Video Analyzer for Social Distancing

Shivani Salokhe
Department of Computer and Information Technology
College of Engineering, Pune
Pune, India

Yadnyi Deshpande
Department of Computer and Information Technology
College of Engineering, Pune
Pune, India

Shivani Datar
Department of Computer and Information Technology
College of Engineering, Pune
Pune, India

ABSTRACT
The use of social distance as a barrier to the spread of the infectious Coronavirus Disease has been shown to be successful (COVID-19). Individuals, on the other hand, are not accustomed to keeping track of the necessary 6-foot distance between themselves and their surroundings. The spread of this deadly disease could be delayed by an active surveillance system capable of detecting distances between individuals. The paper proposes an AI-based system for automating the task of monitoring social distancing through surveillance. Creating a video analyzer platform to help authorities ensure that social distancing standards are observed in public areas. The tool will monitor if people wear masks and social distancing is maintained in public areas.

General Terms
Machine learning, computer vision

Keywords
Social distancing, Object detection, Face detection, Image augmentation.

1. INTRODUCTION
It becomes extremely difficult for officials to keep a close eye on whether or not social distancing is being observed. Recent advances in this direction advocate for the use of intelligent devices to track and record human behaviour in public spaces. Focus is on some typical generic algorithms. As these exhibit different characteristics, by comparing them and their incorporation within the tool, the aim is to draw meaningful conclusions.

To detect and keep track of people in a video stream, the key aspects involved in the development consist of :

1.1. **Object Detection**: For detection and classification of objects in motion, three advanced techniques are considered - R-CNN, Fast R-CNN and YOLO.

1.2. **Face Detection**: To locate human faces within an image. MTCNN and DSFD face detection algorithms were referred. WHO has recommended wearing masks so as to prevent onward transmission in public areas. The aim is to also detect face masks in real-time video streams.

1.3. **Object Distance Calculation**: To calculate the distance between objects. Two techniques were taken into account : DBSCAN Clustering Algorithm and Vector based distance calculation.

This application will not only be helpful to reduce the efforts and strength required to manually check the places around, but also to save time and quick analysis.

2. MOTIVATION
Currently, many of the places around the world that have surveillance cameras installed, utilise a simple, record-and-watch-later approach. While this approach is suitable for retrospective analysis, it has zero real-time capabilities. Employing people for the task to continuously monitor live feeds and take preventative action is not a scalable solution.

In recent years, with the sheer increase in compute power, it is now possible to carry out complex image processing pipelines and techniques directly on the live footage, in real time, and create “active”, “smart” systems that can replace or be retrofitted to the aging “passive” systems of today.

When the novel coronavirus pandemic struck, it was known quite early on that social distancing would serve as the best defense against such an infectious disease, alongside wearing masks. However, in public places that were being surveilled already, there was no “active” or “smart” system that could immediately be put to use to detect if masks were being worn and social distancing norms being followed.

Given that the coronavirus may mutate, or an outbreak of such an infectious disease is possible in the future and other threats that can be seen on video are likely, it is crucial that our systems, move from being “Digital Video Recorders” (DVR) to “Decision Support Systems” (DSS).

This new generation of “active”, “smart” systems should be modular, and the modularity would make adding various modules like the ones that can detect masks, distance between people, face recognition, crime detection, arms detection and more possible.

3. THEORETICAL CONCEPTS OF IMPLEMENTATION
When humans look at an image, they immediately recognise what objects are present, where they are, and their interaction. Image processing is the method of compressing, improving, or extracting useful information from a digital image using a tool or algorithm. As a result, the main tasks of image processing are a sequence of recognition: classification, localization, and object detection, with accuracy, speed, cost, and complexity as main challenges.

3.1 Object Detection
Object detection is a form of computer vision that helps recognize objects in an image or video and locate them. Object recognition can be used to count objects in a scene with the method of identification and localization and determine and monitor their precise positions, all while accurately labeling them. To approach higher accuracy and speed, R-CNN (Region-based Convolutional Neural Network), Fast R-CNN (Fast Region-based Convolutional Neural Network), YOLOv3 (You Only Look Once) are possible algorithms.

3.1.1. **R-CNN**: CNN's have been commonly used to

categorise images. However, finding an object in an image and drawing bounding boxes around it is a difficult task. This problem is solved using R-CNN algorithms. R-CNN extracts the top 2000 area proposals from millions of ROI proposals in a picture and feeds them to a CNN model using selective search algorithms. This bypasses the issue of choosing a huge number of regions. However, since each picture has a classification of 2000 regional proposals, the network's training takes a long time. Since each test image takes about 47 seconds, it cannot be applied in real time.

3.1.2. **Fast R-CNN**: In Fast R-CNN, the input image is fed to CNN instead of the region proposals, resulting in a convolutional feature map. Using the convolutional function map, it recognises the region of proposals and warps them into squares. In training and analysis sessions, Fast R-CNN is marginally faster than R-CNN. Using the area proposals, as opposed to not using them, greatly slows down the algorithm during testing. As a result, the area proposals slow down the R-CNN algorithm by acting as bottlenecks. Detecting objects takes around 2 seconds per image, which can be slow when dealing with large real-world datasets.

3.1.3. **YOLOv3**: To overcome these problems, an algorithm which is differet from a region based algorithm is necessary. YOLO, an Incremental Improvement, uses a single convolutional network to predict bounding boxes and class probabilities for these boxes. Some important terms in YOLO:

3.1.3.1. Intersection Over Union (IOU) :

IOU can be calculated by dividing the Area of Intersection by the Area of Union of two boxes, so it must be ≥ 0 and ≤ 1. The IOU between the expected bounding box and the ground truth box is ~1 when predicting bounding boxes.

$$IOU = \frac{Area\ of\ Intersection}{Area\ of\ Union}$$

3.1.3.2. Precision & Recall :

Precision is the ratio of true positive(true predictions) (TP) and the total number of predicted positives(total predictions).

$$Precision = \frac{True\ Prediction(TP)}{\mathrm{TP} + \mathrm{FP}}$$

Recall is the ratio of true positive(true predictions) and the total of ground truth positives.

$$Recall = \frac{True\ Prediction(TP)}{\mathrm{TP} + \mathrm{FN}}$$

3.1.3.3. Average Precision (AP) and Mean Average Precision(mAP):

Precision and recall are combined in AP. The mean of AP determined for all classes is the mAP.

With the latest version 3, YOLOv3 is fast, has at par accuracy, and this makes it a very powerful object detection model.

For each bounding box, the network produces class probability and bounding box offset values. To locate the object within the image, bounding boxes with a class probability greater than a certain threshold are chosen and used. YOLO has a higher average precision than Fast RCNN for small objects. Application of Object Detection in this domain needs models to be very fast with a little compromise on accuracy, but YOLOv3 is also very accurate.

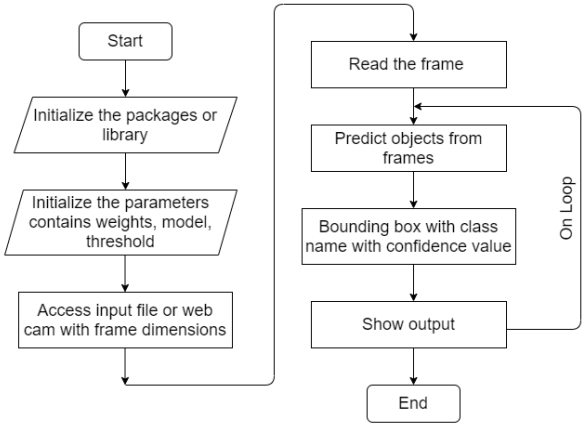

Figure 1. Detection based on YOLO

3.2 Face Detection

Face detection is the first and most important step in face recognition; it recognises human faces in digital images. A face recognition algorithm locates human faces in an image and returns face bounding boxes with high precision. MTCNN (Multi-Task Cascaded Neural Network), and DSFD have been considered (Dual Shot Face Detector).

3.2.1. **MTCNN**: MTCNN can perform both face recognition and alignment tasks at the same time. It consists of three neural network cascades: P-net, R-net, and O-net, each of which processes different combinations of image pixels. In the natural world, it is more resistant to changes in light, angle, and facial expression, and the face detection effect is improved. However, it is ineffective at detecting faces that are obscured or have poor resolution, which is in direct conflict with our requirement for mask detection and real-time analysis.

3.2.2. **DSFD**: It generates enhanced functionality from the original image using a function enhance module.

3.2.2.1. Feature Enhance Module:

The Feature Enhancement Module (FEM) can enhance existing features to make them more distinguishable and robust. FEM merges hierarchical feature maps from high and low-level output layers, but ignores current layer information.

3.2.2.2. Progressive Anchor Loss:

Two significant loss functions used in object detection are a regression loss for the face area and a classification loss for determining whether or not a face is detected. The weighted sum of classification loss (such as softmax loss) and box regression loss is typically used to measure the objective loss in detection (e.g. L2 loss).

3.2.2.3. Improved Anchor Matching:

During preparation, an anchor compensation technique is used to ensure that tiny faces fit enough anchors. It is capable of recognising faces in a variety of positions. DSFD performs well on the Wider Face dataset's fast, medium, and hard levels. It works well with augmented images, which are essential for detecting face masks.

3.2.3. Face Mask & Blurring Augmentation:

With image augmentation, the main areas of DSFD function well. For classifying whether a face is correctly masked or not, an updated ResNet50 model can be used. A diverse dataset that contains faces in different orientations and

lighting conditions is needed for this classifier to function properly in all conditions. For better outcomes, people of different ages and sexes should also be covered by the dataset. It is a difficult job to locate such a wide variety of photographs of individuals wearing a face mask. Therefore, there is a need to apply different forms of augmentation.

Because of quick motion, the blurriness could trigger face cut out of focus or random noise capture as we are going to work on real-time video frames. Hence, blurring effects are to be considered.

3.2.3.1. Motion Blur : It is the apparent streaking of moving objects in a photograph or a sequence of frames. Add blur to fast moving Shapes.

3.2.3.2. Average Blur : In this process, the central element of the image is replaced by the average of all the pixels in the kernel area.

3.2.3.3. Gaussian Blur : The image is convolved with a Gaussian filter instead of the box filter. The Gaussian filter is a low-pass filter that removes the high-frequency components..

3.3 Object Distance Calculation

To track the required 6-feet (2-meters) distance between people and their surroundings object distance calculation is necessary. It is the most critical feature of this software. An object distance calculation algorithm calculates the distance between objects detected based on specified conditions. There are two possible methods for calculating distance: Density based using Clustering Algorithms and Vector Distance based.

3.3.1. **Vector based distance calculation**: Every object is assigned coordinates with respect to its position in the image/frame. These coordinates are then transformed to give real world coordinates using a transformation matrix. The position vectors are then used to calculate distances between people using Euclidean distance. This method gives good results. But the transformational matrix depends on the region under study as it will help to determine the object sizes relative to object positions, the floor plan of that region. It is fairly difficult to calculate this transformational matrix. A critical social distance needs to be estimated to detect violations which is a fairly complex process. Much of the accuracy will depend on the camera calibration and positioning.

3.3.2. **DBSCAN Clustering Algorithm**: The algorithm operates on the concept of finding points with a predefined radius inside the circle of influence of a point. It focuses on identifying clusters. The DBSCAN algorithm uses:

3.3.2.1. minPts: For an area to be considered compact, it must have a certain (minimum) number of points clustered together (a threshold).

3.3.2.2. eps (ε): A distance measure that will be used to locate the points in the neighbourhood of any point.

3.3.2.3. Density Reachability and Density Connectivity: Density Reachability and Density Connectivity are two terms that can help us understand these parameters. When it comes to density, reachability means that a point is reachable from another if it is within a certain distance (eps) of it. Connectivity, on the other hand, uses a transitivity-based chaining approach to assess whether or not points belong to a specific cluster.

DBSCAN is easy to use and can be easily applied to the purpose at hand. Based on the social distancing norms, the radius will be determined and any other person coming inside this circle identified around a human will be flagged as a breach of norms. It's also an unsupervised algorithm, which means that training is not necessary. This would decrease a large amount of work and not change the system's accuracy. The algorithm, however, is very sensitive to the parameters mentioned. Use DBSCAN Algorithm - no training, has a significant workload and is easy to implement.

4. SYSTEM DESIGN

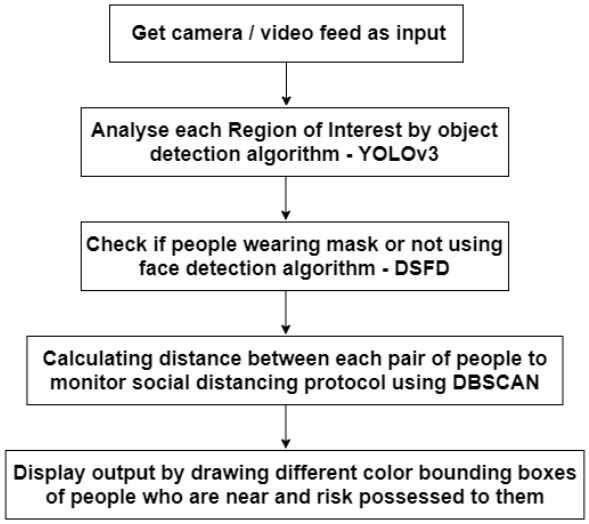

Figure 2. System Flow Diagram

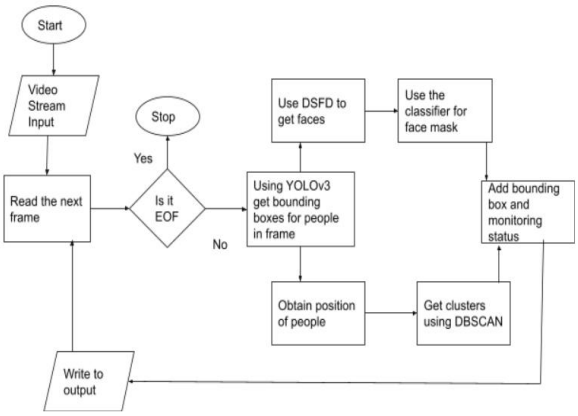

Figure 3. Pipeline Diagram

5. RESULT

5.1 Person Detection

You Only Look Once (YOLO) is a cutting-edge, real-time object detection algorithm. This project used Version 3 (trained on the COCO dataset) with a resolution of 416x416 to obtain the bounding boxes of individual people in a video frame. A resolution of 320x320 can be used to achieve a higher processing speed (lowered accuracy). A resolution of 512x512 or 608x608 can also be used to improve detection accuracy (lowered speed). The height and width parameters can be modified to adjust the resolution. YOLOv3-tiny can also be used to improve performance. However, the detection accuracy will suffer as a result.

Table 1. Person Detection performance

Method	Backbone	mAp	Time
Yolov3	Darknet	51.5	22

5.2 Face Mask Augmentation

For classifying if a face is properly masked, a slightly modified ResNet50 model (with base layers pretrained on imagenet) is used. On top of the base layers, a combination of AveragePooling2D and Dense (with dropout) layers is applied, followed by a Sigmoid and Softmax classifier.

Figure 4. Mask Augmentation

The image on the left-end is the original, while the image in the middle shows the points (green dots) where a face mask is put on it artificially. These points are discovered by manipulating facial landmarks such as the nose bridge and chin. In the observed nose bridge points, the top point is near the 1st and 2nd points. The first, last, and msiddle points in the detected chin points are near the left, right, and bottom points, respectively.

Resize and rotate the masked image to match these four reference points and paste it over the original.

5.3 Blurring Augmentation

When working on video frames, it's very likely to come across blurred faces. Hence, apply a blurring effect to some of the training data at random.

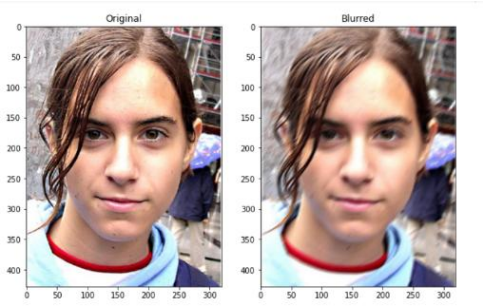

Figure 5. Blurring Augmentation

5.4 Face Detection

As can be seen, the machine recognises the individuals in the frame and their faces (if they are visible) and places green or red bounding boxes around them depending on whether they are safe or not.

Figure 5. Face Detection Output

Table 2. Face Detection performance

Method	mAp per category		
DSFD	Easy 96.6	Medium 95.7	Hard 90.4

6. CHALLENGES

Traditionally, face detection systems are designed to detect uncovered faces. There is no known dataset available of masked faces. Therefore, creating a new dataset by augmenting masks on the faces and then training the face detection model on this dataset will be a challenge. Moreover, datasets for face views from different angles are unavailable. One of the major challenges facing this project is being to run efficiently in real time for this computationally intense problem. Augmenting information on existing footage in real time is difficult. A powerful GPU is required for higher performance, i.e. closer to real time.

7. CONCLUSION AND FUTURE SCOPE

When infectious diseases such as H1N1 and COVID-19 are present, social distancing is an efficient non-pharmaceutical strategy for preventing the pandemic from worsening. The social distancing tool, when properly applied, will help minimise disease transmission and severity, reducing the strain on healthcare systems and leaving more time for government countermeasures. Furthermore, research shows that social distancing strategies and programs in response to the COVID-19 pandemic have significant economic benefits. It provides a smart solution to detect if people wear masks, track the movement of the general public so as to keep a safe distance while in public places. The tool is easy to incorporate and can be implemented at numerous places to monitor large crowds.

Till a few years before, computer vision technology was not advanced enough to carry out complex tasks in real time. But today, there is a capability of upgrading older passive recording systems. In a public setting, such as a university with numerous facilities where people gather in a classroom, a virtual fence or wall that surrounds an individual with a minimum radius can be built using a smart application. Various features such as mask detection, social distancing detection, weapon detection can be added thus making our systems smart enough to tackle unprecedented times.

8. REFERENCES

[1] Yang, Dongfang, et al. ‘A Vision-Based Social Distancing and Critical Density De- tection System for COVID-19’. ArXiv:2007.03578 [Cs, Eess], July 2020. arXiv.org, http://arxiv.org/abs/2007.03578.. .

[2] Rezaei, Mahdi, and Mohsen Azarmi. 'DeepSOCIAL: Social Distancing Monitoring and Infection Risk Assessment in COVID-19 Pandemic'. ArXiv:2008.11672 [Physics], Aug. 2020. arXiv.org, http://arxiv.org/abs/2008.11672.

[3] Ghorai, Arnab, et al. Digital Solution for Enforcing Social Distancing. SSRN Scholarly Paper, ID 3614898, Social Science Research Network, 31 May 2020. papers.ssrn.com,https://papers.ssrn.com/abstract=3614898

[4] Cristani, Marco, et al. 'The Visual Social Distancing Problem'. ArXiv:2005.04813 [Cs, Eess], May 2020. arXiv.org, http://arxiv.org/abs/2005.04813..

[5] Du, Juan. 'Understanding of Object Detection Based on CNN Family and YOLO'. Journal of Physics: Conference Series, vol. 1004, Apr. 2018, p. 012029. DOI.org (Crossref), doi:10.1088/1742-6596/1004/1/012029..

[6] Girshick, Ross. 'Fast R-CNN'. ArXiv:1504.08083 [Cs],Sept. 2015. arXiv.org, http://arxiv.org/abs/1504.08083.

[7] Redmon, Joseph, and Ali Farhadi. 'YOLOv3: An Incremental Improvement'. ArXiv:1804.02767 [Cs], Apr. 2018. arXiv.org, http://arxiv.org/abs/1804.02767.

[8] Redmon, Joseph, et al. 'You Only Look Once: Unified, Real-Time Object Detection'. ArXiv:1506.02640 [Cs], May 2016. arXiv.org, http://arxiv.org/abs/1506.02640.

[9] Degtyarev, N., and O. Seredin. 'Comparative Testing of Face Detection Algorithms'. ICISP, 2010. Semantic Scholar, doi:10.1007/978-3-642-13681-8_24.

[10] Li, Jian, et al. 'DSFD: Dual Shot Face Detector'. ArXiv:1810.10220 [Cs], Apr. 2019. arXiv.org, http://arxiv.org/abs/1810.10220.

[11] Ku, Hongchang, and Wei Dong. Face Recognition Based on MTCNN and Convolutional Neural Network. 2020. Semantic Scholar, doi:10.22606/fsp.2020.41006.

[12] Schubert, Erich, et al. 'DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN'. ACM Transactions on Database Systems, vol. 42, no. 3, July 2017, p. 19:1–19:21. August 2017, doi:10.1145/3068335.

[13] Wang, Xiang, et al. 'A Survey on Face Data Augmentation'. Neural Computing and Applications, vol. 32, no. 19, Oct. 2020, pp. 15503–31. arXiv.org, doi:10.1007/s00521-020-04748-3.

[14] Kollias, Dimitrios, et al. 'Deep Neural Network Augmentation: Generating Faces for Affect Analysis'. International Journal of Computer Vision, vol. 128, no. 5, May 2020, pp. 1455–84. DOI.org (Crossref), doi:10.1007/s11263-020-01304-3