Monitoring social distancing through human detection for preventing/reducing COVID spread

Mohd. Aquib Ansari¹ · Dushyant Kumar Singh¹

Received: 10 October 2020 / Accepted: 26 March 2021 / Published online: 14 April 2021
© Bharati Vidyapeeth's Institute of Computer Applications and Management 2021

Abstract COVID-19 is a severe epidemic that has put the world in a global crisis. Over 42 Million people are infected, and 1.14 Million deaths are reported worldwide as on Oct 23, 2020. A deeper understanding of the epidemic suggests that a person's negligence can cause widespread harm that would be difficult to negate. Since no vaccine is yet developed, social distancing must be practiced to detain COVID-19 spread. Therefore, we aim to develop a framework that tracks humans for monitoring the social distancing being practiced. To accomplish this objective of social distance monitoring, an algorithm is developed using object detection method. Here, CNN based object detector is explored to detect human presence. The object detector's output is used for calculating distances between each pair of humans detected. This approach of social distancing algorithm will red mark the persons who are getting closer than a permissible limit. Experimental results prove that CNN based object detectors with our proposed social distancing algorithm exhibit promising outcomes for monitoring social distancing in public areas.

Keywords COVID-19 · Social Distancing · Surveillance · Human Detection · CNN

1 Introduction

COVID-19 (Corona Virus) is an infectious disease that has proved to be an epidemic, declared by the World Health Organization (WHO). It was first reported in Wuhan, China, in late 2019. As of Oct 23, 2020, 217 countries and regions around the world are affected by COVID-19 and reported approximately 42 Million confirmed cases and 1.14 Million deaths. Figure 1 illustrates the total number of cases and the total number of deaths from Jan 22, 2020, to Oct 23, 2020 [1]. According to World Health Organization [2], a person can become infected with COVID-19 if he comes in contact with other virus-infected persons. Till date, no medicine or vaccine is yet developed to quell this deadly virus. Therefore, there is a need to look for an alternative control measure to prevent the spread of this fatal virus.

As it is well said, prevention is better than cure, WHO has suggested several safety measures to minimize the transmission of coronavirus. In the present scenario, social distancing [3, 4] has proved to be one of the most exquisite alternative methods as a spread stopper. Social distancing can also be referred to as “physical distancing,” which means maintaining a distance between yourself and the people around you. Social distancing helps to lessen physical contact or interaction between possibly COVID-19 infected persons and healthy individuals. According to WHO's standard prescriptions, everyone should keep a distance of at least 6 feet between each other to follow the social distancing. This is a prominent way to break the chain of contagion. Therefore, all the affected countries have adopted social distancing.

Monitoring social distancing in real-time scenarios is a challenging task. It can be possible in two ways: manually and automatically. The manual method requires many

✉ Mohd. Aquib Ansari
mansari@mnnit.ac.in

Dushyant Kumar Singh
dushyant@mnnit.ac.in

¹ CSed, MNNIT Allahabad, Prayagraj, India

Fig. 1 Total Cases vs. Total Deaths [1]

physical eyes to watch whether every individual is following social distancing norms strictly. This is an arduous process as one can't keep their eyes for monitoring continuously at 24×7 . Automated surveillance systems [5, 6] replace many physical eyes with CCTV cameras. CCTV cameras produce video footage, and an automated surveillance system inspects this footage. The system raises alerts when any suspicious event occurs. In view of this alert, security personnel can take relevant actions. Therefore, the automated monitoring system has surpassed several limitations of the manual monitoring method.

This research aims to limit the impact of the coronavirus epidemic with minimal harm to economic artifacts. In this paper, we have proposed an effective automatic surveillance system that helps to locate each person and monitors them for the social distancing parameter. This application is suitable for both indoor and outdoor surveillance scenarios. It can be used significantly in various places like railway stations, airports, megastores, malls, streets, etc. The proposed approach can be seen as a combination of two main tasks, mentioned as:

(i) Human detection and tracking

(ii) Monitoring of social distancing among humans

In the first task, this research addresses the problem of human detection and tracking [6, 9, 12] in the surveillance video. Human detection is a two-stage process that involves the localization of an object in the first stage and classification of the localized object in the second stage. This paper has presented a human detection technique based on visual specific learning through deep neural networks in the video feed. The second task focuses on calculating distance among humans in public areas using our proposed algorithm. The decision is made on social distancing if followed. If not, then the persons who do not follow the social distancing criteria are highlighted with a red rectangle. On seeing this, security personals can take

any action related to social distancing rules so that it can be followed strictly.

This paper is structured into five sections. Section one describes the motivation and introductory knowledge of social distancing. Section two is designed to provide a vast study on traditional and recent approaches to various human detection techniques. Section three focuses on deep learning based human detection models. The experimentations and its detailed analysis are styled in section four. At last, the conclusion followed by the future scope is described in section five.

2 Literature review

In 2001, a very popular approach for object detection was proposed by Viola and Jones [10]. They used Haar features for features extraction and cascade classifiers with ada-boost learning algorithm for classification purposes. This method is 15 times faster than traditional approaches. Fu-Chun Hsu et al. [11] proposed a hybrid approach to detect the head and shoulders by fusing motion and visual characteristics. The authors found that the Histogram of Oriented Optical Flow (HOOF) descriptor is a better choice for segmenting the moving object in video sequences and can handle cluttered and occluded environments efficiently. Vijay and Shashikant [13] proposed a real-time pedestrian detection for advanced driver assistance. This system detects the pedestrian using Edgelet features to improve the accuracy and a classifier based on the k-means clustering algorithm to lessen the system complexity. Suman Kumar Choudhury et al. [14] proposed an advance pedestrian system by incorporating the background subtraction technique to extract moving objects, Silhouette Orientation Histogram, and Golden Ratio Based Partition to extract meaningful information from the moving objects and HIKSVM for object classification. This system can deal with occlusion efficiently and achieved accuracy up to

98.36%. Seemanthini and Manjunath [15] deployed the human detection technique for an action recognition system. Singh et al. [17] proposed a human detection framework for extensive surveillance in the city through CCTV cameras. They used the background subtraction technique to segment moving objects, HOG descriptor to extract features and SVM for object classification.

Earlier, object detection frameworks implemented the Sliding Window concept [18] for object localization within an image. According to this approach, an image is divided into a particular size of blocks or regions. Further, these blocks are categorized into their respective classes. Various handcrafted feature extraction techniques like HOG [8], SIFT [19], LBP [29], etc. are used to evaluate the attributes or features. Furthermore, these attributes are used to build the classifier to locate the object on the image's grid. However, this grid-based archetype requires high computational cost and sometimes yields high false-positive rates. Therefore, an effective object classification & localization framework is needed to detect several objects with diverse scales within an image. Additionally, it should reduce the computational cost and false-positive rate. Recently, significant advances have been observed in object detection using deep convolutional neural network (CNN) [20, 22–24]. Convolutional neural networks (CNN) are a class of intensive, feed-forward artificial neural networks that have been used to perform accurately in computer vision tasks, such as image classification and detection. CNN is capable of extracting robust features with the help of the convolution process. Its strong attribute representation capability played a vast role in object detection [7, 21]. Aichun, Tian, and Qiao [16] proposed a deep hierarchical model for multiple human upper body detection. This model employs a candidate-region convolutional neural network (CR-CNN) with multiple convolutional features to accommodate the local as well as contextual information from the image and has achieved accuracy up to 86%.

The researches presented in literature illustrate that the object detection is of vital role in computer vision due to its number of practical use cases, e.g., face detection, pedestrian, detection, activity recognition, medical imaging, etc. This paper has extended the role of object detection to reduce the vivid spread of COVID-19. Therefore, we aim to develop an application for analyzing social distancing among persons using an efficient object detector.

3 Proposed state of the art framework for monitoring social distancing

The overall scenario of monitoring for social distancing in public, as proposed, is presented here in Fig. 2. CCTV cameras available at any public place can be used for

surveillance, i.e., monitoring social distancing. Video stream/frame sequences received from these cameras are fed to the object detection and tracking module for locating human presence in the scene. The parameters like 'centroid' of object/person location and 'distance' among many such centroids are evaluated for measuring the degree of social distancing practiced. An alert is generated in changing the color of the bounding box of humans detected, from green to red. The color of the bounding box is green until there is a permissible distance between any two persons. As when this decreases, the color of bounding boxes changes to red, which presents the social distancing violation.

Sliding window based region proposals is a simple and straightforward approach to design an efficient object detector. According to this approach, the image or frame is divided into the particular size of blocks or regions. Further, these blocks are categorized into their respective classes. The categorization of blocks can be possible by different machine learning and deep learning paradigms. It might also be possible that regions contain part of the object, which introduces many bounding boxes around the object. To deal with this problem, the Non-Maximum Suppression (NMS) [26] algorithm is used to locate the object correctly within an image, which suppresses the low bounding boxes and keeps only the best.

This paper has exercised deep learning based technique to detect the presence of human with the help of the sliding window based region proposal algorithm. The proposed technique is quite helpful in object detection and localization, which is described in Sect. 3.1. Furthermore, these employed techniques are used in the social distancing algorithm to see if people are following the distancing criterion. The algorithm of social distancing is described in Sect. 3.2.

3.1 Proposed CNN model

Convolutional neural network (CNN) [7, 25] has drawn much attention to the research community's attitude and can be successfully embedded in a broader image classification paradigm. It takes an image as input, assigns significance to different objects within an image based on trainable weights & bias, and effectively differentiate each object. This paper introduces two CNN based sequential models to detect the presence of an individual within an image. The general overview of these proposed models is shown in Table 1. These models consist of a convolutional layer, pooling layer, flatten, fully connected layer 1 & 2, and output layer. The only difference between these two models is that Model 1 consists of two convolutional layers with two pooling layers, while Model 2 consists of three convolutional layers with three pooling layers. Due to this

Fig. 2 Complete block diagram for practicing social distancing

Table 1 Proposed models configuration

Model		Model 1		Model 2	
Layer	Size	Output shape	Parameters	Output shape	Parameters
Conv2D	32 filters	(None, 126, 62, 32)	896	(None, 126, 62, 32)	896
MaxPooling2	(2, 2)	(None, 63, 31, 32)	0	(None, 63, 31, 32)	0
Conv2D	48 filters	(None, 61, 29, 48)	13,872	(None, 61, 29, 48)	13,872
MaxPooling2	(2, 2)	(None, 30, 14, 48)	0	(None, 30, 14, 48)	0
Conv2D	64 filters	–	–	(None, 28, 12, 64)	27,712
MaxPooling2	(2, 2)	–	–	(None, 14, 6, 64)	0
Flatten	–	(None, 20,160)	0	(None, 5376)	0
FC1	512	(None, 512)	10,322,432	(None, 512)	2,753,024
Dropout	0.30	(None, 512)	0	(None, 512)	0
FC2	128	(None, 128)	65,664	(None, 128)	65,664
Output Layer	1	(None, 1)	129	(None, 1)	129
Total trainable parameter		10,402,993		2,861,297	

variation, Model 1 produces approximately 10,402,993 trainable parameters, whereas Model 2 produces approximately 2,861,297 trainable parameters.

Figure 3 shows the graphical structure of Model 2, which takes a color image of size $128 \times 64 \times 3$ as input and produces its predicted value as output. It has three convolutional layers, three pooling layers, two fully connected layers, and one output layer. The first convolutional layer involves 32 filters of convolution of each size of

3×3 , while the second and third convolutional layer involves 48 filters and 64 filters of convolution, respectively. The convolutional layer uses (1, 1) stride value. The pooling layer involves (2, 2) pool size to reduce the size of an image. Two fully connected layers (FC) of size 512 and 128 respectively are used to train the network. The size of the output layer is one neuron that indicates returns True or False value. We used ‘Relu’ activation function in convolutional layers and fully connected layer. While

Fig. 3 Proposed CNN architecture of Model 2

‘Sigmoid’ function is used in the output layer that yields output vectors where each element is a probability. A dropout rate of 30% is used in the first FC layer to overcome the overfitting problem.

3.2 Proposed social distancing monitoring algorithm

It is the second phase of our proposed framework. The proposed social distancing monitoring algorithm carried two main functions. Function1 helps to find out the locations of the objects in an image. It uses the human detection technique and provides the human locations in the form of coordinate values like X_A (left), Y_A (top), X_B (right), and Y_B (bottom). From these coordinate values, the centroid values of different objects are identified. The evaluation of the centroid value for an object is shown in Eqs. 1 and 2.

$$X = \frac{(X_A + X_B)}{2} \quad (1)$$

$$Y = \frac{(Y_A + Y_B)}{2} \quad (2)$$

where X_A , Y_A , X_B , and Y_B are the coordinate values (left, top, right, bottom) of an object. X and Y are centroid coordinates or values. Further, these parameters are passed to the next function to measure social distancing.

Function2 finds out the distance between two objects using Euclidean distance [27], which decides the closeness between them and shown in Eq. 3. The decision is made on comparing this distance vector with the pre-define threshold value. If Euclidean distance is less than some threshold value, then it is assumed that these two objects are not obeying the criteria of social distancing or have not made enough distance between them. On breaching these security concerns, the spread of the coronavirus could be possible. So, an alert is generated to the security personals by drawing the red rectangle around the objects. Therefore, an intended person or observer can take appropriate action or ask them to maintain social distance.

$$\text{Euclidean Distance}(d) = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \quad (3)$$

where (X_1, X_2) and (Y_1, Y_2) are centroid values of two objects.

Algorithm: Social Distancing Measurement

Input: V_N : Video V Containing N number of frames of size 320x320x3 or 418x418x3

Output: D: Distance vector between two objects

Initialize Parameter:

```

Threshold = 85.0,           // Distance between two objects
Human_Count = 0,           // Number of persons present within input scene; default value = 0
cv = [],                   // Coordinate values of object presence
lcv = [],                  // List of coordinate values of objects presence
cen = [],                  // Centroid values of object
center = [],               // List of centroid values of objects

```

Function1 Object_Coordinates (V)

Pick = Human_Detection_Framework(V_N) // provides the number of objects with their coordinate values.

For (X_A, Y_A, X_B, Y_B) in Pick:

```

    cv = [ $X_A, Y_A, X_B, Y_B$ ]
    cd1 = (( $X_A + X_B$ ) * 0.5)           // Centroid Value for Coordinate X
    cd2 = (( $Y_A + Y_B$ ) * 0.5)           // Centroid Value for Coordinate Y
    cen = [cd1, cd2]
    center.append(cen)
    lcv.append(cv)
    Human_Count += 1

```

End For

Return Human_Count, lcv, center, Image

End Function1

Function2 Social_Distancing ()

Human_Count, lcv, center, Image = Object_Coordinates(V_N)

For z in range(Human_Count):

For x in range(z):

d = distance.euclidean(center[z], center[x]) // Calculate Euclidean Distance between two Objects

If $d \leq \text{Threshold}$:

// Draw red rectangles around the objects

END If

End For

End For

End Function2

4 Experiments and analysis

In this paper, CNN based techniques have been developed to detect the presence of humans. In addition, the practice of social distancing is performed from these proposed techniques. All the experimentations have been performed on Intel core i3-5005 CPU@2.00 GHz processor of 64-bit type system and Google Colab in Python. We used the INRIA image dataset [28] for training purposes. It consists of a total of 6562 images in which 4146 images are

negative, and 2416 images are positive. We split our image dataset into training and testing module, in which 2316 positive and 4046 negative images are used for training purposes, and 100 positives and 100 negative images are used for testing purposes. This dataset contains static images and incorporates variations in humans with 64×128 resolution. In testing with real-time video sequences for sliding window-based modules, the minimum window size is (64, 128), step size is (10, 10), the downscale is 1.25. It process approximately 567 windows

each of size 64×128 for an image of size $264 \times 400 \times 3$.

The proposed technique has adopted the CNN architecture for human detection. It uses sliding window concept for region proposal and Convnet for human detection. As a part of experimentation for deriving an optimized model, the two different models, namely Model 1 and Model 2, have been proposed. These Models are hyper-tuned with different parameters like Batch size, Dropout rate, Activation function, Optimizer, and Epochs. Table 2 illustrates the hyper-parameter tuning for different variants of these Models.

These proposed models (namely Model 1 and Model 2) are trained and tested over different hyper-parameters and provide appropriate outcomes, presented in Table 3. Model 1 is hyper tuned with ‘8’ batch size, ‘30%’ dropout rate, ‘Relu’ activation function for the convolutional layer and FC layer, ‘Sigmoid’ activation for the output layer, ‘Adam’ optimizer, and ‘120’ epochs. It yields 97% testing accuracy. The structure of Model 2 is hyper-tuned in the same way as Model 1, except both model has a different structure, dropout rate, and optimizer parameter. Model 2 yields 98.50% testing accuracy.

On performing the experimentations, it is observed that the training and testing costs of CNN based models are highly expensive while running in our system (i3 Processor with 4 GB Ram). Conversely, it runs smoothly in the Google Colab platform (in GPU environments) with less timing cost. Table 4 shows the overall training and testing time comparison through our system and Google Colab. Here, an image of size $264 \times 400 \times 3$ is used to evaluate the testing time of the proposed model.

Figure 4 illustrates the accuracy and loss curve of over 120 epochs for Model 1 and Model 2. On analyzing both Models, we find that Model 2 provides more encouraging results than Model 1, which offers higher accuracy and lower loss value.

Table 5 shows the comparative analysis of our proposed Models with existing human detectors. Upon exploration, it is found that both models provide excellent results. But, Model 2 has achieved the highest accuracy among all and proved to be the most efficient human detection technique.

An appropriate positioning and placement of the camera for likely receiving the video stream in a physical real-time system is a most challenging task. In the experiment’s context, it is seen that if the camera is placed near to the objects/humans, the object seems bigger and if the camera is placed away from objects, the object size reduces in the images captured. This creates a problem in acquiring relevant features for object/human detection. Therefore, the camera location is adjusted based on practical calibration taking our algorithm in view. Figure 5 exhibits some resulting images for performing social distancing, which carries raw detection before applying NMS and final detection result after applying NMS.

5 Conclusion

This article suggests deep learning based human detection techniques to monitor social distancing in the real-time environment. These techniques have been developed with the help of deep convoluted network that has used sliding window concept as a region proposal. Further, they are used with the social distancing algorithm to measure the distancing criteria among people. This evaluated distancing criteria decide whether two peoples are following social distancing norms or not. The extensive experiments were performed with CNN based object detectors. In experiments, it is found that CNN-based object detection models are better in accuracy than others. Sometimes, it produces some false positive instances when dealing with real-time video sequences. In the future, different modern object detectors like RCNN, Faster RCNN, SSD, RFCN, YOLO, etc. may be deployed with the self-created dataset to increase detection accuracy and reduce the false positive instances. Additionally, a single viewpoint obtained from a single-camera can’t reflect the result more effectively. Therefore, the proposed algorithm may be set for different views through many cameras in the future to get more accurate results.

Table 2 Hyper parameter tuning for proposed models

Model	Batch Size	Drop-out	Activation Function			Optimizer	Epochs	Environment
			Convolutional Layer	FC Layer	Output Layer			
Model 1	8	0.30	Relu	Relu	Sigmoid	Adam	120	Our System and Google Colab
Model 2	8	0.50	Relu	Tanh	Sigmoid	Ada-Delta	120	Our System and Google Colab

Table 3 Outcomes for Model 1 and Model 2

Model	True positive	True negative	False positive	False negative	Training accuracy	Training loss	Validation accuracy	Validation loss
Model 1	100	94	0	6	0.9957	0.0213	0.9700	0.00178
Model 2	100	97	0	3	0.9981	0.0057	0.9850	9.6017e-05

Table 4 Time Comparison between Model 1 and Model 2

Model	Model 1		Model 2	
	Our system	Google colab	Our system	Google colab
Training time	8.51 h	34.34 min	9.36 h	37.99 min
Testing time	9.25 s	1.34 s	9.49 s	1.41 s

Fig. 4 Accuracy and loss curve w.r.t. epochs for Model 1 and Model 2

Table 5 Comparison with Existing Human Detection Approaches

Authors and Year	Techniques used	Accuracy (%)
Fu-Chun Hsu et al. [11] in 2013	HOG + SVM	65.52
	HOOF + SVM	88.48
	HOG + HOOF + SVM	86.26
Vijay and Shashikant [13] in 2015	Edgelet Features + Cascade Structure of K-Means Clustering	95.00
Suman Kumar Choudhury et al. [14] in 2018	Background Subtraction + Silhouette Orientation Histogram + Golden Ratio Based Partition + HIKSVM	98.36
Seemanthini and Manjunath [15] in 2018	Cluster Segmentation + Temporal Tracking + HOG + SVM	89.59
Aichun, Tian and Qiao [16] in 2019	Candidate Region Convolutional Neural Network	86.00
D. K. Singh et al. [17] in 2020	Background Subtraction + HOG + SVM	81.00
Proposed model 1		97.000
Proposed model 2		98.500

Fig. 5 Outcomes of Model 2 for practicing social distancing

References

1. Worldometer. COVID-19 coronavirus pandemic. In: <https://www.worldometers.info/corona-virus/>, 2020, [Accessed 10 June 2020].
2. World Health Organization (2020) Coronavirus disease (COVID-19) advice for the public. In: <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public>. [Accessed 10 June 2020].
3. Adlhoeh C, et al. (2020) Considerations relating to social distancing measures in response to the COVID-19 epidemic. European Centre for Disease Prevention and Control.
4. Singhal TA (2020) Review of Coronavirus Disease-2019 (COVID-19). Indian J Pediatr 87:281–286. <https://doi.org/10.1007/s12098-020-03263-6>
5. Singh DK, Kushwaha DS (2016) Tracking Movements of Humans in a Real-Time Surveillance Scene. In: Pant M, Deep K, Bansal J, Nagar A, Das K (eds) Proceedings of Fifth International Conference on Soft Computing for Problem Solving. Advances in Intelligent Systems and Computing. Springer, Singapore
6. Abughalieh, Karam, Shadi Alawneh (2020) Pedestrian orientation estimation using CNN and depth camera. No. 2020–01–0700. SAE technical paper.
7. Jiao L et al (2019) A survey of deep learning-based object detection. IEEE Access 7:128837–128868. <https://doi.org/10.1109/ACCESS.2019.2939201>
8. Singh DK (2019) Human action recognition in video. In: Luhach A, Singh D, Hsiung PA, Hawari K, Lingras P, Singh P (eds) Advanced informatics for computing research ICAICR 2018 communications in computer and information science. Springer
9. Karpagavalli P, Ramprasad AV (2017) An adaptive hybrid GMM for multiple human detection in crowd scenario. Multimed Tools

- Appl 76:14129–14149. <https://doi.org/10.1007/s11042-016-3777-4>
10. Viola P, Jones M (2001) Robust real-time object detection using a boosted cascade of simple features. *IJCV* 12:18–24
 11. Hsu FC, Gubbi J, Palaniswami M (2013) Human head detection using histograms of oriented optical low in low quality videos with occlusion. In: 2013, 7th International conference on signal processing and communication systems (ICSPCS), IEEE, pp 1–6
 12. Bahri H, Chouchene M, Sayadi FE et al (2019) Real-time moving human detection using HOG and Fourier descriptor based on CUDA implementation. *J Real-Time Image Proc.* <https://doi.org/10.1007/s11554-019-00935-1.6>
 13. Gaikwad V, Lokhande S (2015) Vision based pedestrian detection for advanced driver assistance. *Proc Comput Sci* 46:321–328
 14. Choudhury SK, Sa PK, Padhy RP, Sharma S, Bakshi S (2018) Improved pedestrian detection using motion segmentation and silhouette orientation. *Multimedia Tools Appl* 77(11):13075–13114
 15. Seemanthini K, Manjunath S (2018) Human detection and tracking using hog for action recognition. *Proc Comput Sci* 132:1317–1326
 16. Zhu A, Wang T, Qiao T (2019) Multiple human upper bodies detection via candidate-region convolutional neural network. *Multimedia Tools Appl* 78(12):16077–16096
 17. Singh DK, Paroothi S, Rusia MK, Ansari MA (2020) Human crowd detection for city wide surveillance. *Proc Comput Sci* 171:350–359. <https://doi.org/10.1016/j.procs.2020.04.036>
 18. Gajjar, Vandit, Ayesha Gurnani, Yash Khandhediya (2017) Human detection and tracking for video surveillance: A cognitive science approach. In: Proceedings of the IEEE International Conference on Computer Vision Workshops.
 19. Najva N, Edet Bijoy K (2016) SIFT and tensor based object detection and classification in videos using deep neural networks. *Procedia Comput Sci* 93:351–358. <https://doi.org/10.1016/j.procs.2016.07.220>
 20. Agnes SA, Anitha J, Pandian SIA et al (2020) Classification of mammogram images using multiscale all convolutional neural network (MA-CNN). *J Med Syst* 44:30. <https://doi.org/10.1007/s10916-019-1494-z>
 21. Zhang H, Hong X (2019) Recent progresses on object detection: a brief review. *Multimedia Tools Appl* 78(19):27809–27847
 22. Chahyati D, Fanany MI, Arymurthy AM (2017) Tracking people by detection using CNN features. *Procedia Comput Sci* 124:167–172. <https://doi.org/10.1016/j.procs.2017.12.143>
 23. Ansari, Mohd Ali, Dushyant Kumar Singh (2018) Review of deep learning techniques for object detection and classification. In: international conference on communication, networks and computing. Springer, Singapore.
 24. U. Ojha, U. Adhikari and D. K. Singh (2017) Image annotation using deep learning: A review. International Conference on Intelligent Computing and Control (I2C2). Coimbatore, 2017, pp. 1–5.
 25. Marquez ES, Hare JS, Niranjan M (2018) Deep cascade learning. *IEEE Trans Neural Netw Learn Syst* 29(11):5475–5485
 26. Wang D et al (2019) Daedalus: breaking non-maximum suppression in object detection via adversarial examples. *arXiv* 6:1945–1902
 27. Ansari MA, Dixit M (2017) An enhanced CBIR using HSV quantization, discrete wavelet transform and edge histogram descriptor. International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, pp. 1136–1141.
 28. INRIA Person Dataset (2018) Available:<http://pascal.inrialpes.fr/data/human/>.
 29. R. Jayaswal, J Jha (2017) A hybrid approach for image retrieval using visual descriptors, 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, 2017, pp. 1125–1130, doi: <https://doi.org/10.1109/CCAA.2017.8229965>.