PEDESTRIAN DETECTION IN AUTONOMOUS DRIVING APPLICATION USING CONVOLUTIONAL NEURAL NETWORK

[1]R.SUBHASHNI, [2]E.SRIE VIDHYA JANANI

[1]PG Scholar Computer Science and Engineering,
[2]Asst. Professor Computer Science and Engineering,
Anna University Regional Centre Madurai
E-mail: [1]subhash.alaaj@gmail.com, [2]esrievidhya@gmail.com

Abstract— Pedestrian detection is of high importance to autonomous driving applications. Methods based on Neural Network have shown significant improvements in detection rate, which makes them suitable for this application in which reducing the False Discovery Rate is very important. Convolutional neural network (CNN) has achieved great success in the field of computer vision. CNN takes input data only as image in fixed size and this arises problem in scaling. Hence this paper discusses a filter based feature extraction. Henceforth, the object identification is done without any size constraint. Pedestrian detection also faces the challenges of background clutter and large variations in pedestrian appearance due to pose and changes in viewpoint etc. One of the key contributions is also towards this issue by training the network accordingly. This paper ultimately focuses on reducing the false discovery rate and increasing the accuracy of the detection method. The precision predictive value obtained is 51.46% with a false discovery rate of 48.54% using the benchmark data. The F-Measure value is 65.35%. The number of iterations to minimize the error was achieved to be 1100 epoch and the classification rate of the input data as objects and background is 97.40%.

Keywords— Convolutional Neural Network, Miss rate, False discovery rate, F-Measure.

I. INTRODUCTION

Pedestrian detection is naturally very important in applications such as driving assistance or autonomous driving. More than 3000 pedestrians are killed each year by traffic accidents in India. Looking at the reason of these accidents, it is mostly due to driver's carelessness. Similar statistics have been reported in other countries as well. Henceforth, in recent years the development of pedestrian detection systems is of great importance. Most of the proposed systems use a camera as the sensor, because cameras help to provide the high resolution needed for accurate classification and position measurement. In later section, we give a brief overview of the pedestrian detection systems explained in the literature and tell the problems in the design of their component for classification which in turn motivates the method proposed in this paper. The pedestrian detection applications are mainstream with several car manufacturers already offering warning systems for pedestrian detection application [1], [2], [3], and others want to have highly integrated systems [4]. This has also seen varied development in the field of computer vision [5], [6], [7], [8], [9], incorporating various classification algorithms to identify pedestrians such as Adaboost, SVM, Decision Forests and many more[7], [8], [9]. Recently, deep learning proves to be the top approach for pedestrian detection [10], [11], [12], they also show remarkable results in computer vision applications such as object detection, machine vision [13], [14], [15], [16]. Deep networks readily exchange knowledge between domains like by training on one domain and improving results by fine tuning in another [15]. However, their main drawback has been the accuracy of classification.

II. RELATED WORK

Pedestrian detection is a predominant topic in computer vision. To extract useful information from available image sequences is not a trivial task due to several reasons as follows [15,16]. (1) Pedestrian detection involves a complex uncontrolled and highly influencing external environment. The illuminating conditions keep on changing due to weather. Pedestrians are found mostly in city traffic areas where the background texture (e.g. nearby buildings, vehicles, poles and trees) form a highly messy environment. (2) A wide range of variations exist in pedestrian appearance because of clothing, pose, shadow, motion, size and skin colour. Background clutters in a detection window also confuse the detectors. (3) If the camera is fit on a moving vehicle, this will increase the difficulty to differentiate between background objects and the real pedestrians. (4) Image processing generally involves large computing power and pedestrian detection for intelligent vehicle needs a fast response when considering the speed of the moving vehicle. Previous work on neural networks for pedestrian detection has relied on special-purpose designs, e.g. handcrafted features. Though these proposed methods perform ably, current top methods are all based on decision trees learned via Adaboost [2]. The second is to design good models on single features. A good model is helpful to approximate the feature's Bayes risk. As in [3], One significant model is the part based model. It assumes that an object is composed of deformable parts, whose appearance information is independent, but their spatial information is restricted each other. Deep networks are also versatile, as they do not need task-specific or hand-crafted features and can perform

equally well on both rigid (e.g. traffic signs, cars) and deformable object categories without having to separately model parts and their relationships. Furthermore, in [4], deep networks readily transfer knowledge between domains by training one and improving results by fine tuning in another. However, their main drawback has been the speed of classification. In the area of object detection, the sliding window detection has been the most common. It essentially does image sampling and evaluates every possible patch and is often applied in a cascade style, i.e. progressively employs harder classifiers. This approach is more or less similar to the traditional sliding window cascades as it will in fact examine all locations.

It differs by offering a larger field of view [5] to make its decision, and thus speed up the detection. As in [6,7], An alternative to a cascade approach is to first identify proposal boxes. Previous methods that apply proposal windows are much slower (in tens of seconds) and were not applied to applications that demand accurate localization. However, there are technical issues related to training and testing of the CNNs: the prevalent CNNs require a fixed input image size, which limits aspect ratio and the scale of the input image. This when applied to images of random sizes, current methods mostly try to fit the input image to the fixed size, either via cropping or via warping. But the region after cropping may not depict the entire object, while the warped part may result in unwanted geometric manipulation. Recognition accuracy can be compromised because of content loss or distortion. Besides, a pre-defined scale may not be suitable when scale of objects vary. Fixing input sizes overlooks the issues involving scales. In [8], it introduces a spatial pyramid pooling (SPP) layer to remove the fixed-size constraint of the network.

The last is to take advantage of both models and features, of which the most important work is probably CNN-based detection [9], [10]. It has a powerful and flexible model to implement complex spatial structures and can learn useful features directly from the training data. Success has been achieved in several computer vision tasks, e.g., digit recognition [11], image classification [12], scene labelling [13], etc. Despite the remarkable results, the CNN-based detection system currently has its own problem. That is, it lacks a module to effective model the spatial relationship of high level semantic features. Since each feature has its own spatial prior and generally appears once in a certain area, traditional operations in CNN like convolution, max pooling and average pooling are not a proper choice as we analysed before. This work is related to the work of [9], for both are the CNN-based detection system. The most important difference is that we adopt our proposed spatially weighted max pooling to model high-level semantic features and focus on the spatial modelling process. And we will show that after using this new module to replace traditional convolution operation in learning high-level semantic features, the performance in terms of the detection accuracy will be enhanced greatly. Different algorithms [19] are proposed to detect pedestrians in the image sequences. Two main trends in recent research are motion and shape based. Motion based method takes into account temporal data and detects the features in movement of candidate patterns. On the contrary, the shape based approaches rely on the shape features to identify pedestrians. Motion based method uses rhythmic features or patterns of motions unique to human beings. Motion based approaches provide a better way to minimize the number of false positive candidates, but there are several disadvantages on

motion based approaches. Firstly, motion-based schemes cannot detect stationary pedestrians or unusually posed pedestrians. Secondly, the pedestrian's feet or legs should be visible in to extract features or patterns. Thirdly, this identification procedure requires a chain of images, which delays the identification until several frames are identified and there by this increases the processing time. Shape based methods allow [14] the recognition of both moving and stationary pedestrians. The major difficulty associated with this approach is to overcome the wide range of variations in pedestrian appearances due to different pose, articulations, lighting effect, clothing etc. The key issues here are to find a concise yet sufficient human shape feature representation that could achieve variability and maintain a balance between accuracy of detection and processing time.

III. SYSTEM DESCRIPTION

The major objectives of pedestrian detection system are Minimizing Run-time and detecting the objects from their background. Our method is also based on deep networks, it is very accurate, and at the same time is several times faster than other network methods. It avoids Fixed Input image size, which limits both the aspect ratio and the scale of the input image either via cropping or wrapping. This section discusses the methodology that has guided the main activities of this research fig 1. As it can be recognized from the objectives, the research is typically quantitative in nature.

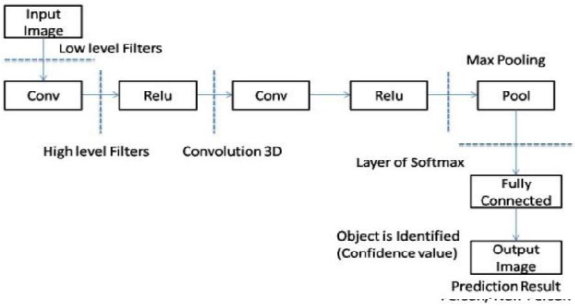

Fig 1 Application framework

The block diagram of our pedestrian detection system is depicted in Fig. 2.This introduces typical CNN building blocks, such as convolution and pooling units and linear filters, with a particular emphasis on understanding back propagation algorithm. It also depicts the size of the input and the properties of the output data with additional specification on each layer involved.

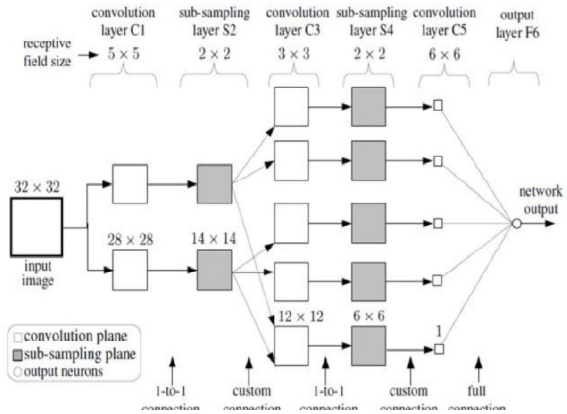

Fig 2 System Architecture

The layers involved in above given architecture are discussed below, (1) Convolutional: Parameters are trained by BP Algorithm. It extracts Low Level Features(Edges, Lines etc.), (2) Pooling: It Reduces variance and Computes Max or Avg value of a particular feature, (3) ReLU (Rectified Linear Units): It provides activation function and improves Nonlinear properties for decision making, (4) Fully Connected: After several conv. and MP layer, we have high level reasoning layer to extract a predictive value, (5) Dropout: For preventing Over fitting among data, (6) Loss Function: the loss functions used are Softmax and Sigmoid cross entropy loss

IV. EXPERIMENTAL EVALUATION

A. Caltech pedestrian detection dataset

We evaluated our results on the Caltech Pedestrian detection dataset [14], which has been the main benchmark for pedestrian detection and a large number of methods have been evaluated on it. We use the standard training and test protocols established in the Caltech benchmark and report the results by measuring the average miss rate. We use the code provided in the toolbox of this benchmark to do the evaluation. The results below are obtained when training with the standard Caltech-Training dataset for training. Other works have included additional Inria dataset, but we chose not to because the INRIA dataset is less relevant to pedestrian detection for autonomous driving.

B. Convolutional neural network (CNN)

CNN has been used in several applications. MATLAB library can is used to create and train a convolutional neural network. Convolutional neural networks processes two-dimensional (2D) image. A CNN consists of three main types of layers: convolution layers, sub-sampling layers, and an output layer. Network layers are arranged in a feed-forward structure: a sub-sampling layer follows each convolution layer, and the last layer is followed by the output layer. The convolution and sub-sampling layers are considered as 2D layers, whereas the output layer is considered as a one dimensional layer. In CNN, each 2D layer is made up from several sectors. A sector consists of neurons that are arranged in a 2D array. Feature map is the output of a plane. For CNNs to perform different visual recognition tasks, hence a training algorithm must be developed. The objective of training a network is to reduce an error function, in terms of the networks actual and desired outputs.

C. Implementation Details

After setting up the network, it is ready to be trained. To begin this process initial weights are set randomly. Then, the training, or learning, begins. Training a network refers to continuously passing over the input dataset in a recursive manner, so that upon every iteration the network learns a new feature of the dataset. Our human brain is similar to this, as we remember the incidents that occur often and we learn by them, the system also leans recursively. Supervised training is done for the network. The network then processes the inputs and compares actual and desired outputs. Calculated errors are then propagated back through the system, asking the system to adjust the weights to have control on the network. This process occurs iteratively and the weights are continually changed. The number of iterations to minimize the error was achieved to be 1100 epoch and the classification rate of the input data as objects and background is 97.40%. The training plot depicting the reduction in MSE is depicted in training plot (fig 3),

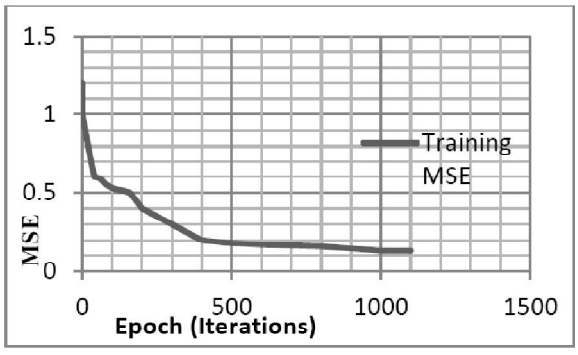

Fig 3 Training plot

The MSE assesses the quality of a predictor. It is good to change the inputs so that the average error over the training data set is tending to zero. This heuristic is applicable at all layers which indicates

that we want the average of the outputs of a node n a layer to be close to zero because these outputs go as inputs to the next layer. The epochs determine the MSE; henceforth if we obtain a minimum MSE then the paper is worthy. We evidently see the reduction of graph error rate to minimum value.

The training of the network with the same set of data is processed iteratively as the connection weights are ever refined. In total, the dataset has 2000 images and corresponding target data. This dataset is used for training the network. The training is followed up by testing the network using test data. The test data is of 10,000 images and they are classified at the rate of 93.59%. Testing is carried out in that trained network to check the efficiency of that network to predict unknown dataset. This dataset is used for testing and based on that a confusion plot is made for object detection (fig 4).

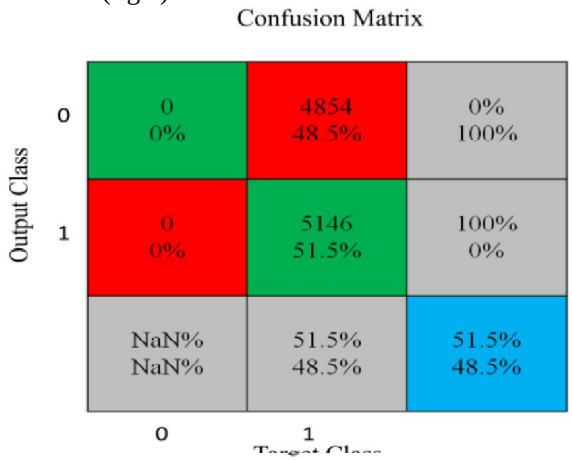

Fig 4 Confusion plot (Test)

D. Analytical Results

Confusion matrix is a table that reports the number of false negatives[FN], false positives [FP],true negatives [TN] and true positives[TP]. This allows more detailed analysis than mere proportion of correct guesses (accuracy). It is not a reliable metric, because it will yield misleading results if the data set is unbalanced. The below given Table I summarizes the F-Measure, False Discovery Rate, Precision Predictive Value and Classification Rate.

TABLE I Analytical Results

PARAMETERS	ACCURACY (%)
PPV	51.46%
FDR	48.54%
F-Measure	65.35%
FPR	30.1%
Classification Rate (TRAIN)	97.40%
Classification Rate (TEST)	93.59%

CONCLUSION AND FUTUREWORK

This network is a flexible solution for handling different scales, sizes, and aspect ratios. These issues are significant in visual recognition, but have received little consideration in the context of deep networks. This Suggests solution to train a deep network with a pooling layer. The resulting net shows outstanding accuracy in detection tasks. This shows that by pre-training on various tasks, convnets can attain top performance on this task. The experience with convnets indicate that they show good promise on pedestrian detection, and that reported best practices do transfer to said task. This demonstrates end-to-end detection solution that is based entirely on deep neural networks, but the proposed solution can become even more interesting when the capacity of the neural networks becomes bigger, i.e. when they can handle larger inputs. For example, right now we can consider a 4x larger field of view, and thus getting the neural network do 16x detections simultaneously. But for networks with even larger fields of view, one can obtain even more speedups. Further enhancement for this paper work would be to identify objects or pedestrians from a real time scenario or videos, which involves various frames of simultaneous object detection.

REFERENCES

[1] He, K.; Zhang, X.; Ren, S.; Sun, J., "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol.37, no.9, pp.1904,1916, Sept. 1 2015.

[2] R. Benenson, M. Omran, J. Hosang, and B. Schiele. Ten years of pedestrian detection, what have we learned? In ECCV, CVRSUAD workshop, 2014.

[3] M. A. Fischler and R. A. Elschlager, “The representation and matching of pictorial structures,” Computers, IEEE Transactions, 2012

[4] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “Cnn features off-the-shelf: an astounding baseline for recognition,”

[5] Angelova, A.; Krizhevsky, A.; Vanhoucke, V., "Pedestrian detection with a Large-Field-Of-View deep network," Robotics and Automation (ICRA), 2015 IEEE International Conference on , vol., no., pp.704,711, 26-30 May 2015

[6] K. van de Sande, J. Uijlings, T. Gevers, and A. Smeulders, “Segmentation as selective search for object recognition,” ICCV, 2011

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” http://arxiv.org/pdf/1311.2524v4.pdf, 2013.

[8] CNN Lab Stanford University [online]. Available: http://white.stanford.edu/teach/index.php/An_Introduction_to_CNN

[9] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun, “Pedestrian detection with unsupervised multi-stage feature learning,” in Computer Vision and Pattern Recognition, 2013.

[10] M. Szarvas, A. Yoshizawa, M. Yamamoto, and J. Ogata, “Pedestrian detection with convolutional neural networks,” in Intelligent Vehicles Symposium, 2005.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, 1998.

[12] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in Neural Information Processing Systems 25, 2012.

[13] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," Pattern Analysis and Machine Intelligence, IEEE Transactions, 2013.
[14] Caltech Pedestrian Detection Benchmark [Online]. Available: http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/
[15] L. Zhao, "Dressed Human Modeling, Detection and Parts Localization", Ph.D. Thesis, The Robotics Institute, Carnegie Mellon University, July 2001.
[16] A. Shashua, Y. Gdalyahu and G. Hayun, "Pedestrian Detection for Driving Assistance Systems: Single-frame Classification and System Level Performance" in Proc. of IEEE Intelligent Vehicle Symposium, 2004.

★★★