Person Re-Identification using Convolutional Neural Network

Basavaraja A S [1], Dr.Sridharmurthy S.K[2].

[1]P. G Scholar Department of Electronic and communication, UNIVERSITY B.D.T. COLLEGE OF ENGINEERING, DAVANGERE-577004, Karnataka, India.
[2]Professor, chairman, Department of Electronic and communication, UNIVERSITY B.D.T. COLLEGE OF ENGINEERING, DAVANGERE-577004, Karnataka, India.

---***---

Abstract: In proposed system person Re-id is done using non-overlap multiple camera observation system by applying deep CNN. Here features are extracted through jointly trained multi-phase forward and backward propagation. Method is executed in three phases, convolutional phase, pooling and fully connected. Most of the previous algorithms underperform by only emphasizing on designing hand-crafted features and metric method either consecutively or separately. Proposed method formulates different deep ranking framework that is able to handle both of these key components simultaneously to increase their result.

The model takes an RGB image as the input and compares with the trained features and results a similarity value which indicates whether two images is of the same person. To obtain better configuration the depth of the neural network is 15 weight layer and using 5*5 convolution kernels, zero padding and stride value is one. The CNN self learns directly from input image pixel and their similarity scores through joint representation.

Keywords: convolution neural network, Max_pooling, fully connected layers.

1. Introduction

Person Re-Id recognition is one among the fundamental problems in computer vision. Experiments are presented in the form of compounded variations in visual appearance across, Dissimilar camera views, Human positions, Illuminations, Background clutter, Occlusions, Relatively small resolution, the altered location of the cameras, Uniform dress, putting the border box, Tracking the image in videos and tracking people across cameras to searching for them in a large gallery from grouping photos as shown in figure1.

Person re-id is needed to measure the resemblance between two pedestrian images such that similarity score is high in case if it shows same identity otherwise low score for different identities.

In the model proposed we try to combine this into a single model which consists of a Convolutional Neural Network (CNN) encoder which helps in creating image encodings. We could have used some of the recent and advanced classification architectures but that would have increased the training time significantly.

For the most part CNN has three layers which includes convolution, pooling and fully connected layers.in order to build this model these layer are required. The network itself adjusts the weight of neurons based on the losses obtained during the train of system. In order to obtain the better classification of the image the hidden layer networks should be more than 10 layers.

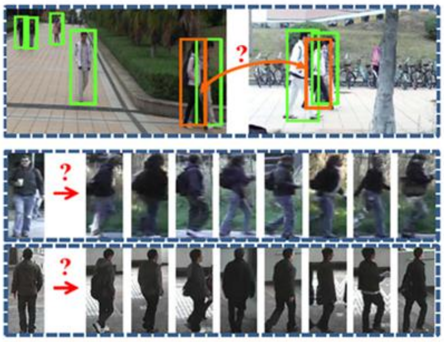


Figure 1: person re-id with some challenges of occlusion, angle, resolution and background with the data set viper and cuhk01.

This entire problem faced in all proposed systems, are removed by the using number of algorithms, by training the models in depth. In order to obtain the better result uses the CNN model

2. Objectives

1. In proposed system Person Re-ID is rank based framework, here we compare the similarity between the pair of a dull images with via patch matching and joint learning.
2. The best method to do the rank based frame work algorithm is with the CNN model. Ranking is obtained

within the gallery set; to obtain the better ranking model should train in well manner.
3. Better analysis of experimental results and code implementation, evaluation of proposed system implemented is done.
4. In training of CNN model involutes both forward and backward propagation in order to reduce the loss and self-evaluation model.

3. Problem Statements

Person Re-Identification is found by various methods but common challenges is observed are the Tests are presented in the form of compounded variations in visual appearance across Different camera views , Occlusions ,Human poses, Illuminations ,Background clutter, , Relatively low resolution, The different placement of the cameras ,Uniform clothing ,Putting the boundary box, Tracking the image in videos and tracking people across cameras to searching for them in a large gallery, from grouping photos.

Person re-id using CNN method, has three layers architecture which includes convolution, pooling and fully connected layers.in order to build this model these layer are required. Train the dataset captured form different cameras with resolution and poses using this proposed system. After this given the query images. The network has an ability to find the person.

4. Methodology

The technology involved here first preprocessing of input images found in dataset to a fixed size then extracting the features from those preprocessed images later these features are faded to the neural network and this model is used to generate automated learning any given input image. Further explanation about methodology is explained in the further chapters.

5. Literature Review

[1]. The proposed method exploits solve the person re-id problem by using salience matching strategy. In this method patch matching is adopted and patch salience matching is estimated. Person is identified by the minimizing the human salience and reliable feature information from the two different cameras views. This method is get best results on the VIPeR dataset and CUHK dataset.

[2].The method solving the person re-id problem extracting feature and learning discriminative data followed by matching the temples by distance formula. This method involves identification based on the ranking. It basically gives the ranking based on matching. If 1st rank matching more which indicates the accuracy of the system is good. Predication result well get more accurate. Here basically used SVM algorithms. This method his more scalable and high performance method.

[3].This method basically involves the “extracting features using CNN method from input images and comparing those features across images by using metric methods. This basically computes cross-input neighborhood differences, which extracting local relationships between the 2 input images from each input image”. Here significantly performs the state of the art on data set and a medium-sized data set and is immune to overfitting.

[4].In this paper, propose a jointly handle occlusions and background clutter, misalignment, photometric and geometric transforms completely by unique filter pairing neural network (FPNN). All this are finding by the jointly of adjacent pairs. To perform this it uses the photometric transform. Here it uses the CNN system in order to perform the mix of complex photometric and geometric transforms to compare the input image with the trained network.

[5]. This paper proposes extract the important feature of the body by applying the correlation between input images from the 2 different cameras of the same resolution or different. Difference between them is expressed in Histograms Gradients (HOG) and this comparison all expressed in the CMC curve.

[6]. Approach solved under a transfer learning framework by the insight that different visual metrics should be optimally learned for various candidate sets. Given an outsized training set, the training samples are selected and reweighted consistent with their visual similarities with the query sample and its candidate set. A weighted metric is online learned and transferred from a generic metric to a candidate set specific metric. The re-id performance is presented using the CMC curve.

[7].In this paper, present person re-id on the cross dataset by a deep learning framework supported convolutional neural networks to extract hand-crafted features from the probe image using cosine metric which is used to calculate the similarity. This method also evaluate better the performances on most of the dataset.

[8].In this paper, we propose a robust feature extraction model named Discriminative Local Features of Overlapping Stripes A weighted metric is

online learned and transferred from a generic metric to a candidate set specific metric. The re-id performance is presented using the CMC curve.

[9]. Person re-id in the video frames are detects based on the end to end comprehensive baseline which includes the number of detectors and recognizers and this method uses the 6 near-synchronized cameras. This contains number of raw frames and identities which are sounded position box and identity. Finally classification of image is obtained from the metric confidence weighted similarity.

[10].one of the best approaches to person re-id is the covariance descriptor. In this methods color and important features of a person by using the Gaussian distribution of pixel of the images, here basically perform the mean and covariance of the image patches and edges of the images. Features are extracted by performing the number of iteration of Gaussian distribution and finally done the normalization to better classification of the images.

[11].in this model re-id based on the Hilbert space with zero mean of Gaussian distribution with algorithm of Keep It Simple and Straightforward Metric which basically perform on the Cross-view Discriminant Analysis

6. Literature Review Summary

Person Re-Identification is found by various methods , but common challenges is observed are the Challenges are presented in the form of compounded variations in visual appearance across, Different camera views, Human poses, Illuminations, Background clutter, Occlusions, Relatively low resolution, The different placement of the cameras, Uniform clothing, Putting the boundary box, Tracking the image in videos and tracking people across cameras to searching for them in a large gallery, from grouping photos.

This entire problem faced in all proposed systems, are removed by the using number of algorithms, by training the models in depth. In order to obtain the better result uses the CNN model. This method is similar to the artificial neural networks. This model train the networks by itself by method of forwarded propagation and back propagation methods.

For the most part CNN has three layers which includes convolution, pooling and fully connected layers.in order to build this model these layer are required. The network itself adjusts the weight of neurons based on the losses obtained during the train of system. In order to obtain the better classification of the image the hidden layer networks should be more than 10 layers.

7. Design

For the most part CNN has three layers which includes convolution, pooling and fully connected layers.in order to build this model these layer are required.

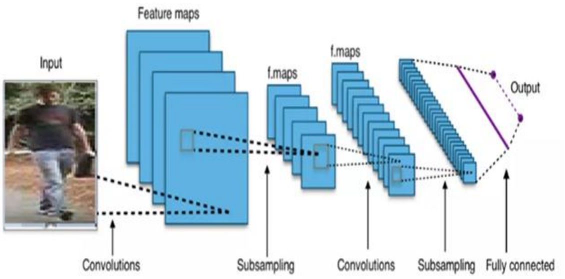


Figure 2: proposed CNN system

The basic functionality of the instance CNN is classified into four fundamental sectors

1. The image which consists of pixels values these values will extracted by input layer if we take any additional kind of CNN.
2. Neurons output are governed by the convolution layers. These are joined to input common domain. In order to detect output of these neurons it carries out inner outcome in the middle weights. This layer also contains one more important nonlinearities function called ReLu. This function is mainly used to normalization convolution layer output.
3. Given input decimation and proportionality of spatial are done by the pooling layer added.in with in a period of that awakening it lowers the parameter quantity. Which basically perform the down sample
4. As set up in classification of neural network, the entirely joined layers carry out. The task there by makes ideas in order to generate outcome of class among awakening. This can be utilized in classification. In order to enhance execution we can utilize ReLu in the middle of these entirely joined with decimation methods are used by CNNs to classification by giving of class outcome along with regression is also done here. thus the layer next to layer earliest input transformation occur in CNNs

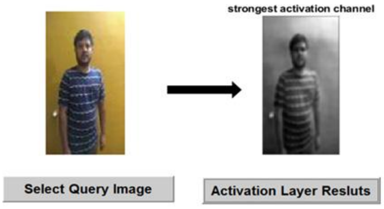
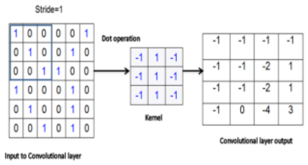


Figure 3: Activation Layer Results

The design of this model along with development takes much time. Here we look over into the discrete layers by presenting excitable together with connectivity.

Convolutional layer

In order to handle CNN the convolution layer takes major part in the system. In this layer training network will start to extract the important feature from the input dataset. In order to get the better data we have to define the kernel size first and number of the kernels. Kernel size should be small in order to train the model better. In our proposed model we took the 3*3 kernel. 60 number of kernel. Padding also we have to declare along with the number of stride to convolute kernel on the input image as show in figure 4.



1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

-1	1	-1
-1	1	-1
-1	1	-1

-1	-1	-1	-1
-1	-1	-2	1
-1	-1	-2	1
-1	0	-4	3

Figure 4: convolution layer results

The utilization of memorable core is targeted by the framework of layers. The geometric measurement of core normally tiny in nature, nevertheless they have the capability to escalated the whole of the input till at the bottom.to provide a two dimensional activation map, the layer trust around individual over geometric measurement of the argument. This happens when info strike the layers.

This is first layer which is used to declare the image size; in our proposed model size of the image is 128*64*3. This numbers are corresponding to the height, width and size of the image channel. If Channel size is 1 it indicates the dataset are proceed in gray scale format. If Channel size is 3 it indicates the dataset are proceed in GRB format. All this data set is get shuffle automatically during train of the networks.

This layer significantly reduces the difficulty in the system via optimization of result. Those were optimized via 3 excited variables they are depth, stride and zero padding. Thus lowering this extreme variable will sufficiently minimize entire neurons of the system, and also sufficiently lower the design identification capacity of the system.

In order to get better result the stride value should be One, but it will take more time to extract the feature and it will take more speed processors and storage space. Depth of analysis on image is many depending on this part only.

Introducing Non-Linearity function

One of the most important operations is implemented after every convolution operation, for this operation most frequently used nonlinear function is the ReLu. Is an element wise operation which replaces zero value for the negative pixels of the image. This function definition is as shown in the figure 5 and results of the ReLu function are shown in the figure 6.

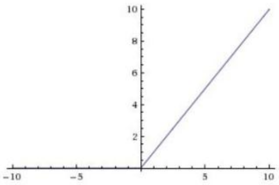


Figure 5: Rectified linear unit activation function

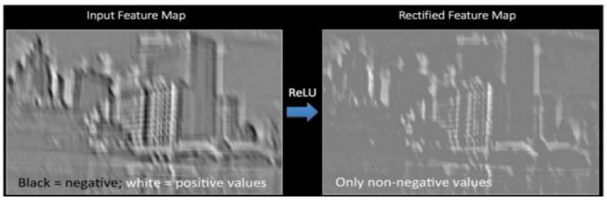
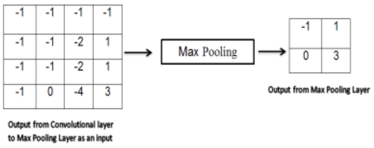


Figure 6: Output after a ReLu operation.

Pooling layer

The polling layer important goals are slightly decrease the geometric structure representation which will lower the quantity of the framework and these estimate the problems related to system.


Max Pooling
Output from Convolutional layer to Max Pooling Layer as an input
Output from Max Pooling Layer

-1	-1	-1	-1
-1	-1	-2	1
-1	-1	-2	1
-1	0	-4	3

-1	1
0	3

Figure 7: output of max pooling layer.

It works on the every refreshing element with in the code and its geometric structure by utilizing the function which is called as MAX. They are available in shape of maximum pooling layers in the heart spatial structure of two cross two along with footsteps of two beside geographical structure of the 25% initial proportions. While the depth capacity should be keep in the quality portions.

The calamitous characters of these layers the 2 normally perceived routines of max_pooling which are customarily strides and filters of these layers. This pair is adjusted to two cross two.

This will permit the layers to increase via everything of geometrical structure of the code. That will overlay pooling that utilized the point of stride about the core magnitude place. We have thanks the calamitous identity of the core magnitude which is in excess of three that will be generally largely of declined the execution of the system.

The main principle after the maximum polling CNN building might hold common pooling. The common pooling layers are contained the pooling neurons cell which are ready to discharge a mass of ferment actions which counting mean pooling and standardization. This class will primarily specify in the utilization of max pooling.

This layer is mainly used to reduce the number features output from the convolution layers i.e. this layer act as the down sampling layer. This rate is mainly depending upon the value on the stride. If stride value is less, will get better data to train the model and get large number of features from the dataset. In proposed system stride value is 2.

Fully connected layer

The neurons weight in proceeding layer is connected to each neuron in subsequent fully connected layer. This mean all the results from the pooling layers are get combines. By this we can classify the input data.in proposed system we use 13 fully Connected Layer.

This layer carry the cells that are attached face to face with another of same neuron within the 2 adjoining layers, not joined to whichever layers in them. This is correlative to that which is presented in orthodox notch.

One more layer is used at the output FC layer. It's normally called as activation function is mainly used to normalize the output. The result of this function should be positive number there sum is near to 1.

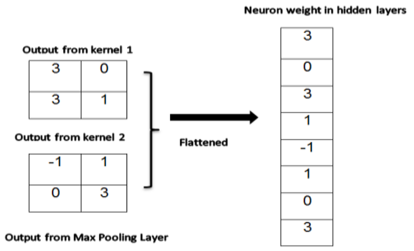


Figure 8: flatten the result from pooling layers

8. Flow chart of CNN system

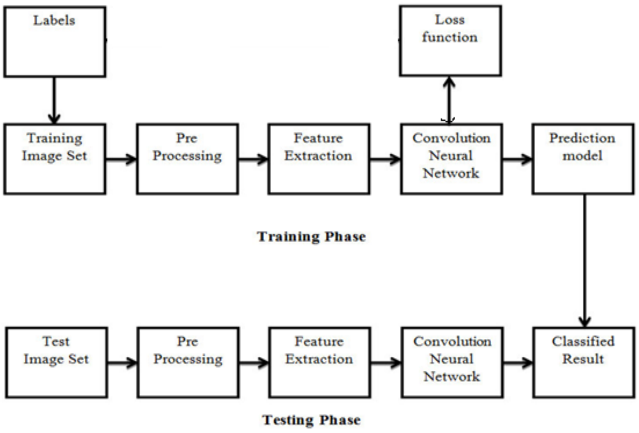


Figure 9: flow chart of proposed system for Person Re-Id using CNN.

Data set contains images of different persons captured with cameras of different resolution and views. These captured images are labeled as Person1, Person2…and so on. Some of these images will act as training set and rest of them as testing set.

After the labeling, images are to be preprocessed for image resizing and normalization. Then vital features will be extracted to train the model. The neural networks extract the features by performing the convolution and pooling on the data set.

During model training, if any important features are lost model should be retrained by back propagating to extract the lost features again. After reduction of the losses the trained network is ready for classification of the input data. In the testing phase, again the images are to be preprocessed and normalized. Then vital features will be extracted.

Then compared with the trained model. If any features are matched with the trained model then it will classify the images.

Algorithm for CNN based Classification

1. Perform the convolution operation in the first layer.

2. The activation layer controls the transfer of signals from one layer to a different layer.
3. Fasten the training period by using rectified linear measure (ReLu).
4. To extract the important feature performs pooling i.e. down sampling.
5. The neurons weight in proceeding layer is connected to each neuron in subsequent fully connected layer.
6. Feedback is given to the neural network by feeding the Loss weight.

9. Implementation

9.1Datasets

13 people's datasets are used to evaluate the model these are similar to the standard dataset like VIPeR, Market-1501, GRID, CUHK01, CUHK03 Example images of each of these datasets are shown in Fig. 10.

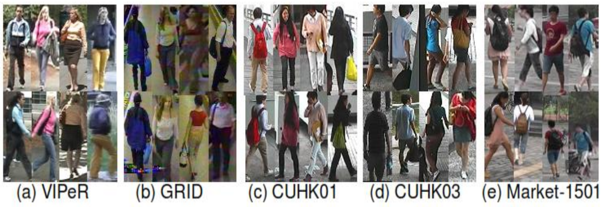

(a) VIPeR (b) GRID (c) CUHK01 (d) CUHK03 (e) Market-1501

Figure 10: Different data set

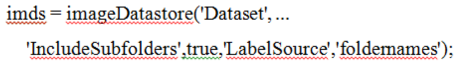
All this data set contains the number of images with different view of angle and this data sets have many challenges, those are occlusion, illumination, pose, viewpoint and background clutter.

The proposed method contains 13 persons dataset , which contains around 192 photos, in which 70% of the data set are used to train the model. Reaming 30% of data are used to testing the model.

During training of the model the fixed size image about 128*64*3 is used. After taking the photos from different cameras they are to be converted into one standard size in order to reduce the error generation. After the training, to check the model, one query image is selected which should also of the same size i.e. 128*64*3.

9.2 Code and Result

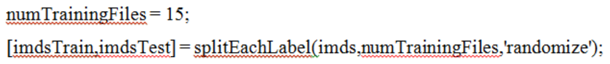
First Load the dataset to train the Network of different persons. This should be stored in the folder name as dataset in current folder.



```
imds = imageDatastore('Dataset', ...
   'IncludeSubfolders',true,'LabelSource','foldernames');
```

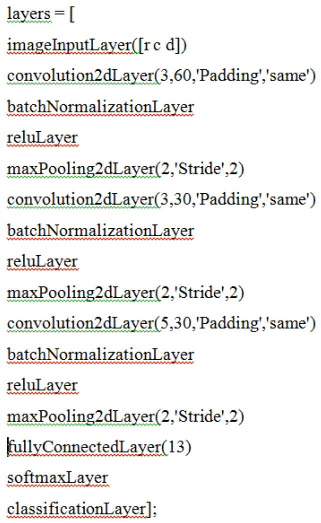
The dataset has 13 persons each of which contains around 20 images per persons. The Each image is 128*64*3 pixels.

9.2.2 Divide the dataset for training and testing.



```
numTrainingFiles = 15;
[imdsTrain,imdsTest] = splitEachLabel(imds,numTrainingFiles,'randomize');
```

9.2.3 The CNN architecture



```
layers = [
imageInputLayer([r c d])
convolution2dLayer(3,60,'Padding','same')
batchNormalizationLayer
reluLayer
maxPooling2dLayer(2,'Stride',2)
convolution2dLayer(3,30,'Padding','same')
batchNormalizationLayer
reluLayer
maxPooling2dLayer(2,'Stride',2)
convolution2dLayer(5,30,'Padding','same')
batchNormalizationLayer
reluLayer
maxPooling2dLayer(2,'Stride',2)
fullyConnectedLayer(13)
softmaxLayer
classificationLayer];
```

Image Input layer

This is first layer which is used to declare the image size; in our proposed model size of the image is 128*64*3. This numbers are corresponding to the height, width and size of the image channel. If Channel size is 1 it indicates the dataset are proceed in gray scale format. If Channel size is 3 it indicates the dataset are proceed in GRB format. All this data set is get shuffle automatically during train of the networks.

Convolution layer

In this layer training network will start to extract the important feature from the input dataset. In order to get the better data we have to define the kernel size first and number of the kernels. Kernel size should be small in order to train the model better. In our proposed model we took the 3*3 kernel. 60 number of kernel. Padding also we have to declare along with the number of stride to convolutes kernel on the input image.

Normalization Layer

This layer mainly used in between the convolutional layers in order to normalize the activations and gradients of the extracted features in the convolution layers. Normalization function is the nonlinearities. In proposed system we use the ReLu function.

Pooling Layer

This layer is mainly used to reduce the number features output from the convolution layers i.e this layer act as the down sampling layer. This rate is mainly depending upon the value on the stride. If stride value is less, will get better data to train the model and get large number of features from the dataset. In proposed system stride value is 2.

Fully connected layer

The neurons weight in proceeding layer is connected to each neuron in subsequent fully connected layer. This mean all the results from the pooling layers are get combines. By this we can classify the input data.in propsed system we use 13 fully Connected Layer.

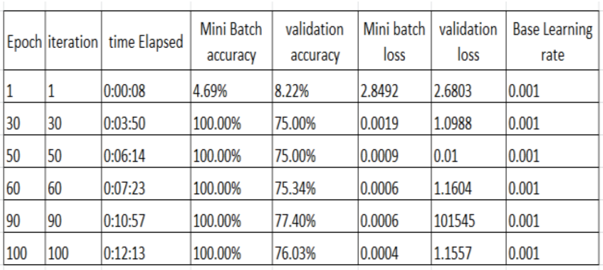
Soft_max layer

This layer is used at the output FC layer. It's normally called as activation function is mainly used to normalize the output. The result of the function should be positive number there sum is near to 1.

Classification Layer

This is the last layer which is mainly used for classification of input images by comparing with features that are extracted during the training of the model. In order to perform this classification one activation function is used i.e. softmax.

Options to default settings for the train the model with predefined values such as some SGDM, with initial learning rate of 0.01., epochs rate of 100, and start the training with a validation frequency about 30. This setting can be changed as per the requirement in order to improve our proposed model.

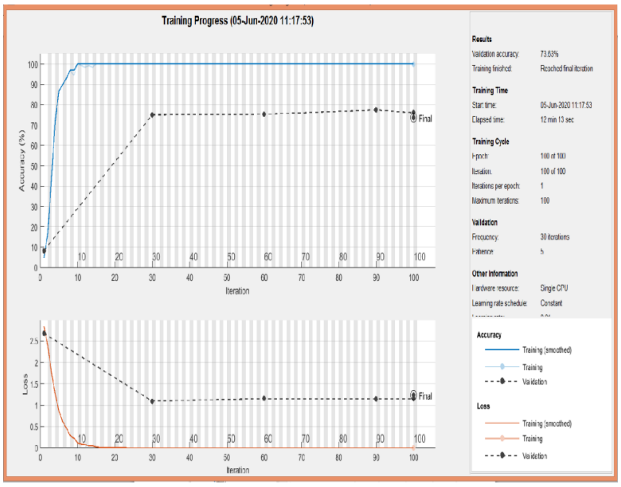


Epoch	iteration	time Elapsed	Mini Batch accuracy	validation accuracy	Mini batch loss	validation loss	Base Learning rate
1	1	0:00:08	4.69%	8.22%	2.8492	2.6803	0.001
30	30	0:03:50	100.00%	75.00%	0.0019	1.0988	0.001
50	50	0:06:14	100.00%	75.00%	0.0009	0.01	0.001
60	60	0:07:23	100.00%	75.34%	0.0006	1.1604	0.001
90	90	0:10:57	100.00%	77.40%	0.0006	101545	0.001
100	100	0:12:13	100.00%	76.03%	0.0004	1.1557	0.001

Table 1: Result of trained CNN model with validation accuracy

As shown in the table as in the begin of the iteration the accuracy and validation are very less, so the validation loss are more. As so as the number Epoch and iteration are increases accuracy is get increased and finally validation loss is get reduced.

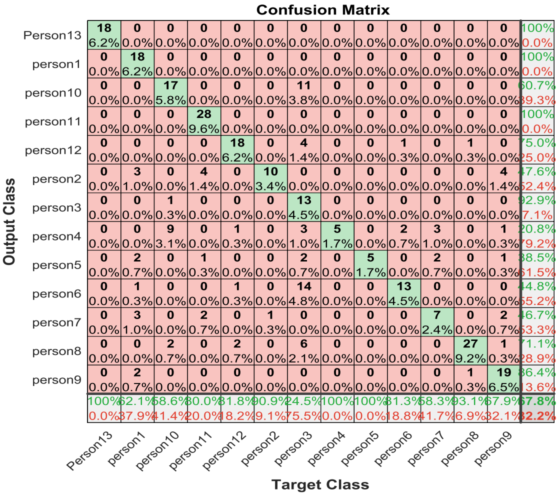
The measure problem in the CNN model to train well it requires the more data and more weighted layer. As layers are increased it takes more time to train the networks and requires huge amount process memory



Results
Validation accuracy: 73.63%
Training finished: Reached final iteration
Training Time
Start time: 05-Jun-2020 11:17:53
Elapsed time: 12 min 13 sec
Training Cycle
Epoch: 100 of 100
Iteration: 100 of 100
Iterations per epoch: 1
Maximum iterations: 100
Validation
Frequency: 30 iterations
Patience: 5
Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant

Figure 11: Result of trained CNN model with validation accuracy

The above result show the output of trained network, the validation accuracy can be improved by increasing the number of iteration and epoch's rate.



	Person13	person1	person10	person11	person12	person2	person3	person4	person5	person6	person7	person8	person9	
Person13	18 6.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
person1	0 0.0%	18 6.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
person10	0 0.0%	0 0.0%	17 5.8%	0 0.0%	0 0.0%	0 0.0%	11 3.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	60.7% 39.3%
person11	0 0.0%	0 0.0%	0 0.0%	28 9.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
person12	0 0.0%	0 0.0%	0 0.0%	0 0.0%	18 6.2%	0 0.0%	4 1.4%	0 0.0%	0 0.0%	1 0.3%	0 0.0%	1 0.3%	0 0.0%	75.0% 25.0%
person2	0 0.0%	3 1.0%	0 0.0%	4 1.4%	0 0.0%	10 3.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 1.4%	47.6% 52.4%
person3	0 0.0%	0 0.0%	1 0.3%	0 0.0%	0 0.0%	0 0.0%	13 4.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	92.9% 7.1%
person4	0 0.0%	0 0.0%	9 3.1%	0 0.0%	1 0.3%	0 0.0%	3 1.0%	5 1.7%	0 0.0%	2 0.7%	3 1.0%	0 0.0%	1 0.3%	20.8% 79.2%
person5	0 0.0%	2 0.7%	0 0.0%	1 0.3%	0 0.0%	0 0.0%	2 0.7%	0 0.0%	5 1.7%	0 0.0%	2 0.7%	0 0.0%	1 0.3%	38.5% 61.5%
person6	0 0.0%	1 0.3%	0 0.0%	0 0.0%	1 0.3%	0 0.0%	14 4.8%	0 0.0%	0 0.0%	13 4.5%	0 0.0%	0 0.0%	0 0.0%	44.8% 55.2%
person7	0 0.0%	3 1.0%	0 0.0%	2 0.7%	0 0.0%	1 0.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	7 2.4%	0 0.0%	2 0.7%	46.7% 53.3%
person8	0 0.0%	0 0.0%	2 0.7%	0 0.0%	2 0.7%	0 0.0%	6 2.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	27 9.2%	1 0.3%	71.1% 28.9%
person9	0 0.0%	2 0.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.3%	19 6.5%	86.4% 13.6%
	100% 0.0%	62.1% 37.9%	58.6% 41.4%	80.0% 20.0%	81.8% 18.2%	90.9% 9.1%	24.5% 75.5%	100% 0.0%	100% 0.0%	81.3% 18.8%	58.3% 41.7%	93.1% 6.9%	67.9% 32.1%	67.8% 32.2%

Figure 12: Result of trained CNN model with confusion Matrix.

Conclusion

The proposed method is based on ranking the model i.e. state-of-the-art to identify similarity metric which leads to match a search by making up the losses. It creates a relation between image pairs and features thereby integrating them into a single deep ranking framework which computes similarity through joint representation.

Future Enhancement

In the future enhancement to form different ways to form larger-scale outside data for network learning. Additionally to adapt the approach to video data, i.e., the way to measure the similarity of two sequences of detected pedestrian images. Can be also used:

1. Surveillance, Forensic and Biometrics (Airport, Stations)
2. Retail, profiling, customer understating.
3. Re-acquire people in camera networks.
4. Recognize people in biometric systems.
5. Automatic profiling of customers/visitors/tourists.
6. Re-id is helpful for tracking systems.

References

[1]. Lin Wu,Chunhua Shen,Anton van den Hengel, Person Net: Person Re-identification with convolution neural network

[2] .B. Prosser, W.-S. Zheng, S. Gong, and T. Xiang, “Person Re-Identification by support vector ranking,” in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Aberystwyth, U.K., Aug. 2010, pp. 21.1–21.11.

[3] .E. Ahmed, M. Jones, and K. T. Marks, ‘‘An improved deep learning architecture for person re-identification,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, Jun. 2015, pp. 3908–3916.

[4]. W. Li, R. Zhao, T. Xiao, and X. Wang, ‘‘DeepReID: Deep filter pairing neural network for person re-identification,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, Jun. 2014, pp. 152–159.

[5]. Slawomir Bak, Etienne Corvee, François Bremond, Monique Thonnat, Person Re-identification Using Spatial Covariance Regions of Human Body Parts. HAL Id: inria-00496116 Submitted on 29 Jun 2010.

[6]. Wei Li, Rui Zhao, Xiaogang Wang, Human Reidentification with Transferred Metric Learning 2012-ACCV.

[7]. Yang Hu, Dong Yi, Shengcai Liao, Zhen Lei, Stan Z. Li ,Cross Dataset Person Re-identification 2014-ACCV.

[8]. Gheissari, N., Sebastian, T.B., Rittscher, J., Hartley, R.: Person reidentification using spatiotemporal appearance. In: CVPR (2006)

[9]. Person Re-identification in the Wild, Hengheng Zhang ,Shaoyan Sun ,Manmohan Chandraker , Yi Yang In: CVPR (2017)

[10]. Hierarchical Gaussian Descriptors with Application to Person Re-Identification , Tetsu Matsukawa , Takahiro Okabe , Einoshin Suzuki, and Yoichi Sato , IEEE transactions on pattern analysis and machine intelligence (pami), april 2019

[11]. Kernelized Cross-view Quadratic Discriminant Analysis for Person Re-id, Tetsu, Matsukawa and Einoshin ,Suzuki, 16th International Conference , Tokyo, Japan, May 27-31, 2019.