i and D
      Select the N lowest l_2-norm depending on the pruning rate
      Zeroize the weights W of the selected channels
    end for
  end for
  Obtain the compact model with parameters **M'** from $\mathbf{M}$
Output: Compact model with parameters M'

Algorithm 2 Algorithm Description of FPGM

Input: training data: X
Input: pruning rate: P
Input: the model with parameters $\mathrm{M} = \{\mathrm{M}^{(i)}, 0 \leq i \leq L\}$
  Initialize the model parameter M
  for $epoch = 1; epoch \leq epoch_{max}; epoch++$ **do**
    Update the model parameters M based on X
    for $i = 1; i \leq L; i++$ **do**
      Select the $n_{out} \times P$ of W_i channels that satisfy Equation 22
      Zeroize the selected channels
    end for
  end for
  Obtain the compact model with parameters M' from M
Output: Compact model with parameters M'

FPGM pruning scheme

Table 3: Comparison of rank-1 accuracy and network complexity analysis in term of GFLOPS and Parameters taken from the literature.

Dataset	CIFAR10					
Feature Extractor	ResNet56					
Algorithm	Original			Pruned		
	R-1 (%)	GFLOPS	Parameters (M)	R-1 (%)	FLOPS (G)	Parameters (M)
Hao Li [31]	93.04	0.125	0.85	93.06	0.091	0.73
Auto-Balanced [43]	93.93	0.142	N/D	92.94	0.055	N/D
Redundant channel [40]	93.39	0.125	0.85	93.12	0.091	0.65
PP [41]	93.39	0.125	0.85	93.09	0.039	N/D
FPGM [42]	93.39	0.125	0.85	92.73	0.059	N/D

Dataset	ImageNet					
Feature Extractor	VGG16					
Algorithm	Original			Pruned		
	R-1 (%)	GFLOPS	Parameters (M)	R1 (%)	GFLOPS	Parameters (M)
ThiNet [44]	90.01	30.94	138.34	89.41	9.58	131.44
Molchanov [30]	89.30	30.96	N/D	87	11.5	N/D
HaoLi [31]	90.01	30.94	138.34	89.13	9.58	130.87
Channel Pruning [33]	90.01	30.94	138.34	88.1	7.03	131.44

Dataset	ImageNet					
Feature Extractor	ResNet50					
Algorithm	Original			Pruned		
	R-1 (%)	GFLOPS	Parameters (M)	R-1 (%)	GFLOPS	Parameters (M)
Entropy [32]	72.88	3.86	25.56	70.84	2.52	17.38
ThiNet [44]	75.30	7.72	25.56	72.03	3.41	138
FPGM [42]	75.30	7.72	25.56	74.83	3.58	N/D

Adaptive Filter Pruning

The Adaptive Filter Pruning (AFP) module and the Pruning Rate Controller (PRC). The goal of the AFP is to minimize the number of output channels in the model while the PRC tries to maximize the accuracy of the remaining set of output channels. This technique considers a model M can be partitioned into two set of important channels I and unimportant channels U.

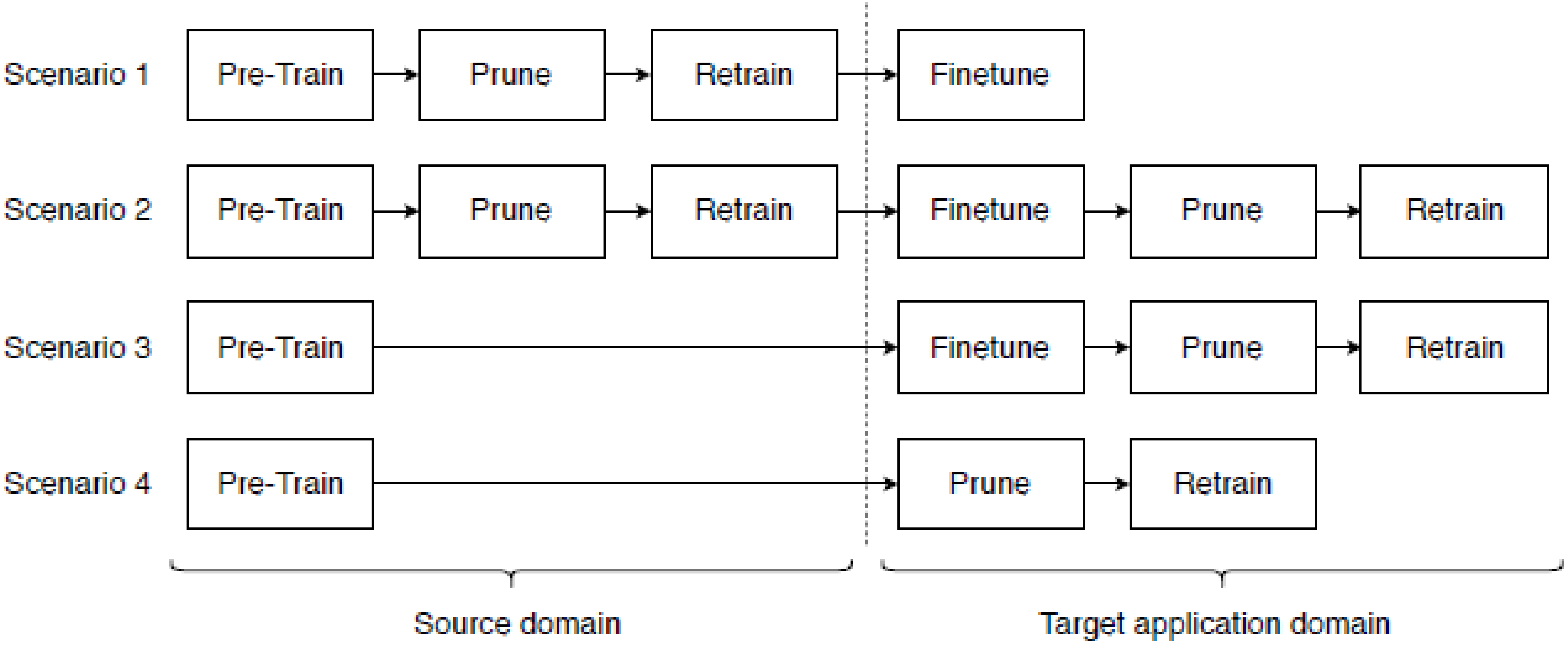

Figure 3: Scenarios for pruning and training a CNN.

Datasets

- ImageNet
- Market-1501
- CUHK03-NP
- DukeMTMC-reID

Performance Analysis

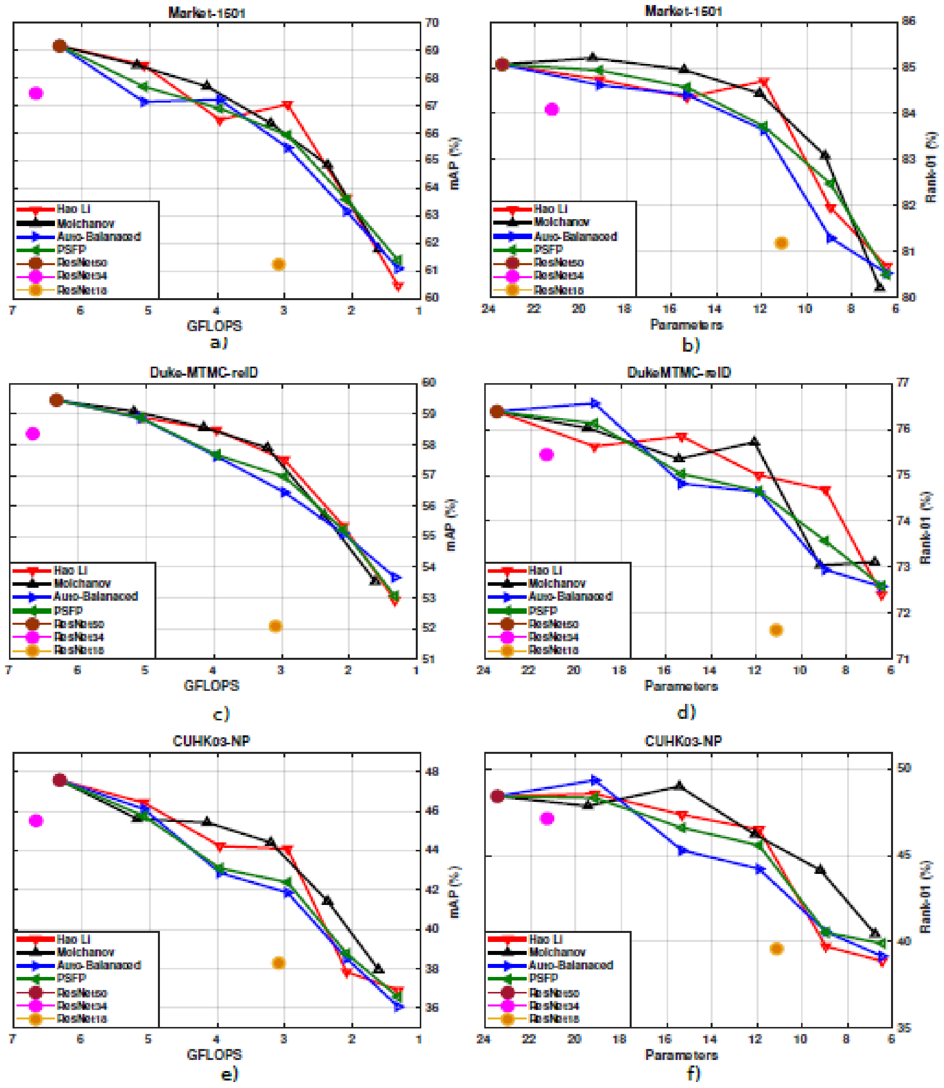

Table 5: Accuracy and complexity of baseline and pruning Siamese networks on ReID datasets. Mean average precision (mAP) and rank-01 accuracy (R-1) are shown in percentage (%).

Networks	Parameters	GFLOPS	Market-1501		DukeMTMC		CUHK03-NP	
			mAP	R-1	mAP	R-1	mAP	R-1
ResNet50	23.48	6.32	69.16	85.07	59.46	76.39	47.57	48.43
ResNet34	21.28	6.67	67.44	84.09	58.36	75.45	45.51	47.14
ResNet18	11.12	3.09	61.23	81.18	52.07	71.63	38.27	39.57
HaoLi	11.90	2.96	67.04	84.71	57.51	75.00	44.08	46.50
Molchanov	12.09	3.21	66.35	84.44	57.90	75.72	44.40	46.21
AutoBalanced	11.90	2.96	65.46	83.64	56.45	74.64	41.85	44.21
Entropy	11.90	2.96	65.16	82.39	56.64	74.64	42.44	44.07
PSFP	11.90	2.96	65.92	83.72	56.96	74.66	42.38	45.58

Conclusion

- discussion about different state-of-art pruning approaches suitable for compressing Siamese networks for person Re-identification
- pruning can considerably reduce network complexity (number of FLOPS and parameters) while maintaining a high level of accuracy
- pruning larger CNNs can also provide a significantly better performance than fine tuning the smaller ones
- both fine tuning and pruning should be performed in the same domain

THANK YOU