A Novel Multikernel Convolutional Neural Network Classification For Pedestrian Detection In Crowded Scenes

[1]M. Angel Shalini, [2]Dr.S. Vijayalakshmi.

[1]Research Scholar, Bharathiar University, Coimbatore, India. E-mail: angelshaliniresearch@gmail.com

[2]Associate Professor, Sri Ramakrishna Arts and Science College for Women. E-mail: vijics@srcw.ac.in

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 28 April 2021

Abstract:

Pedestrian detection is a basic challenge in image processing, and there are many practical applications in the fields of robotics, video surveillance, autonomous driving, and vehicle safety. However, this is still a daunting problem due to the huge differences in lighting, clothing, color, size, and posture. A pedestrian detection method based on MKCNN (Multi-kernel Convolutional Neural Network) classifier is implemented in this paper. High accuracy is achieved by automatically optimizing the feature representation of neural network recognition and regularization problems. We evaluated the proposed method in a complex database that includes pedestrians in an urban environment without being restricted by posture, motion, background, and lighting. Experiments using the pedestrian data set of the California Institute of Technology show that the use of MKCNN can improve detection accuracy. Using the CNN model, our pedestrian detection method provides a higher detection performance for the Caltech dataset.

Keywords: Pedestrian detection, CNN, Kernel, Caltech.

Introduction:

Pedestrian detection has been quite possibly the most broadly considered issue in PC vision. One explanation is that pedestrian detection is the initial step for various applications, for example, keen video observation [1], human finding for military applications, human-robot association, clever advanced administration, and driving help framework. Pedestrian detection is a quickly advancing territory, as it gives the principal data to semantic comprehension of the video recordings. On account of the different way of attire, various conceivable body explanations, diverse enlightenment conditions, the presence of blocking extras, regular impediment between pedestrians, and so forth, pedestrian detection is as yet a difficult issue in PC vision.

Pedestrian detection is an uncommon assignment of item detection. Its mechanical advancement is firmly identified with the improvement of general article detection. This association can be depicted as follows: The overall article detection calculation can be utilized for pedestrian detection after fitting improvement. Pedestrian detection is a sort of particle detection, and the issues it studies can advance the improvement of general item detection from another view. In our paper, we attempt to utilize the highlights of various pieces and improve the grouping precision of CNN. The various portion learning (MKL) model is an adaptable learning model. In the new exploration, the MK learning (MKL) can get higher-order precision than the sole one. As the MKL utilizes various mixes of piece works and has bigger adaptability, its exhibition is typically better. Developing the MK model, indeed, is the way toward looking for the blend of parts to get the best characterization exactness.

The objective of this paper is to introduce our novel pedestrian identifier dependent on Multi-Part Convolutional Neural Organization (MKCNN) in observation recordings. The majority of the proposed frameworks utilize a camera as the sensor since cameras can give the high goal expected to precise characterization and position estimation. Cameras can likewise be imparted to other wellbeing support subsystems in the vehicle, for example, a path keeps help framework, improving the value advantage proportion of the camera. In this paper, we propose a novel, unique pedestrian locator which has a few engaging properties. It contains a bunch of broad part indicators that can be prepared on feebly marked information, i.e., it doesn't need part explanations in the pedestrian jumping boxes. We present a successful CNN design to diminish the hour of highlight extraction and preparing.

The rest of this paper is organized as follows. In Section II, the introduction to the system is presented. In Section III, the previous works are reviewed. We describe the proposed pedestrian detection system in Section IV. Section V shows experimental results and analysis. We concludein Section VI.

Related Works:

There is broad writing on pedestrian detection calculations. Pedestrian detection strategies can be essentially separated into two classes: The first is to utilize conventional pedestrian detection strategy which needs a manual plan to extricate highlights for every proposition and arrange them by a teachable classifier, basic strategies for pedestrian detection task is to utilize sliding window-based procedures for proposition age, highlights removed from picture primarily depend on histograms of slope direction (Hoard [14]) or scale-invariant element change (Filter [15]), and support vector machine (SVM [16]) or Versatile Boosting [17] as the pedestrian order techniques. Those low-level highlights planned by hand-made get great achievement. The subsequent methodology is through the profound learning innovation to accomplish the objective of pedestrian detection, and have beaten condition of-workmanship execution on a few pedestrian datasets. This technique utilizes the convolutional neural organization (CNN) to consequently extricate the worldwide and semantic highlights of the picture, create great up-and-comer boxes and arrange every up-and-comer.

In 2003, Viola and Jones [2] first utilized picture force data and movement data joined with Adaboost classifier to acknowledge pedestrian detection and following, which pulled in the consideration of analysts to the issue. At that point, Dalal and Triggs [3] proposed pedestrian detection techniques dependent on the histogram of situated slope (Hoard) and backing vector machine (SVM) classifiers, which accomplished almost 100% detection impact on the MIT pedestrian dataset [4] and enormously advanced the improvement of pedestrian detection innovation because of its capacity to precisely addressing objects. Afterward, pedestrian detection techniques dependent on fake component extraction joined with AI classifier have become the standard [5–9], and the greater part of the specialists have improved or enhanced on this worldview.

Girshick et al. [10] consolidated customary AI strategies with CNN and proposed a detection system dependent on RCNN, of which the specific inquiry is utilized to get whatever number of article recommendations as would be prudent; CNN is utilized to separate the highlights of the proposition rather than manual extraction and SVM is utilized to characterize the component vectors. The outcomes showed that the RCNN strategy claims the incredible preparing capacity of CNN in the field of PC vision. Afterward, spatial pyramid pooling (SPP) Net [11] and Quick RCNN [12] have been improved by presenting the SPP layer and district of interest (return for capital invested) pooling layer, individually. Be that as it may, the quantity of the proposition is excessively enormous, which is joined by a lot of computational utilization in the proposition's age cycle, and restricts its application situations. In light of this issue, Quicker RCNN [13] proposed a locale proposition organization (RPN), which is more precise than a particular pursuit utilized by RCNN. Likewise, the recommendations are produced under the brought together organization system through network sharing, and the preparation and learning measure is finished by utilizing the Softmax classifier.

Multi-Kernel CNN

Convolutional Neural Networks (CNN) are similar to traditional ANNs in that they are composed of neurons that optimize themselves through learning. Each neuron continues to receive input and perform operations (for example, the dot product is followed by a non-linear function): from the original vector of the original image to the final result of the class score, the entire network will track and represent the function (weight).The last layer contains class loss functions, and all common techniques developed for traditional ANNs will still apply.

The main driver of CNN is that posts will be composed of images. This allows the architecture to be configured to best suit the needs of specific data types. One of the main differences is that the neurons that make up each layer in the CNN are composed of neurons arranged in three dimensions (the spatial dimension of the entrance (height and width) and depth). Depth does not refer to the total number of layers in the RNA but refers to the third dimension of the activation volume.

Overall architecture

CNN consists of three types of layers. These are folded layers, grouped layers, and fully connected layers. When these layers overlap, a CNN architecture is formed. The simplified CNN architecture for pedestrian classification is shown in Figure 1.

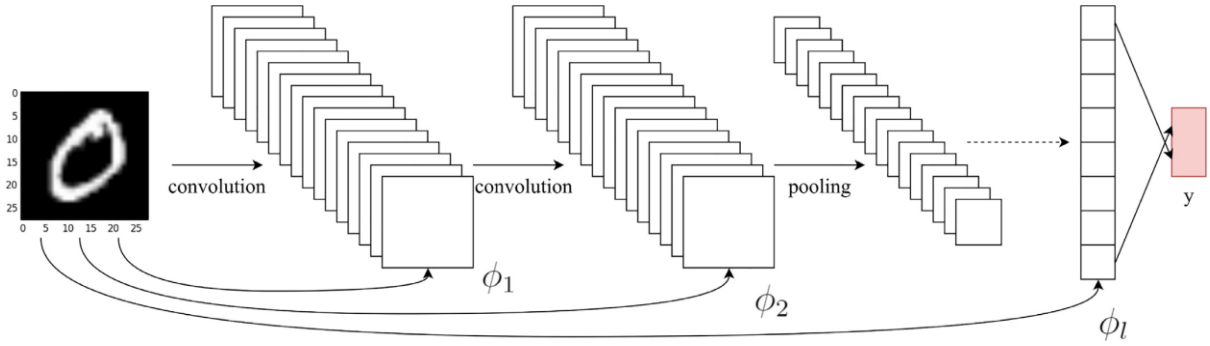


Figure 1. A simple CNN architecture

The core functions of the CNN example above can be divided into four key areas.

1. Like other forms of ANN, the input layer contains the pixel values of the image.

2. The convolutional layer determines the output of neurons related to the local area of the image. Enter by calculating the dot product between its weight and the area associated with the input volume. The linear unit (usually abbreviated as ReLu) aims to apply a "basic" activation function (such as the S-shape) to the activation output produced by the previous layer.

3. The pooling level only reduces the space size of the input, thereby further reducing the number of parameters in the flip-flop.

4. The fully connected layer performs the same function as the standard ANN and attempts to derive category estimates based on triggers. Again, it can be assumed that ReLu can be used between these layers to improve performance.

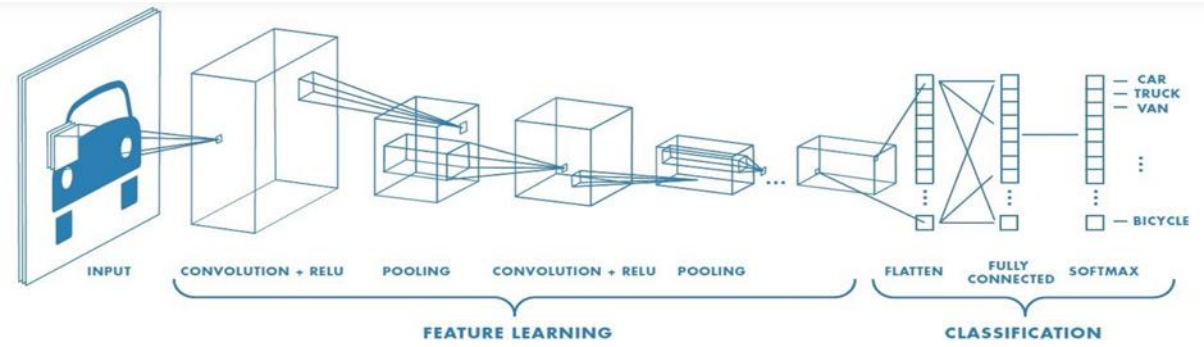
However, it should be noted that it is not enough to understand the general architecture of a CNN architecture. Building and optimizing these models can be time-consuming and confusing. Now, we will examine each layer in detail and explain their hyperparameters in detail.

ConvolutionalLayer:

Convolutional Neural System, referred to as CNN, is a special type of nervous system model designed to process information from two-dimensional images, although it can be used for one-dimensional and three-dimensional information. Using a channel (small matrix), its size can be determined. The channel covers the entire image network. Your task is to reproduce its properties from the estimated value of the first pixel.The channel continuously moves n units along the right side (they may fluctuate) and performs comparison activities. By displaying each location, you will get a frame much smaller than the information grid.

Convolutional Layer has the following components:

- Filters
- Activation maps
- Parameter sharing
- Layer-specific hyperparameters



The parameters of the convolutional layer focus on the use of trainable kernels. In this article, multi-kernel learning is achieved at this level. We recommend using multiple kernels for all convolutional layers. In particular, we use the RBF Gaussian kernel.

$$k(\mathbf{x},\mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{\sigma^2}\right)$$

but with different values of parameter σ for the different computing layers.Each level should have its optimal kernel parameters. For each module, we perform the following process to optimize the kernel parameters separately. The figure 2 shows the specific components that make up each multi-kernel CNN block.

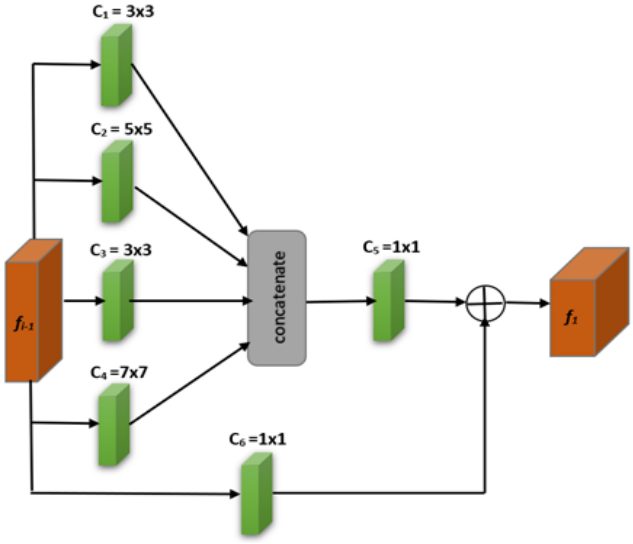


Figure 2. Components of the multi-kernel CNN block

Example:

In Figures3 and 4 below, to detect the horizontal and vertical images with the help of a matrix, let's consider a greyscale image, with a 6 x 6 matrix and a filter of 3x3 applied to it.

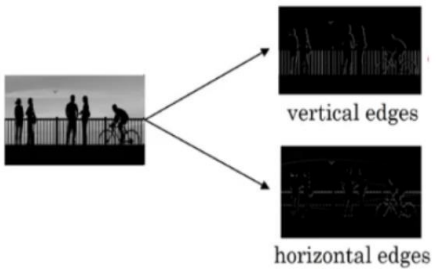


Fig 3: Identifying the edges

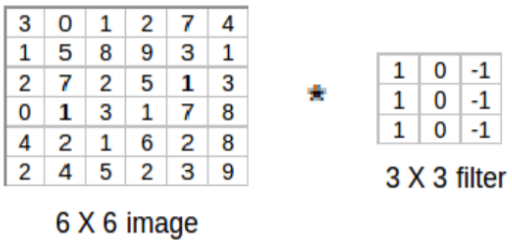


Fig 4: Scanning of the image pattern

After the above calculations of the matrix, we get the matrix as shown in fig:5. To calculate it, we take the initial 3 X 3 framework from the 6 X 6 picture and increase it with the channel. Let's consider the following matrix of 4x4 order and the calculation takes place as: for example, 3*1 + 0 + 1*-1 + 1*1 + 5*0 + 8*-1 + 2*1 + 7*0 + 2*-1 = - 5. To compute the second component of the 4 X 4 order, we will move the channel one step ahead to the right side of the original Greyscale matrix and so on:

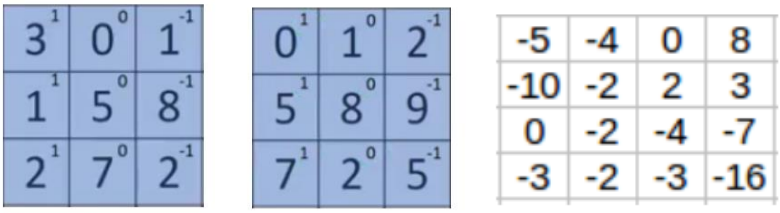


Fig 5: Matrix calculation in convolutional layer

Fig 6: Convolving over the entire image

The way to detect the vertical edge in the image is to look for the pixel values as, if the pixel values are greater, then brightness at that part of the image will be more and if the value is less, it will be dark.

Pooling Layer:

Spatial pooling (also called down-sampling) down-samples each feature map, but contains the most important data. There are different types of spatial pooling: maximum, average, sum, etc. to represent the spatial neighborhood (such as a 2×2 window), and occupy the largest component of the newly designed highlight map in this window) or the sum of all components in this window. Gradually, Max Pooling seems to work better.

As shown in the figure 7 below, Max pooling takes the biggest component from the rectified feature map. Taking the biggest component could likewise take the normal pooling. The entirety of all components in the element map call as sum pooling.

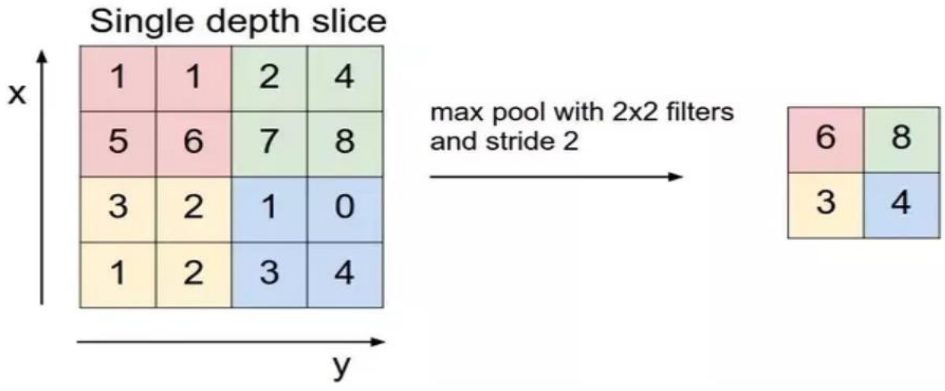


Fig 7: Max Pooling

Fully Connected Layers:

In a fully connected layer, every neuron in one layer is connected to every neuron in the other layer. FC works like a traditional neural system, multilayer perceptron (MLP). And the structure created in the previous stages of CNN. As shown in the figure 8 below, the feature map matrix is transformed into a vector with an FC layer (x1, x2,...xn), and we combine them to create a model.Then, we use the activation function to classify the output.

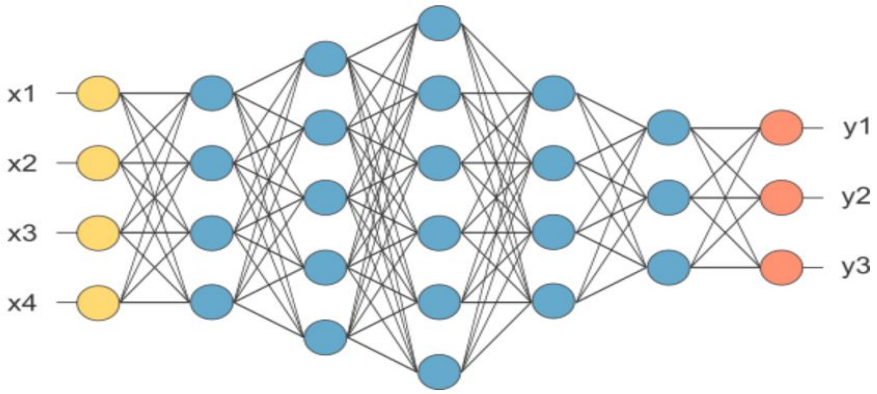


Fig 8: Fully connected Layer

Experiments and results

In this section, we use a PC with a 3.6 GHz Intel i3 processor, 16 GB RAM, and a TITAN XGPU with 12 GB RAM. We use the famous Caltech Pedestrian Dataset to train andevaluate our proposed model. The California Institute of Technology's pedestrian record contains approximately 10 hours of 640 x 480 30 Hz video captured from a moving vehicle in an urban environment. Approximately 250,000 images (in approximately 1 minute of 137 segments) were evaluated for a total of 350,000 bounding boxes and 2300 individual pedestrians. The annotation contains the time correspondence between the bounding box and the detailed occlusion labels.



. S D

In this section, a confusion matrix (CM) is used to evaluate the performance of the proposed training model, and then various CM metrics are inferred. In particular, the metrics such as sensitivity, specificity, precision, F1 score, and precision are defined as follows:

$$Sensitivity = \frac{TP}{TP+FN}$$

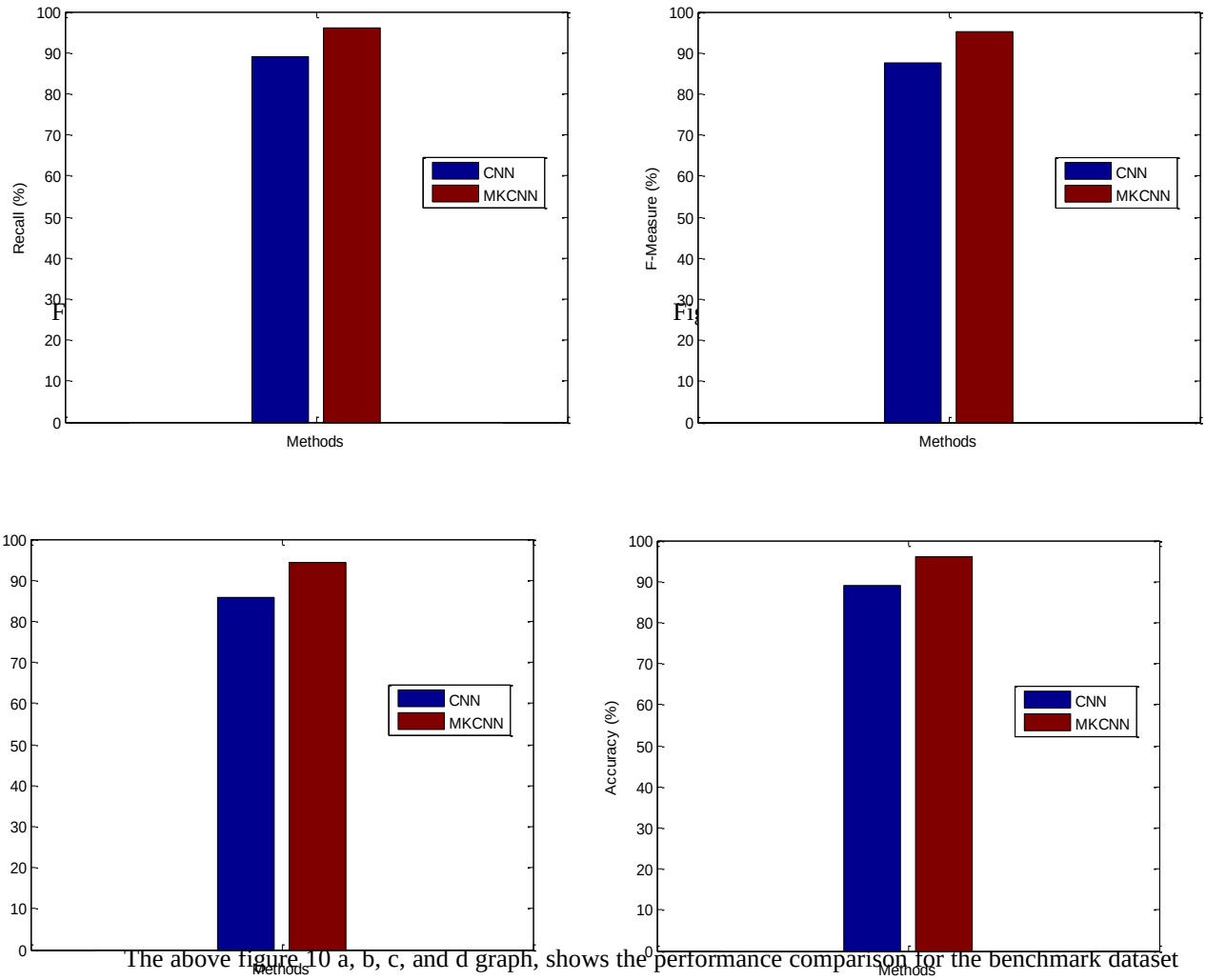
$$Specificity = \frac{TN}{TN+FP}$$

$$Precision = \frac{TP}{TP+FP}$$

$$F1 - score = 2x\frac{Sensitivity\ x\ Precision}{Sensitivity + Precision}$$

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN}x100\%$$

where TP, TN, FP, and FN are the numbers of true positive, true negative, false positive, and false negative, respectively.



F Fig

The above figure 10 a, b, c, and d graph, shows the performance comparison for the benchmark dataset namely Caltech Pedestrian Detection Dataset using the Accuracy, Precision, Recall, and F-Measure as the performance metrics. The above graph shows that the proposed MKCNN classification algorithm shows better improvements in all the aspects than the traditional CNN.

Conclusion:

In this paper, we introduced the multi-kernel CNN pedestrian detection and classification unit using the Caltech pedestrian detection dataset. This work presents the idea of a multi-core convolutional neural network and shows how to study and use it to detect pedestrians in crowded scenes. The proposed method uses an improved kernel for CNN cascade so that CNN cascade can significantly reduce processing time while maintaining high CNN performance.The experiment tested the MKCNN method using the Caltech pedestrian records and showed a significant improvement in recognition performance and classification speed compared with traditional CNN.

References:

1. R. Girshick. Fast R-CNN. In International Conference on Computer Vision (ICCV), (2015).
2. Paul, V., et al.: Detecting pedestrians using patterns of motion and appearance. In: The 9th IEEE International Conference on Computer Vision, pp. 734–741. IEEE Computer Society, Washington DC (2003)
3. Navneet, D., Bill, T.: Histograms of oriented gradients for human detection. In: The 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 886–93. IEEE Computer Society, Washington DC (2005)
4. Constantine, P., Tomaso, P.: A trainable system for object detection. Int. J. Comput. Vision 38(1), 15–33 (2000)
5. Dollar, P., et al.: Fast feature pyramids for object detection. IEEE Trans. Pattern Anal. Mach. Intell. 36(8), 1532–1545 (2014)
6. Lahouli, I., et al.: Hot spot mthod for pedestrian detection using saliency maps, discrete Chebyshev moments and support vector machine. IET Image Proc. 12(7), 1284–1291 (2018)
7. Enzweiler, M., Gavrila, D.: Monocular pedestrian detection: Survey and experiments. IEEE Trans. Pattern Anal. Mach. Intell. 31(12), 2179–2195 (2009)
8. Ess, A., et al.: A mobile vision system for robust multi-person tracking. In: The 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society,Washington DC (2008)
9. Felzenszwalb, P., et al.: A discriminatively trained, multiscale, deformable part model. In: The 26th IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, Washington DC (2008)
10. Girshick, R., et al.: Region-based convolutional networks for accurate object detection and segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 38(1), 142–158 (2015)
11. He, K., et al.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: The 13th European Conference on Computer Vision, pp. 346–361. Springer Verlag, Berlin (2015)
12. Girshick, R.: Fast R-CNN. In: The 15th IEEE International Conference on Computer Vision, pp. 1440–1448. IEEE Computer Society, Washington DC (2016)
13. Ren, S., et al.: Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39(6), 1137–1149 (2015)
14. Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: An evaluation of the state of the art. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), (2012).
15. Dalal N, Triggs B. Histograms of Oriented Gradients for Human Detection[C] Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. IEEE: 886-893, (2005)
16. Y. Ke and R. Sukthankar. “PCA-SIFT: A more distinctive representation for local image descriptors”. CVPR, Washington, DC, USA, 66-75, (2004).
17. S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. volume 0, pages 1–8, Los Alamitos, CA, USA. IEEE Computer Society.1, (2008)