Holistic Interpretation of Public Scenes Using Computer Vision and Temporal Graphs to Identify Social Distancing Violations

November 30, 2021

Abstract

The COVID-19 pandemic has caused an unprecedented global public health crisis. Given its inherent nature, social distancing measures are proposed as the primary strategies to curb the spread of this pandemic. Therefore, identifying situations where these protocols are violated, has implications for curtailing the spread of the disease and promoting a sustainable lifestyle. This paper proposes a novel computer vision-based system to analyze CCTV footage to provide a threat level assessment of COVID-19 spread. The system strives to holistically capture and interpret the information content of CCTV footage spanning multiple frames to recognize instances of various violations of social distancing protocols, across time and space, as well as identification of group behaviors. This functionality is achieved primarily by utilizing a temporal graph-based structure to represent the information of the CCTV footage and a strategy to holistically interpret the graph and quantify the threat level of the given scene. The individual components are tested and validated on a range of scenarios and the complete system is tested against human expert opinion. The results reflect the dependence of the threat level on people, their physical proximity, interactions, protective clothing, and group dynamics. The system performance has an accuracy of 76%, thus enabling a deployable threat monitoring system in cities, to permit normalcy and sustainability in the society.

1 Introduction

The COVID-19 pandemic is one of the largest global health catastrophes in recent history. It is a viral infection, which spreads from person to person, that causes a wide range of complications, primarily in the respiratory system [1] along with other systems [2, 3]. As per current statistics, even though the virus has a comparatively small case mortality rate, it has amassed a massive fatality count due to its high infectivity. The World Health Organization (WHO) estimates that the virus has infected around 220 million people and claimed more than 4.5 million lives as of September 2021. Despite the availability of effective vaccines against virus spread, medical complications and mortality; complete global vaccination

coverage is still far due, owing to vaccine production, distribution, and other logistic and regulatory issues. Furthermore, emerging variants cast some non-trivial obstacles for vaccine efficiency [4, 5], in addition to the existing vaccines not being one hundred percent efficient [6, 7]. Therefore, mitigating the spread of the disease through social distancing, mask-wearing, hand washing, sanitizing, and other practices of hygiene still remains indispensable by and large [8, 9] to restore normalcy whilst ensuring safety of health.

As such, non-pharmaceutical interventions (NPI) are still essential for the effective containment of the spread of COVID-19 even with vaccination roll-out. NPI techniques refer to measures that people can take, that can reduce the opportunities for the spread of the pandemic. COVID-19 spreads through droplet [10] and aerosol [11, 12] transmission and therefore, the recommended NPI for containment is primarily social distancing. Social distancing is the behavioral modification in which people minimize physical interactions with each other. The currently accepted guidelines require a minimum physical distance of 1 m between two people [13], refraining from touching potentially contaminated items/surfaces, and wearing face masks. Studies in [14, 15] have proven that maintaining interpersonal distances of more than 1m reduces the spread of COVID-19 by 75e%. Furthermore, [16, 17] have shown that avoiding physical interactions between people can drastically reduce the spread, while [18] depicts the importance of wearing masks to mitigate the spread of COVID-19.

Humans, being a social species, tend to exhibit group behaviors frequently. Therefore, even the most mindful persons may violate social distancing protocols occasionally [19, 20]. Even such occasional violation of social distancing protocols may garner a risk of contracting COVID-19 depending on the proximity or duration of the violation [21, 22]. Conversely, monitoring such violations of social distancing protocols (i.e. proximity, duration as well as the intensity of sudden events such as maskless cough or sneezing) provide vital tools for contact tracing, monitoring, and eventually pandemic control. Nevertheless, if a certain set of individuals remain in a group throughout an observation window, then whatever interaction during that observation window, has no impact on the potential of contracting COVID-19 within the group — hence commonly known as a "bio-bubble" or simply a "bubble". In essence, observing social distancing protocol violations is a task with many caveats. Thus, such monitoring is tedious to do manually, whereas automating such a task needs meticulous analysis [23]. Technology has been used in different ways to automate this monitoring process. The main two avenues of research have been (a) intrusive solutions where people are actively contributing to the measurement (by handheld devices etc.) and (b) non-intrusive solutions with zero burden on the people (which could be deployed to any situation irrespective of who is being monitored).

The first type (intrusive techniques) requires a signal to be transmitted by the people being tracked, i.e. methods of this type require some kind of active beaconing by each tracked person. The metrics extracted from the analysis of these signals can be either absolute locations of people (which could be used to calculate the distances between individuals) or relative positions (which itself is indicative of the

social distancing situation). Such a wearable device based on an oscillating magnetic field for proximity sensing to monitor social distancing to prevent COVID-19 has been presented in [24]. This system was shown to be more robust than Bluetooth sensing devices [25], especially in determining the distance threshold limit. However, it is practically difficult to deploy a solution of this type in a public space in a real-world situation. Thus, a non-intrusive solution is preferable for large-scale deployment in public spaces as the people who are being tracked are done so passively.

Research in non-intrusive techniques to monitor social distancing has led to a large body of work utilizing computer vision techniques. The major sub-tasks in those approaches are detection and tracking of people and the state of the art for both these sub-tasks are now primarily dominated by convolution neural networks (CNNs), which is a type of artificial neural networks (ANN). The detection and localization task together is achieved by region proposal networks such as RCNNs [26], Single Shot Detection (SSD) [27], and YOLO [28] which operate on individual frames. Tracking is done on a sequence of frames by classical methods such as Kalman filtering [29], particle filtering [30] or modern methods such as SORT [31] and deepSORT [32]. Most recent applications combine YOLO and deepSORT to form powerful tools which can achieve object detection and tracking in real-time.

The work in [33] is an example of a CNN framework built on the aforementioned detection and localization algorithms to detect people, calculate Euclidean distance between them and spot social distancing violations. A similar approach using YOLOv3 is performed in [34, 35] for birds-eye view (overhead) camera footage. However, such overhead viewpoints are not practically deployable in public settings. An SSD-based model is presented in [36], which also performs person detection and social distancing violation identification. In the light of a system identifying social distancing violations and temperature checks, [37] uses thermal camera images for person detection using YOLOv2. The use of multiple object detection deep learning frameworks for people identification and the calculation of social distancing violations after the perspective transform is performed in [38]. The performance is compared for each of the deep learning models Faster RCNN, SSD, and YOLO. Reference [39] utilizes the YOLOv4 model for people detection in low light instances to enforce social distancing measures. In [40], a spatio-temporal trajectory-based social distancing measurement and analysis method is proposed. This problem has been further examined in [41, 42, 43].

While various solutions proposed in the literature strives to assess the adherence to social distancing protocols, they fall short of incorporating factors such as mask-wearing, critical to the current COVID-19 pandemic. The presence or absence of a mask on a person greatly affects the efficacy of the social distancing protocols [18]. Similarly, inter-person interactions such as hugs, kisses, and handshakes are severe concerns than mere distancing amongst individuals [16, 17] as far as the person-to-person spreading of COVID-19 is concerned. The detection of mask-wearing [44]-[46] as well as the detection of dyadic interactions [47]-[50] have been explored in computer vision as isolated and distinct problems. However,

to the best of the knowledge of the authors of this paper, those factors have not been incorporated into a unified and holistic solution for detecting violations of social distancing protocols in the literature. Ignoring such factors vastly undermines the robustness of vision-based techniques to tackle the social distancing problem for COVID-19. The absence of a holistic system impedes the potential of deep learning techniques for uncovering the patterns associated with the spread of COVID-19 from person to person. Furthermore, as more medical research unfolds and reveals more information about how the COVID-19 spreads, those findings may also be incorporated into such a deep learning framework.

In this light, the system proposed in this paper analyzes the special interactions within a single frame as well as the temporal interactions manifested over multiple frames. Within a single frame, an analysis is conducted to recognize handshake interactions between people and the mask-wearing behavior of the persons. Each new interaction of a person with a new person increases the risk of spreading COVID-19, depending on the nature of violation of the social distancing protocols. On the other hand, if a certain set of people are in a certain group and they remain so until the end of observation, there is no change in the risk of spreading COVID-19. This property is also incorporated in the proposed system.

In this paper, the design, implementation and testing of a complete end-to-end system comprising of a framework to fit in different computer vision and deep learning based techniques, a representation to store the output of the deep learning models, and an interpretation technique to evaluate the threat level of a given scene are discussed. The key contributions of this paper are as follows:

- A deep learning based system to monitor social distancing violations and COVID-19 threat parameters. The system can utilize multiple computer vision modules to extract different information from the video sequence such as the number of people, their location, their physical interactions, and whether they wear masks.

- A temporal graph representation to structurally store the information extracted by the computer vision modules. In this representation, people are represented by nodes with time-varying properties for their location and behavior. The edges between people represent the interactions and social groups.

- A methodology to interpret the graph and quantify the threat level in every scene based on individual behavior, proximity and group dynamics extracted from the graph representation.

- A novel approach for dyadic human interaction detection and localization in a multiple-person setting.

2 Proposed solution

In this section, the graph-based computer vision framework proposed to quantify the risk of transmission of COVID-19 in different public scenarios is explained in detail. The input video feed from closed circuit television (CCTV) footage is first used to extract vital information such as the people, handshake interactions and face masks through computer vision models. The proposed system then quantifies the risk of transmission of COVID-19 by encoding the extracted information into a temporal graph and interpreting the graph through a function for threat of transmission developed in this paper. For a high level overview of the proposed system, we refer the reader to Fig. 1.

The system takes a video stream $V_{in}(t)$ as the input, where t denotes the frame number. In general, we assume that this video stream is captured from a CCTV system camera mounted at a desired vantage point. At each second, the camera captures several frames which depends on the frames per second (FPS). It was assumed that this frame rate is constant and known. The input $V_{in}(t)$ is a 3-dimensional matrix of the size $[H \times W \times 3]$ where $H \times W$ is the resolution of the camera and 3 corresponds to the number of color channels (Red, Green, Blue). The elements of this matrix can take non-negative integer values up to 255 (8-bit colors).

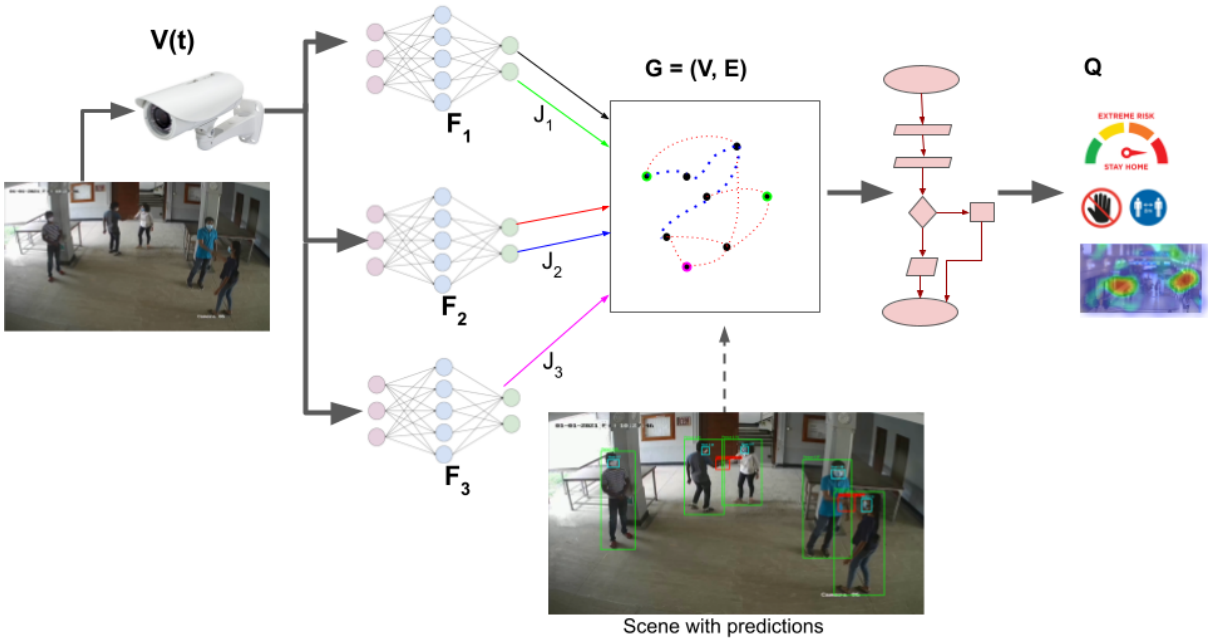


Figure 1: A high-level overview of the proposed system

The video feed, $V_{in}(t)$ was fed into a collection of functions F_i ; $i \in I$, where each F_i processes a video frame $V_{in}(t)$ and outputs different information as,

$$J_i(t) = F_i\big(V_{in}(t)\big) \tag{1}$$

where $J_i(t)$ is an output, such as the locations of the people, the handshake interactions or the presence of face masks. While the functions F_is process individual frames, processing a sequence of frames is required to analyze this information across time. Therefore, another set of stateful functions $\overline{F_i}$ were used to track the above interpretations given by F_is, through time as,

$$[S_i(t), \overline{J_i}(t)] = \overline{F_i}\Big(V_{in}(t), J_i(t), S_i(t-1)\Big) \tag{2}$$

where $S_i(t)$ is the state information and $\overline{J_i}(t)$ is the tracking interpretations based on the sequential information (i.e. considering previous frames as well). The statefulness here refers to the ability of the function to remember the calculation (e.g. location of a person) from the previous frame for the calculations for future frames.

In the above description, $I \subset \mathbb{Z}$ is the index of the set function that outputs the spatial information to detect and localize people, interactions, and face masks. The list of functions are as follows:

1. People detection (F_1) and tracking ($\overline{F}_1$)
2. Distance estimation (F_d) and group identification (F_g).
3. Identifying and localizing physical interaction (handshakes) (F_3).
4. Mask detection (F_4).

The outputs from the functions listed above (F_i's) are extractors of vital information for the social distancing violation measure. The information extracted by these functions were encoded in a graph $G = (V, E)$ where vertices V denotes the set of people in the video with extracted information embedded as vertex properties, and edges E denote the interactions between those people. Sections 2.1 - 2.5 describe the functionalities of each component of the system, which operate together to populate the graph G and Section 2.6 provides a detailed description of the information stored in the graph. Finally, the graph G was interpreted as explained in Section 2.7 to provide actionable insights based on the threat level analysis of the video being analyzed. For ease of understanding the notations used in this work are described in Table 1.

2.1 People detection and tracking

The detection and tracking task is the first step towards the social distancing violation problem. In this section, the people detection and tracking models of the proposed framework are discussed. The people in the scene were identified using the F_1 detection model and then tracked as they moved along multiple frames through the $\bar{F_1}$ tracking model. The detection model used for this purpose provides the position of the person with a bounding box, whereas the tracking model assigns unique IDs to each person and tracks them across time.

Table 1: Notations and description [JH: Remove what is not needed]

Notation	Definition
$V_{in}(t)$	Input video feed.
F_1, $\bar{F}_1$	People detection and tracking.
F_d	Distance estimation.
F_g, $\bar{F}_g$	Group identification and tracking.
F_3	Identifying and localizing physical interaction (handshakes)
F_4, $\overline{F}_4$	Mask detection and tracking.
$J_i(t)$	Output of model F_i.
$S_i(t)$	State information.
$bb_{pk}(t)$, $ID_{pk}(t)$	Bounding box encompassing person k at time t and their unique index.
$bb_{hk}(t)$, $ID_{hk}(t)$	Bounding box encompassing handshake interaction k at time t and their unique index.
$bb_{mk}(t)$, $ID_{mk}(t)$	Bounding box encompassing the face of person k at time t and their unique index.
u, v	The 2D coordinates of the center of the bounding box
h, r	The height and aspect ratio of the bounding box.
R, R'	The coordinates of the reference points in the video frame and 2 dimensional floor plane respectively.
M_T	Transformation matrix for the perspective transform from CCTV perspective to floor plane.
$s_{(i,t)}$	Standing location of person i at time t in the CCTV perspective.
$floorLocation_{(i,t)}$	Standing location of person i at time t in the floor plane.
$dist_{(i,j,t)}$	Distance between a pair of people i and j at time t.
P_i	Person i in the frame.
$G(t)$	Graph at time t
$V(t)$	Vertices of graph G at time t given by $\{v_1(t), v_2(t), \ldots, v_n(t)\}$, each vertex corresponding to person P_i with the vertex parameters embedded.
$E(t)$	Edges of graph G at time t given by $\{e_{1,1}(t), e_{1,2}(t), \ldots, e_{i,j}(t), \ldots, e_{n,n}(t)\}$, where $e_{i,j}$ is the edge between person(vertex) i and j.
$T(t)$	Threat level of frame at time t
$\mathbb{P} = \{p_d, p_h\}$	Primary parameters- set of parameters that have a direct attribute to COVID-19 transmission.
$\mathbb{Q} = \{q_g, q_m\}$	Secondary parameters- set of parameters that are relevant to COVID-19 transmission when two individuals are in close proximity.
ϵ_j	Tuneable parameter dictating influence of parameter q_j on overall threat level.

The detection model outputs a time-varying vector that represents the spatial localization information of people. It is given as $J_1(t) = \{bb_{p1}(t), bb_{p2}(t), \ldots, bb_{pk}(t), \ldots, bb_{pn}(t)\}$ where n is a the number of bounding boxes representing a person at time t and $bb_{pk}(t) = (u, v, r, h, c_p)$, is a 5 tuple that represents the bounding box encompassing a detected person at time t. In $bb_{pk}(t)$, variables u and v are the 2D coordinates of the center of the bounding box, r is the aspect ratio of the bounding box, h is the height of the bounding box and c_p is the confidence level of the detection as shown in Fig. 2. The tracking model outputs a vector of the same size $\overline{J_1}(t) = \{ID_{p1}(t), ID_{p2}(t), \ldots, ID_{pk}(t), \ldots, ID_{pn}(t)\}$, where $ID_{pk}(t)$ is the ID assigned to the bounding box which can be either a positive integer or an unassigned (null) value.

End-to-end learning models learn complex features directly from the input without any explicit intermediate stages and predict the output based on the learned features. While there are many such models, in this paper the YOLO network [51] for people detection (F_1) and the DeepSORT algorithm [32] for tracking ($\bar{F}_1$) were utilized, given the robustness and real-time prediction capability. Given an image,

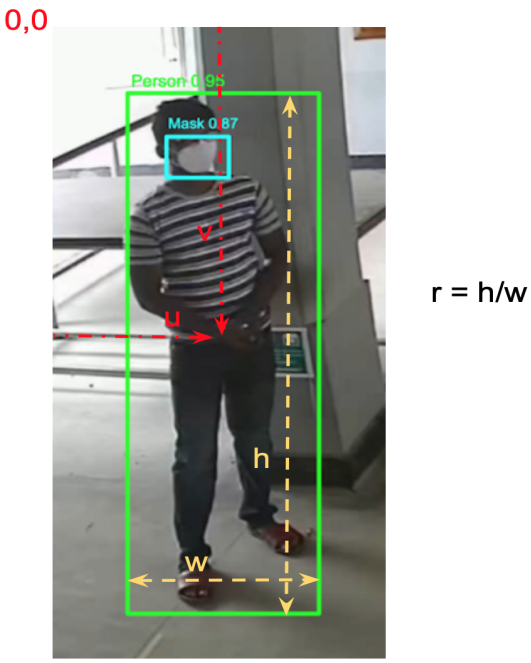


Figure 2: The parameters of the prediction for the bounding boxes

the YOLO network predicts the bounding boxes of multiple predefined object classes present in the scene. Next, after performing non-max suppression [52] and filtering the bounding boxes corresponding to people, the output J_1 is obtained. The DeepSORT algorithm then assigns indexes $\overline{J_1}$ to these identified bounding boxes based on the Mahalanobis distance and the cosine similarity of the deep appearance descriptor between the predicted Kalman state and the new bounding box. The publicly available weights trained using the COCO dataset [53] was used to initialize the weights of the YOLO model, whereas the weights trained using the MOT dataset [54] was used to initialize the DeepSORT model.

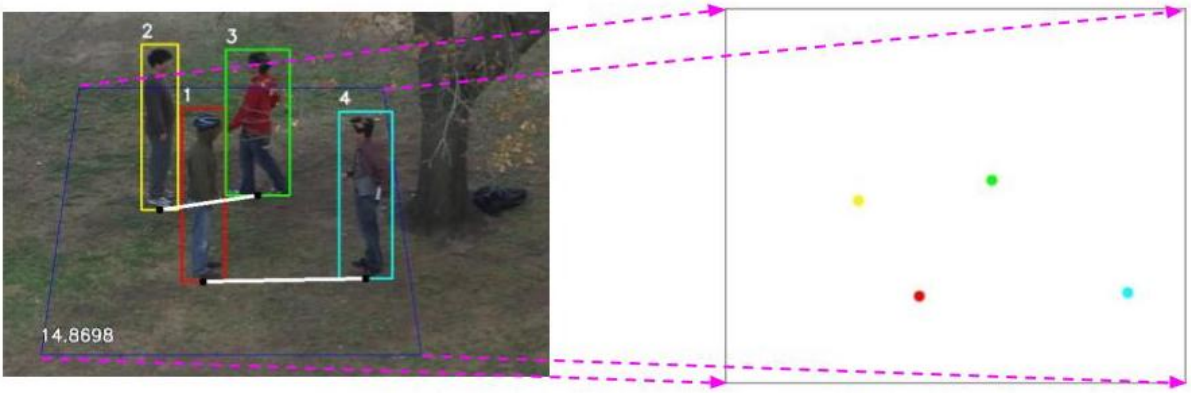


Figure 3: Perspective transformation in action. The right frame is a visualization of how a camera-captured scene (left) is projected to the 'floor plane' after perspective transform. Please note that the trapezoidal floor is being transformed into a square in this scene.

2.2 Distance estimation

The identified people using the detection and tracking models need to be inspected for social distancing violations. In this section the method used for distance estimation between the people identified is discussed. The distance between people was estimated in three steps: first by identifying the standing locations of the people in the video, then by performing perspective transform and finally measuring the Euclidean distance between them [55].

First, the standing locations of the people $s_{(i,t)}$ (denoted by thick black dots in Fig. 3) was derived from the bounding boxes detected in the previous step ($bb_{pk}(t) = (u, v, r, h, c_p)$) as follows

$$s_{(i,t)} = (u, v + 0.5h) \tag{3}$$

Then the standing locations were transferred from a overhead wall mount camera viewpoint to a two-dimensional bird's eye viewpoint through perspective transform. The transformation matrix M_T required to perform the perspective transform was calculated as follows,

$$\begin{aligned} R' &= M_T R \\ R'R^T &= M_T RR^T \\ R'R^T(RR^T)^{-1} &= M_T(RR^T)(RR^T)^{-1} \\ M_T &= R'R^T(RR^T)^{-1} \end{aligned} \tag{4}$$

where $R = \{(x_i, y_i) : i \in \{1, 2, 3, 4\}\}$ is a 2×4 matrix which represents the coordinates of the reference points in the video frame (refer blue trapezoid in Fig. 3 - Left) and $R' = \{(x'_i, y'_i) : i \in \{1, 2, 3, 4\}\}$ is another 2×4 matrix which represents the coordinates in the two-dimensional plane (bird's eye view). We refer to this two-dimensional plane as the ''floor plane'' (refer Fig. 3 - Right). Then, the calculated standing location of the people were projected as points to the two-dimensional floor plane using the M_T transformation matrix as,

$$floorLocation_{(i,t)} = M_T s_{(i,t)} \tag{5}$$

where $s_{(i,t)}$ are the input coordinates from (3) and $floorLocation_{(i,t)}$ are the output coordinates on the floor plane. Finally, the distances between each pair of people i and j were calculated as,

$$dist_{(i,j,t)} = ||floorLocation_{(i,t)} - floorLocation_{(j,t)}|| \tag{6}$$

Since the detected bounding boxes of people cannot be directly used to estimate distances between people due to the overhead camera viewing angle, the estimation is performed after perspective transform.

The transform is performed based on the following assumptions. These assumptions hold for most of the scenes with a CCTV camera (including indoor premises of buildings, roads and footpaths).

1. All the people are on the same plane.
2. The camera is not a fisheye-like camera.
3. The camera is placed at an overhead level.

2.3 Group identification

Identifying the groups of people is an important design goal of the proposed solution, since social distancing violations within a group can be neglected. The group identification model discussed in this section utilizes the people detection, tracking and distance estimation models introduced in Sections 2.1 and 2.2. This was achieved by two algorithms F_g and $\overline{F}_g$. F_g was run on the information from individual frames while $\overline{F}_g$ analyzed the results from F_g across time to properly deduce which people fall into groups based on sustained physical proximity.

Given a frame $V_{in}(t)$, first a matrix $M_1(t)$ is created based on the distances calculated between people which is referred to as the distance matrix. Then, the affinity matrix $M_2(t)$ was calculated as follows,

$$M_2 = \exp\left(-\alpha M_1\right) \tag{7}$$

where α is an input parameter that is used to introduce a scale for the camera and scene pair. Then, clustering was performed on M_2 to split the people into clusters.

$$clusters = spectral\_clustering(M_2) \tag{8}$$

The group identification model considered a person as being part of a group even if they are close to at least one person of the group. This is motivated by the way people walk/talk in groups. Even though usual clustering algorithms try to minimize the distance of individual elements with the center of the clusters, that is not how humans behave. Therefore, the spectral clustering of affinity matrices [56] was used to deliver this result. However, human behavior cannot be analyzed from individual frames. Hence, a temporal analysis of the clusters was introduced to find the actual groups of people with a time threshold τ used for this measurement. The high-level idea is that a group is detected only if it sustains for a time period τ.

Consider people $P_i \in P$ being clustered from a video frame at time t as follows.

$$cluster\_id(P_i, t) \longleftarrow spectral\_clustering(M_2(t)) \tag{9}$$

$$cluster\_id(P_i, t) = cluster\_id(P_j, t) \qquad \text{if} \quad \exists \quad t_0 \text{ s.t} \quad t_0 \leq t \leq t_0 + \tau \tag{10}$$

where P_i and P_j were considered to be in the same social group as per Eq. (10). Social distancing violations between the people in the same social group was ignored in the proposed system as justified in Section 1. For cases with a few people in the frame, a simplified algorithm using naive thresholding of inter personal distance violation occurrences was used in place of spectral clustering.

2.4 Handshake interaction detection

Physical interactions such as handshakes and hugs are of a higher threat level when transmission is considered in the case of COVID-19. A major design goal of the proposed system was to monitor such activity and keep track of the people involved for contact tracing purposes. The interaction detection was done in three steps:

- Detecting and localizing physical interactions using the model in [50].
- Identifying the people involved.

In this work, the physical interactions detected were handshakes, given by $J_3(t)$. The identification of handshakes was performed using an end-to-end deep learning CNN model. The YOLO object detection model re-purposed for handshake detection combined with DeepSORT was used to detect, localize and track the handshakes. The YOLO model performed a prediction of multiple bounding boxes for handshakes with their confidence values. This was then preceded by non-max suppression to identify the most probable bounding boxes for handshakes as a time-varying output (frame level) providing spatial localization as $J_3(t) = \{bb_{h1}(t), bb_{h2}(t), \ldots, bb_{hn}(t)\}$, where n is the number of handshakes detected in the scene. Each bounding box $bb_{hk}(t)$ is of the standard YOLO format specified before in Section 2.1.

Human interaction is an action in space and time. However, the YOLO model performs detection on individual frames. Since an actual event like a handshake persists without intermittency, the interactions were considered as a sequence of frames. Therefore, DeepSORT algorithm was used to assign indexes to each handshake and reduce the loss of detection in a handshake sequence which gives an output $\overline{J}_3(t) = \{ID_{h1}(t), ID_{h2}(t), \ldots, ID_{hn}(t)\}$, where $ID_{hk}(t)$ is the ID assigned to the bounding box. This maintained consistency in detecting a handshake without switching indexes between each frame.

Finally, since the detection and localization model is for handshakes alone, it is necessary to identify the people involved in the handshake interactions. This was performed by calculating the intersection over union (IoU) for the handshake and person bounding boxes. The two bounding boxes of the person's with the maximum IoU for a given handshake is assigned as the two individuals involved in the handshake. The calculation of IoU for two bounding boxes is the area of overlap divided by the area of union as shown in Fig. 4.

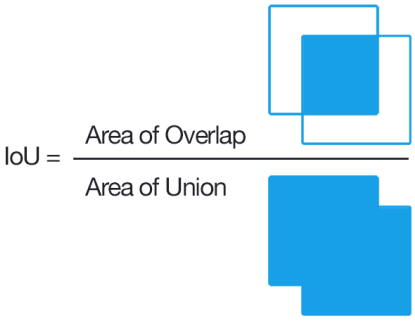


Figure 4: Calculation of IOU for bounding boxes

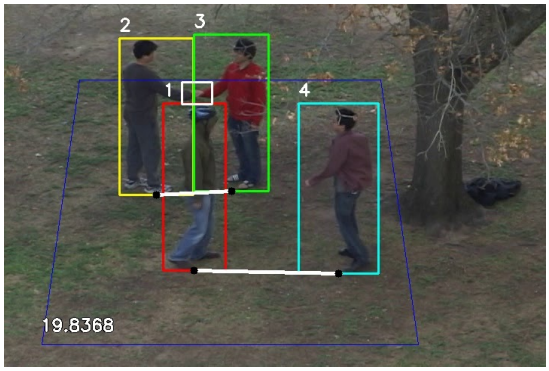
2.5 Mask detection

The wearing of masks is a crucial factor in the case of COVID-19 due to the spreading of the virus by means of aerosol transmission. In this section the model used for the detection of the presence/absence of masks is described. The mask recognition stage of the framework involves identifying and tracking the presence (or absence) of the masks. The model used for this purpose provides both the bounding box of the face and the confidence level of the presence of mask. Similar to the prior object detection models, this model outputs a time-varying vector that represents the spatial localization information of faces given as $J_4(t) = \{bb_{m1}(t), bb_{m2}(t), \ldots, bb_{mk}(t), \ldots, bb_{mn}(t)\}$ where n is a the number of face bounding boxes at time t and $bb_{mk}(t) = (u, v, r, h, c_m)$ is a 5 tuple which represents the bounding box encompassing a detected face at time t. u, v, r, and h have the same definitions as in Section 2.1. The confidence $c_m = (c_{mask}, c_{nomask})$ is a 2 tuple where $c_{mask} \in [0, 1]$ shows the probability of the existence of a mask and $c_{nomask} \in [0, 1]$ is the probability of the absence of a mask. Similar to the Section 2.1, the tracking model outputs a vector of the same size $\overline{J}_4(t) = \{ID_{m1}(t), ID_{m2}(t), \ldots, ID_{mk}(t), \ldots, ID_{mn}(t)\}$ containing unique IDs for each bounding box at time t.

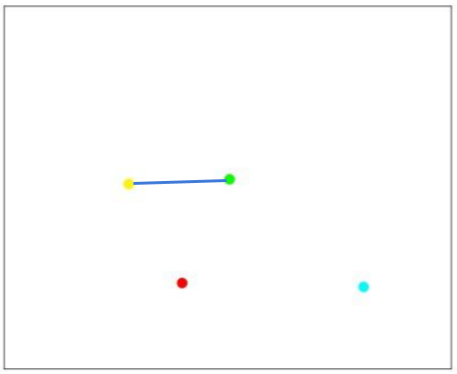
Similar to Section 2.1, the YOLO network was utilized for mask detection and the DeepSORT algorithm was utilized for tracking the masks across frames. The YOLO model was first initialized with the pre-trained COCO weights, and then retrained using the images from the Moxa3k dataset [44] as well as the UT and UOP datasets which were labeled for mask detection. The DeepSORT model used the weights trained using the MOT dataset [57] for initialization.

2.6 Graph representation

The information extracted using different models in Section 2.1 - 2.5 need to be combined in order to provide meaningful insights on the threat level of the given scene. In order to do this, the information is encoded into a graph structure for processing. This section describes how the graph structure is modelled using the different outputs from the models for interpretation.



(a) Bounding boxes for people and handshake



(b) Corresponding graph representation

Figure 5: Graph representation figure

The information retrieved from the video is stored as a time-varying graph $G(t)$ given by,

$$G(t) = \big(V(t), E(t)\big) \tag{11}$$

and

$$V(t) = \{v_1(t), v_2(t), \ldots, v_n(t)\} \tag{12}$$

$$E(t) = \{e_{1,1}(t), e_{1,2}(t), \ldots, e_{i,j}(t), \ldots, e_{n,n}(t)\} \tag{13}$$

where, $V(t)$ is the set of vertices and $E(t)$ is the set of edges between these vertices at time t. Each person P_i is denoted by a vertex $v_i(t)$ which contains the features representing the person extracted from the video as time-varying vertex parameters. The vertex $v_i(t)$ is given by,

$$v_i(t) = [\, location_i(t),\ mask_i(t),\ group_i(t)\,] \tag{14}$$

where $location_i(t) = (x_i(t), y_i(t))$ is a 2-tuple that represents the position of the person P_i at time t obtained through perspective transform to a bird's-eye view position on a 2D plane (refer Section 2.2). $mask_i(t) = c_m$ is 2-tuple which shows the confidence level that a person P_i is wearing a mask at time t. This information is extracted from $bb_{mi}(t)$ depending on the index $ID_{mi}(t)$ (refer Section 2.5). $group_i(t)$ is a matrix that represents the probability that two people belong to the same group (refer Section 2.3). The edge $e_{i,j}(t)$ is a binary value (0/1) that represents the presence (denoted by 1) or absence (denoted by 0) of an interaction between person P_i and P_j at time t (refer Section 2.4). $E(t)$ is stored as a sparsely filled adjacency matrix with null values for instances where interactions are not detected.

2.7 Threat quantification

The information extracted from the models described in the proposed system in Sections 2.1 - 2.6 need to be processed from the created temporal graph, in order to provide a quantifiable metric which

denotes the risk of transmission for the given scene/video. In this section, the detailed derivation of the threat level function which quantifies the threat of the given scene is described in detail.

The set of parameters that contribute to the spread of COVID-19 are listed in Table 2. The parameters are categorised into two as primary and secondary parameters discussed using the threat level function later on in this section. The threat level T was calculated for every frame at time t as follows,

$$T(t) = \sum_{(v_1, v_2 \in V)} T_{v_1, v_2}(t) \tag{15}$$

$$T_{v_1, v_2}(t) = \sum_{p_i \in \mathbb{P}} p_i(v_1, v_2) \times \prod_{q_j \in \mathbb{Q}} \epsilon_j - q_j(v_1, v_2) \tag{16}$$

Table 2: Parameters used in threat quantification

Set	**Notation**	**Description**
$\mathbb{P}$	p_d	Distance between people
	p_h	Handshake interactions between people
$\mathbb{Q}$	q_g	People belonging to the same group
	q_m	People wearing masks

$\mathbb{P} = \{p_h, p_d\}$ is the set of parameters that directly attributes to the transmission of COVID-19 from one person to another. This includes the distance between people and the handshake interactions. As distance between people (people coming close) and their interactions (handshakes) play a primary role in the COVID-19 virus transmission, these values were first considered as the primary parameters $\mathbb{P}$. The probability of two people shaking hands p_h and the probability of them coming extremely close p_d were represented as scalar values in the range $[0, 1]$ where 1 represents a high probability of occurrence [1].

$\mathbb{Q} = \{q_m, q_g\}$ is the set of secondary parameters which are relevant only when 2 people are in close proximity, and in such a case these parameters can increase or decrease the probability of COVID-19 transmission accordingly. This includes whether people are wearing masks or not - since two people not wearing masks is irrelevant if they are far apart, and whether the persons belong to the same group. First, the mask-wearing probability q_m was used to quantify the effect of masks in transmission. Furthermore, people belonging to the same group (q_g) have a similar effect on transmission, since it is assumed that the disease spread between them does not increase depending on what is happening in the video frame (it is more likely they were both infected or not, even before coming into the frame). The values of q_j are in the range $[0, 1]$. $\epsilon_j \geq 1$ is used as a tuneable parameter which dictates the influence of a particular parameter q_j on the overall threat level. A higher ϵ_j values give a lower significance to the corresponding q_j in calculating the total threat $T(t)$.

[1] For the distance probability, $1m$ is used as the threshold distance for being extremely close in this study

By substituting the parameters and setting $\epsilon_m = 2.0$, $\epsilon_g = 1.0$, the equation was rewritten as follows,

$$T_{v_1,v_2}(t) = (p_h + p_d)\,(2.0 - q_m)\,(1.0 - q_g) \tag{17}$$

When analyzing the threat equation in Eq. (16), it can be noted that when the secondary parameter probabilities decrease (i.e. q_j), the effect of the multiplicative term $(\epsilon_j - q_j)$ is higher. This implies that, the effect of the primary parameters p_j to the threat of the given scene are compounded when the two persons have worsening secondary parameters (i.e. are not wearing masks or when they are of different groups). It can also be observed that (17) does not carry any terms with the $p_d p_h$ product. This could be intuitively understood, because shaking hands require them to be physically close and thus, incorporating this term is redundant. While (17) is tuned for the implemented system, the generic form (16) can incorporate any number of parameters being extracted from a video scene.

3 Evaluation

In this section we discuss the methodology used to evaluate the system. The proposed solution was executed on a chosen set of datasets as the input and the results were evaluated using different metrics. The followings subsections describe the datasets, the metrics, and the evaluation execution process in detail.

3.1 Datasets

To evaluate the performance of the individual components of the system (person detection, activity detection and mask recognition) existing datasets MOT [54, 57, 58] and UT-interaction [59] were chosen. However, there are no existing datasets to perform a holistic analysis. Thus, in order to analyze handshake interactions, a new dataset was created from the University of Peradeniya, Sri Lanka premises which is referred to as the UOP dataset.

The **multiple object tracking (MOT)** datasets are a set of image sequences with annotations for people localization and people IDs. Three datasets [54, 57, 58] were used to evaluate the capability of an algorithm to uniquely identify and track a person through a video.

The **UT-interaction** dataset comprises of twenty video sequences of human interactions in a two or four people setting. The actions in the dataset include handshake, punch, point, kick, push and hug. Multiple scenes from the UT-interaction dataset [59, 60] were used to evaluate the capability of the handshake detection algorithm. For the purpose of action localisation, a ground truth for this dataset was created by annotating them since the original ground truth is a bounding box surrounding the two actors and not the action.

The **UOP dataset** is a collection of ten video sequences which were collected from the University

of Peradeniya premises by enacting a scene with human interactions. These videos were recorded by a wall-mounted CCTV camera in the university corridor and waiting area. The ground truth for this dataset was annotated manually for training and evaluation.

3.2 Evaluation metrics

The outputs were evaluated on the given datasets based on two evaluation metrics namely, the average precision (AP) and the mean average precision (mAP). mAP is the key metric used in evaluating detector performance in prominent object detection tasks such as PASCAL VOC challenge [61], COCO detection challenge [53] and the Google Open Images dataset competition [62].

The average precision (AP) is the precision value averaged across different recall values between 0 and 1 [63]. The AP is then calculated using Eq. (18). This was computed as the area under the curve (AUC) of the precision vs recall curve, plotted as a function of the confidence threshold of detection with a fixed intersection over union (IoU) for the bounding box threshold [64]. This IoU threshold is usually maintained at 0.5 in object detection tasks. For multi-class problems, the AP values for all class are averaged to obtain the Mean average precision (mAP) for the detector.

$$AP = \int_0^1 p(r)dr \tag{18}$$

3.3 Model evaluation

3.3.1 People detection

The people detection component used here is of the YOLO network which is a well established detector. Hence, no modifications were introduced to this segment of the detector. The YOLOv4 model which was used here is extensively compared in terms of frame rate and mAP in [47]. The mAP value for YOLO's performance is well established over 65%.

3.3.2 Group identification

The group identification component was evaluated using the existing MOT datasets. Since the ground truth for the datasets considered in this work do not contain the group annotated information, an alternative methodology was required for evaluation. For this purpose, visual inspection of frames was used to determine if two individuals belonged to the same group in a given frame.

3.3.3 Interaction detection

The evaluation of handshake interaction detection component requires the localization information of actions. However, the annotations of ground truth for the UT-interaction dataset focuses only the

actors. Thus the UT-interaction dataset was re-annotated together with the UOP dataset, where 17 video sequences from the UT-interaction dataset and five videos from the UOP dataset were used for the training process.

The training phase of YOLO for handshake detection was done as follows. First, the YOLO model was pretrained on the Imagenet [65] dataset. Then it was retrained to detect hands using the open images dataset [62]. This model was then retrained on the UOP dataset for handshakes using transfer learning to better the detection accuracy.

Since interaction detection is a single-class detection problem, the Average Precision (AP), which is the most versatile metric used in single-class object detection, was used for evaluation. A detection is considered true setting an IoU threshold of 0.5 or greater.

3.3.4 Mask detection

Similar to interaction detection, the mask detection component requires localized information of masks. Thus, the UT-interaction dataset was re-annotated. However, this dataset only consists of unmasked faces, and as such the annotated UOP dataset was used together with the UT-interaction dataset to train and evaluate the mask detection component. The 17 videos from the UT-interaction dataset and the 5 videos from the UOP dataset were used for training. The dataset was annotated with the 2 class information, namely; masked and unmasked faces in frames, where the faces were visible and the presence of mask can be interpreted by a human.

The mask detection model was evaluated using both the AP and mAP measures. First, the models ability to localize the faces was determined by measuring the AP of the localization component of the models disregarding the class labels. The AP (average precision) is measured with a fixed IoU threshold of 0.5.

Next, the performance of the model in terms of both the localization and the accuracy was determined by the Mean Average Precision (mAP) value. Note that, since both the classes correspond to the same object (i.e. faces), this 2-metric evaluation process helps us identify the specific shortcomings of the model considered. For instance, a high AP and a low mAP shows poor mask detection (classification), whereas a high accuracy and low mAP denotes poor face localization.

3.3.5 Graph interpretation - Threat level quantification

The threat level quantification algorithm was tested on the three datasets mentioned earlier. Since there are no publicly available ground truth for videos for this parameter, the results of the algorithm were evaluated by comparison with expert human input. For this purpose, 462 samples of frame pairs from video sequences were chosen. The system was then evaluated by observing the increment/decrement of the inferred threat level $T(t)$ and comparing the results with the expert human input. The performance

of the full system is evaluated using accuracy, precision and recall.

The expert responses were obtained by showing a pair of frames and asking if the threat for COVID-19 spread has increased or decreased from the first frame to the second. Since a high disparity in identifying the impact of COVID-19 spread can exist amongst human experts in certain instances, a ground truth cannot be established for such pairs of frames. To identify such instances, a thresholding minimum majority required to establish ground truth was set as 70% and all frame pairs with a higher disparity (i.e. less than 70% majority) for any given choice (threat increased/decreased) were removed. In the evaluation conducted, 5 such frame pairs were identified and removed. One such frame pair is shown in Fig. 6 to conclude this factor. As it can be observed, it is difficult to assess the change in threat for COVID-19 spread (whether it increases or decreases) across these two frames.

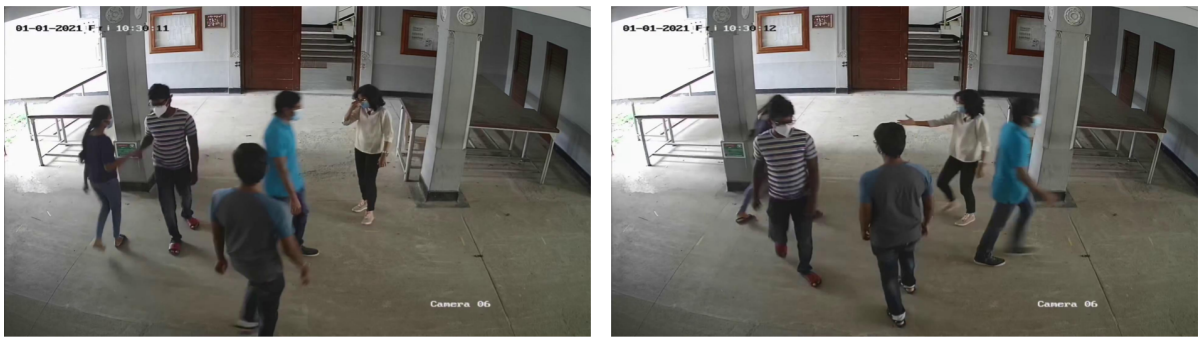


Figure 6: A pair of frames removed from full system evaluation due to disparity in human expert responses.

4 Results and Discussion

The proposed system was implemented using the Python programming language, and Tensorflow and OpenCV[2] libraries. The system is deployed on a high performance computing server with NVIDIA GPU. The output of each component of the system as well as the final output of the entire system are discussed below.

4.1 People detection and tracking

The results shown in Fig. 7 are indicative of the performance of the human detection and tracking segment of the proposed system. The first row shows a sequence of frames where people are detected properly and tracked with unique IDs. However, the model fails to perform satisfactorily in specific scenarios. The bottom row gives examples for the cases the model can fail. From left, (1) a person is not being identified because of occlusion, (2) the identified bounding box is smaller than the person due to

[2]https://github.com/pdncovid/covid-people-graph-public

Figure 7: Results of people detection. Top row shows the cases where people detection model is successful. The bottom row shows instances where people detection is erroneous. Undetected people are marked by the purple oval. The green bounding box of the bottom row middle image does not span the full person.

occlusion and (3) a person going undetected due to the lack of contrast with the background. The model has an mAP = 65% for the people detection task.

As observed in Fig. 7 a given frame from the output consists of multiple markings. The blue quadrilateral on the ground is the reference markings used for perspective transformation. The people detected are identified by uniquely colored rectangular bounding boxes. The location of each person in the 2D plane is marked using a black dot on the bottom edge of the respective bounding box. The threat level for the given frame is the numerical value displayed in the frame. Further details of the relevant markings will be discussed in the subsequent sections.

4.2 Distance estimation

A scene consisting of four people from UTI dataset in considered in Fig. 8 to show how distance between people contributes to the threat level. The distance between people is given by the distance activity matrix shown beside each frame in Fig. 8. Each element (square) in the activity matrix denotes the proximity between the person IDs corresponding to the row and column indices. The color changes to a warmer shade towards yellow when the people are closer, and becomes a colder shade towards blue when they are farther away.

Considering the frames in Fig. 8 the person ID 2 and 3 can be observed to be closer in the second frame than the first frame. This gives rise to a higher contribution to the threat level between them in the second frame and a lower contribution to the threat level in the first frame. This is seen in the

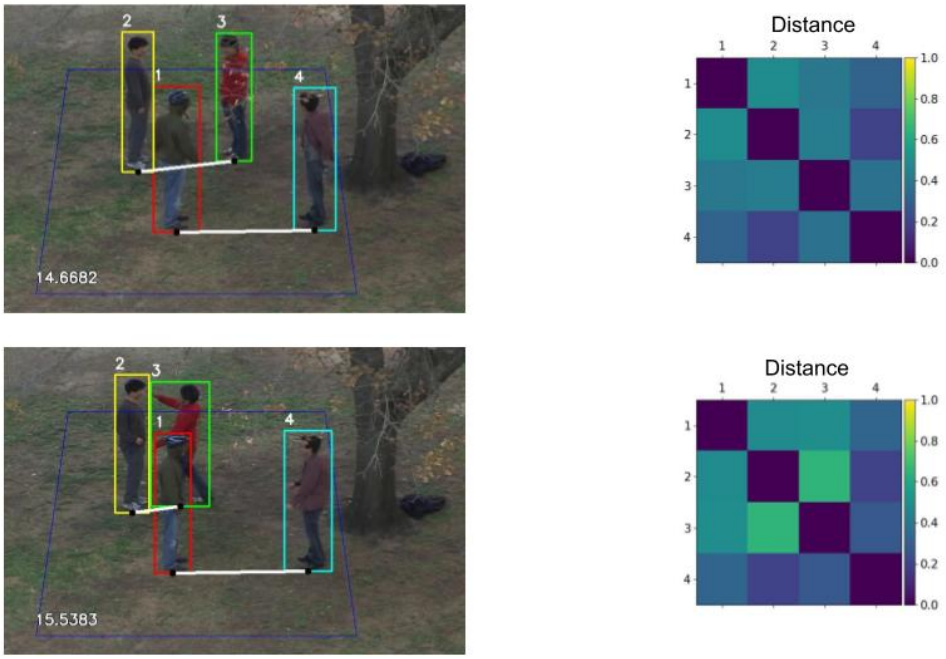


Figure 8: Distance results

distance activity matrix by the blue shade turning to cyan indicating closer proximity between those persons. The reader's attention is drawn to the threat level shown in each frame. As it can be observed, when the distance activity matrix lightens up, the threat level has also risen.

4.3 Group identification

The results for a few frames for the group identification model is shown in Fig. 9. An example from the UTI dataset and Oxford towncenter datasets each is shown here. The frames with the persons detected is shown on the left and the group activity matrices showing the group characteristic is shown on the right. If two people are of the same group, the group activity matrix element corresponding to the row and column of the IDs of these two persons is shown in yellow and otherwise it is shown in blue. The people of the same group are also joined by a white line in the original frame to show this.

4.4 Handshake detection

The performance of the system in detecting handshake interactions is shown in Fig. 10. Here two frames, one each from the UTI and UOP datasets are shown. The visualization of handshake interactions is the same as of the group activity matrices. The interaction activity matrix element corresponding to the row and column of the IDs of the two persons involved in the handshake is denoted by yellow, whereas if there is no handshake, it is depicted in blue. The handshake interaction detection and localization model performance is quantified by the AP evaluated on the UTI and UOP datasets separately and is tabulated in Table 3. The UTI dataset was tested on 3 videos with 418 frames and the UOP dataset was

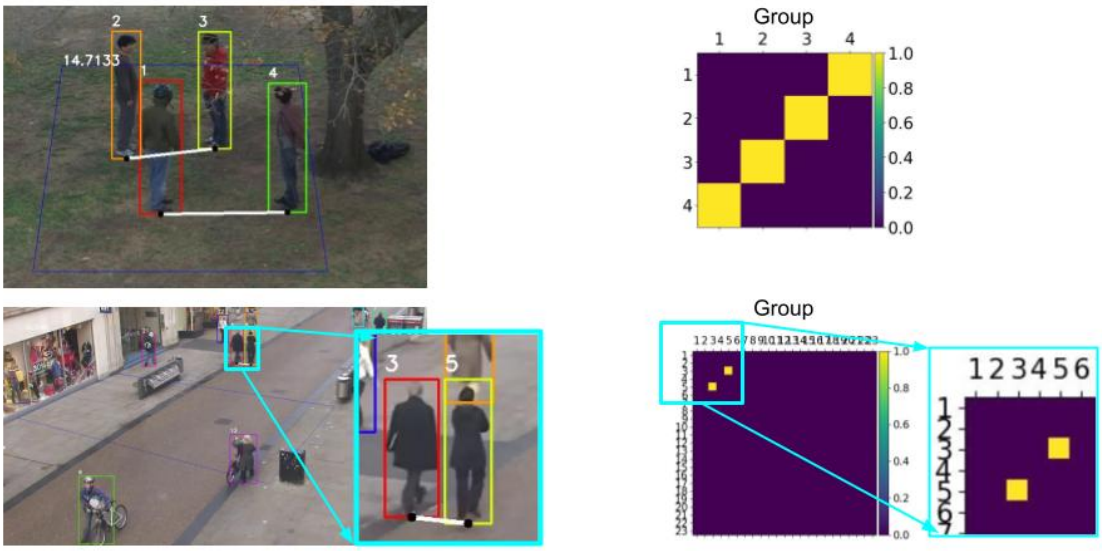
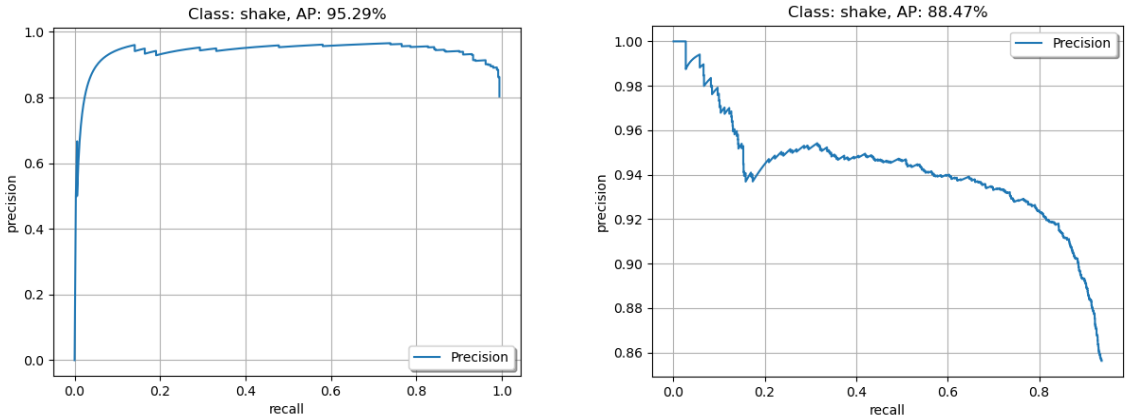


Figure 9: Group identification. Left: video frames where groups of people are denoted by white lines connecting individuals. Right: group activity matrices showing people belonging to the same group by yellow and else blue.

tested on 5 videos with 2786 frames. The precision vs recall curves for the UTI dataset and the UOP dataset are shown in Fig. 11. A detailed analysis of the results are given in [50].



(a) UTI dataset

(b) UOP dataset

Figure 11: Precision vs Recall curves for handshake detector for UTI and UOP dataset.

Table 3: Performance metrics of handshake detection

Dataset	**AP**/%
UT-interaction	95.29
UOP	88.47

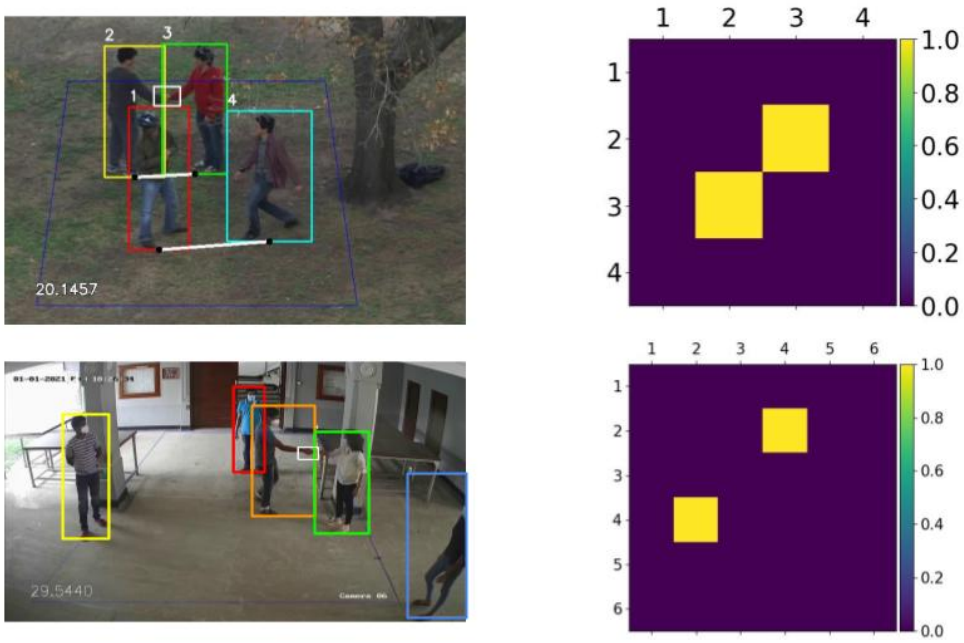


Figure 10: Handshake detection. Left: Sample frame denoting the detected people and white boxes denoting handshakes. Right: Group activity matrix.

4.5 Mask detection

Fig. 12 [Suren: TODO] shows the performance of the system in detecting the presence/absence of masks. One example from the UTI, UOP, and the Moxa3K datasets is shown. Overall the system perform well while dealing with high resolution images (Moxa3K). However, as the resolution drop (UTI / UOP), the efficacy reduces drastically. This can be observed from the Table 4 [Suren: TODO] which lists the numerical evaluation metrics (AP and mAP) on different datasets.

Table 4: Performance metrics of the mask detection.

Dataset	AP /%	mAP /%
UT-interaction	AP	mAP
UOP	AP	mAP
Moxa3K	AP	mAP

4.6 Threat level assessment (end-to-end system)

To evaluate the proposed system performance, the threat level metric provided for each frame of a given scene is evaluated across multiple frames. The successful output of this value is evaluated by the full system for both datasets UTI (Figures 13, 14 and 15), and Oxford (Fig. 16). It should be noted that it is not the absolute values of the threat level that are significant but the increments or decrements between the frames.

Considering Figures 13, 14 and 15 it can be observed that the threat level increases from top to

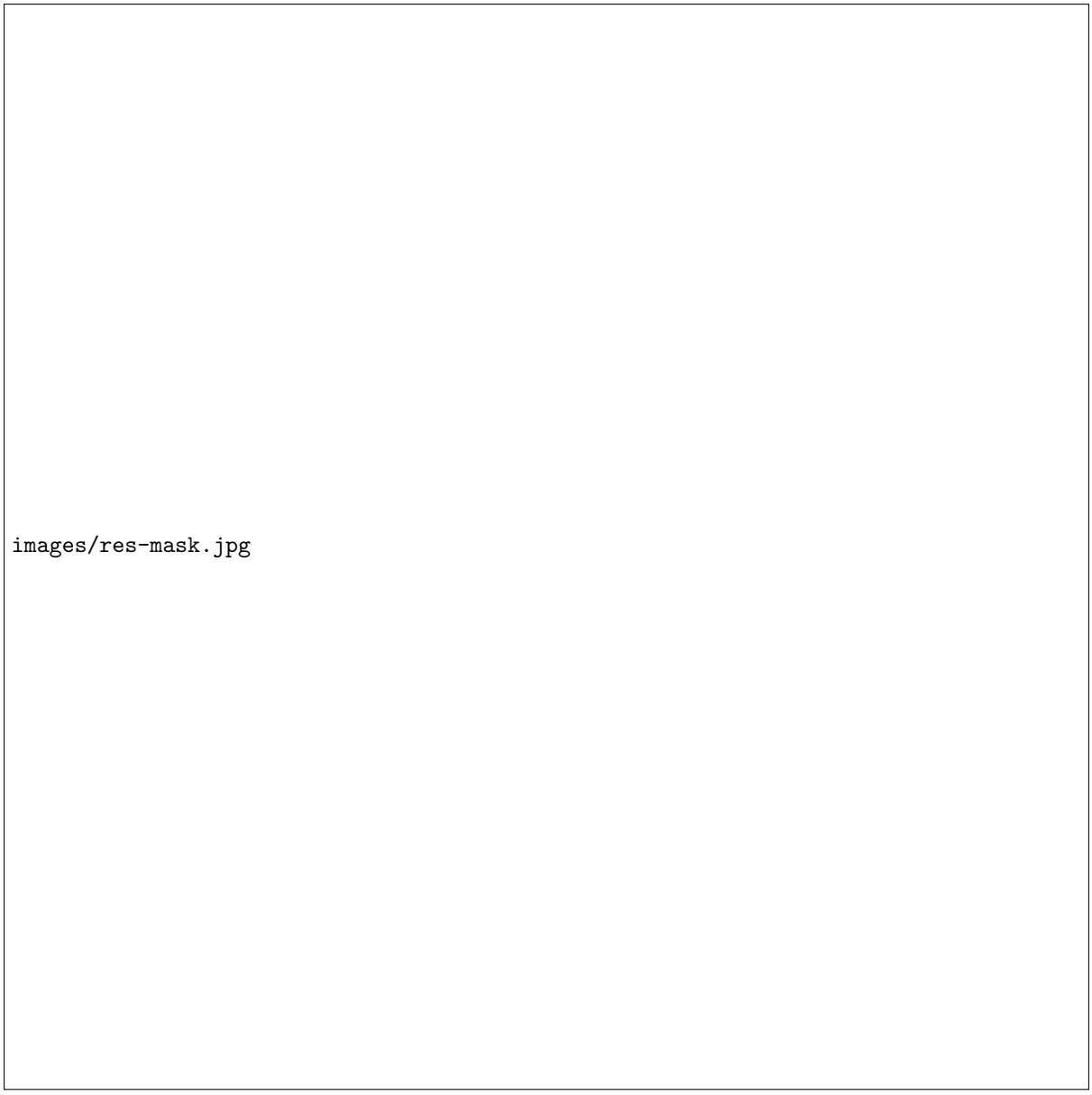


Figure 12: Mask detection.

bottom frames as 14.7, 16.9 and 20.0. From the first frame to the second frame (Fig. 13 to Fig. 14), we can see the distance activity matrix brightening in the right top and left bottom edges. This is due to close proximity of persons ID 1 and 4. This leads to an increase in the threat level of the frame by 16.9 - 14.7 = 2.2. Similarly, when looking at the first and third frames (Fig. 13 to Fig. 15), this time the interaction activity matrix brightens up in the third frame due to the handshake interaction in this frame. This also leads to an increase in threat level, which is by 20.0 - 14.7 = 5.3. It is also clearly observed in the threat activity matrix for the third frame in Fig. 15, where the center squares brighten up to show a significant threat between person 2 and 3. This increment (of 5.3) in threat level is higher than the previous comparison (of 2.2) in Fig. 13 and Fig. 14 since the handshake interaction poses a higher threat than proximity alone. The same can be observed by comparing the second and third frames.

A closer look at the activity matrices for different parameters can act as an ablation study for the algorithm. The four smaller matrices are distance activity matrices (top left), interaction activity matrices (top right), mask activity matrices (bottom left) and group activity matrices (bottom right). [JH: @GIHAN I dont get the "ablation" part and i have mentioned this next part in the prev para before reading this. Now idk wether to delete this or what] The distance-based threat level can be observed to increase for a pair of people in Fig. 14 (two light green color entries in the matrix). Similarly, the interaction being detected in Fig. 15 can be observed (two yellow color entries in the matrix).

A simpler situation is analyzed in Fig. 16. Here, there are only two people belonging to the same group and they are present in the video throughout the time. However, there are no physical interactions like shaking hands. Therefore, the only parameter that dictates the threat level is the number of people and their inter-personal distances in each frame. When analyzing the Fig. 16 the people in the first frame are moving away from each other until the second frame. This is why the threat level goes down from 95.0 to 46.0 from the first frame to the second. In the third frame, new people come into the frame and they get closer to each other. Therefore, an increase in the threat level of 105.3 is observed. However, this dataset does not contain a rich set of scenes to evaluate all components of the proposed system.

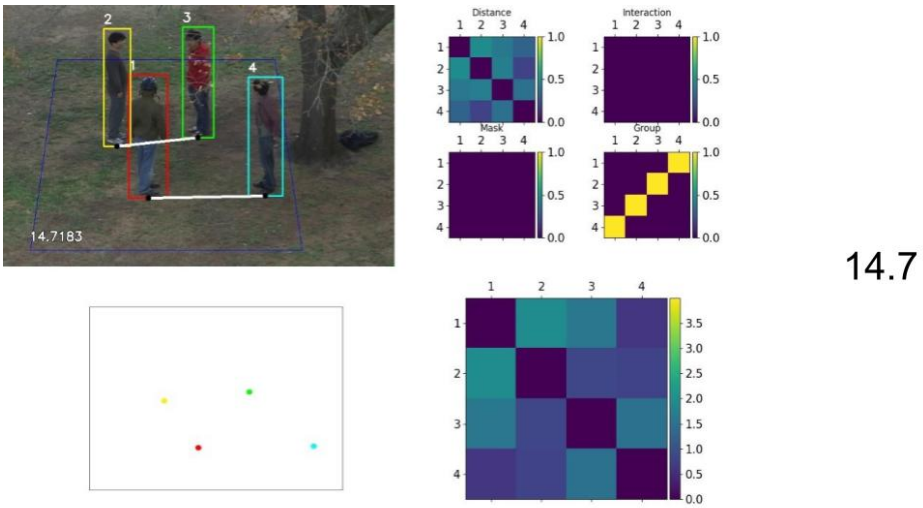


Figure 13: Full system result of UTI interaction dataset at t_1

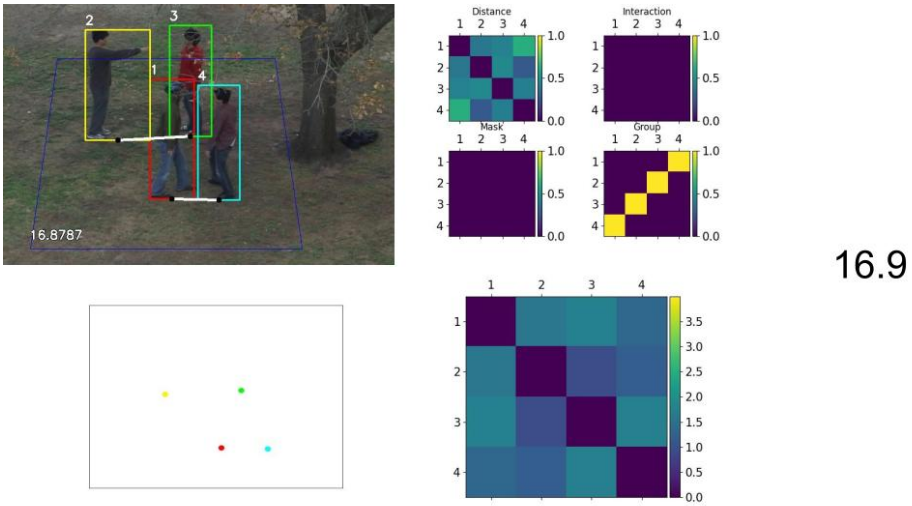


Figure 14: Full system result of UTI interaction dataset at t_2

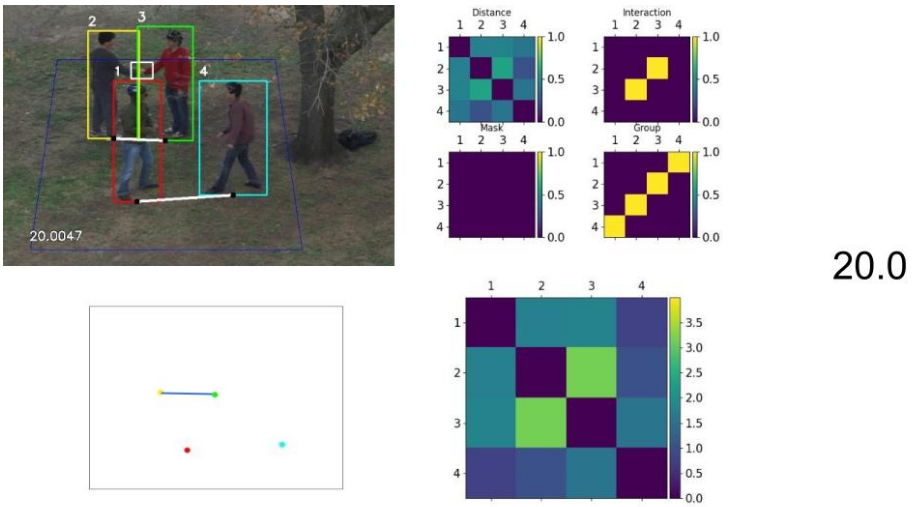


Figure 15: Full system result of UTI interaction dataset at t_3

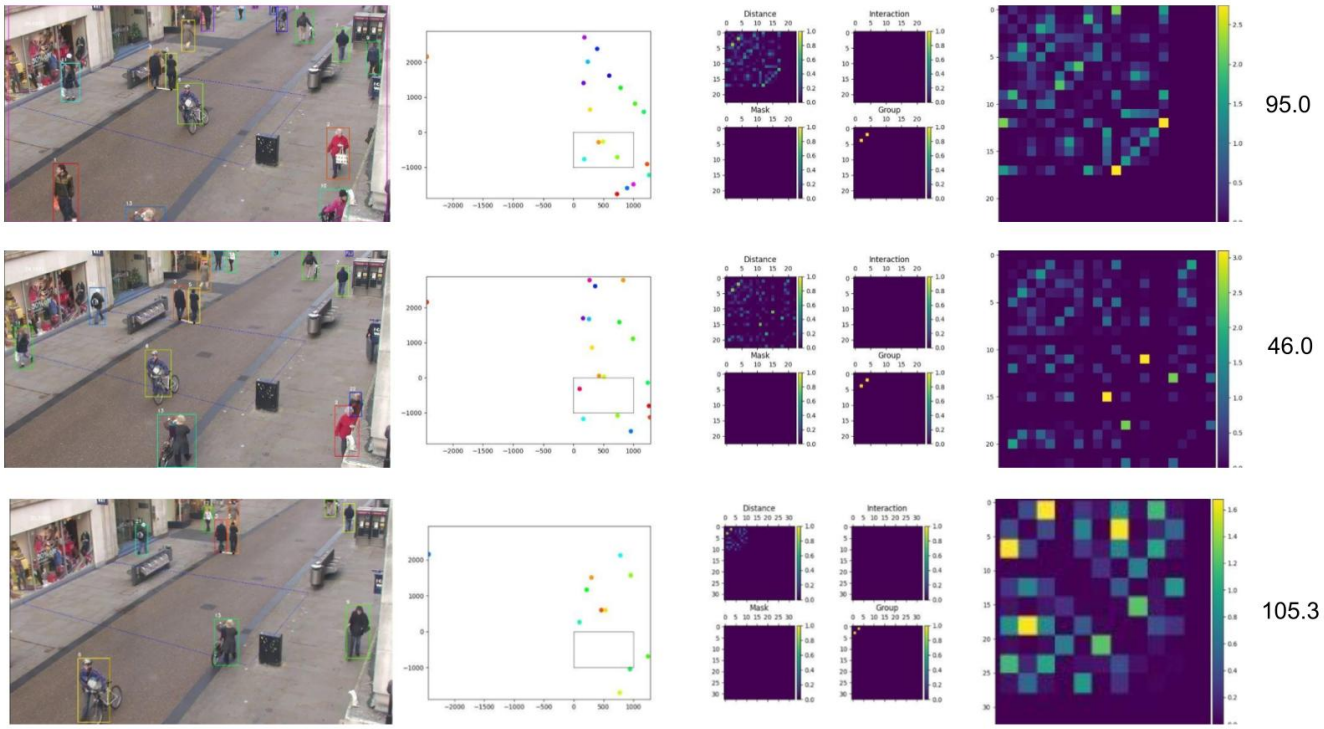


Figure 16: Full system result of oxford dataset

4.7 Full system evaluation

The performance of the full system in comparison to human expert responses is provided in Table 5 in terms of accuracy, precision and recall. It can be noted that the system performance is not biased towards either dataset and is able to generalize with considerable accuracy of nearly 76%.

Table 5: Full system performance

Test	**Accuracy**	**Precision**	**Recall**
UTI dataset	75%	75%	75%
UOP dataset	76%	85%	79%
Overall	76%	81%	77%

Few of the notable failure cases of the system are shown in Fig. 17 - 20, where the threat level predicted was on the contrary to human expert opinion. Out of the 4 cases shown here 3 of them failed to evaluate the proper threat value due to a failure in one of the components in the system pipeline. In Fig. 17, the person indicated by the purple arrow was not detected by the person detection model due to occlusion. Similarly, in Fig. 18 the two individuals hugging are detected as a single person. Since it is the proximity of the three individuals in Fig. 17 and the hugging individuals in Fig. 18 that pose a high threat to COVID-19 spread, the system fails to reflect this, deviating from the expert opinion. In Fig. 19 the high proximity of the individuals in the first frame give a high threat value for the first

frame. However, the handshake interaction model fails to detect the interaction in the second frame hence leading to a lower threat level output by the system and hence failing to identify the increase in threat for COVID-19 spread. In the case of Fig. 20, since the system design was not accounted for incidents such as a pushing action as in the second frame, the system provides a higher threat value for the first frame on the contrary to human expert opinion.

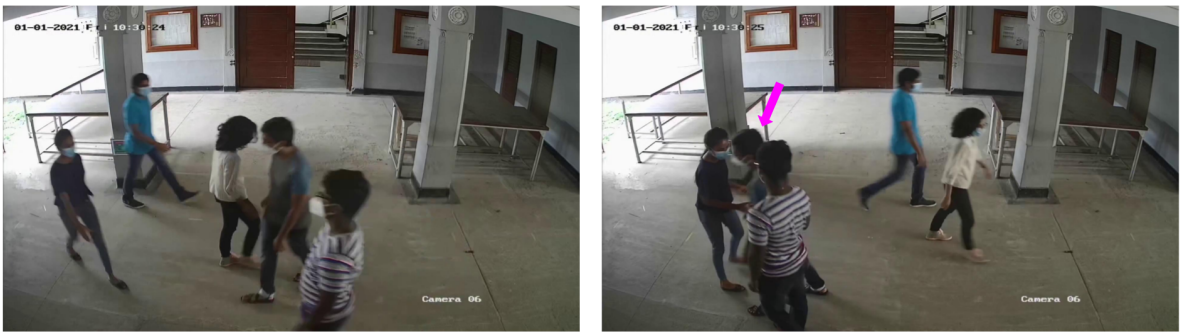


Figure 17: Failure case 1 threat level interpretations
System output for threat - Decreases, Human expert opinion on threat - Increases

Figure 18: Failure case 2 threat level interpretations
System output for threat - Increases, Human expert opinion on threat - Decreases

Figure 19: Failure case 3 threat level interpretations
System threat evaluation output - Decreases, Human expert opinion output - Increases

However, there were few rare cases where in retrospect, the system output was more plausible or instances where the failure by the system was unexplained. Considering Fig. 21, the ground truth from

Figure 20: Failure case 4 threat level interpretations
System threat evaluation output - Decreases, Human expert opinion output - Increases

human expert opinion was that the threat level decreases, which is explained by the handshake interaction in the first frame which is a serious violation of social distancing protocols. However, the system output for threat value increases significantly in the second frame as a new person is identified in the far left. Since an increase in the number of people and closer proximity of this new person in a given space should also be accounted to, this leads to the increased threat value predicted by the system. In the meantime, Fig. 22 is an instance where the system output states the threat for COVID-19 spread has increased, whereas human expert opinion is on the contrary. This deviation by the system is an edge case where the deviation is unexplained.

Figure 21: Edge case 1 threat level interpretations
System threat evaluation output - Increases, Human expert opinion output - Decreases

5 Conclusion

An end-to-end solution for monitoring social distancing protocols, interpersonal interactions such as handshakes, as well as mask-wearing; utilizing CCTV footage based on computer vision and graph theory is presented here. The proposed system provides a practical and versatile mechanism for monitoring crowds to identify possible instances for spreading COVID-19. The model was evaluated using pre-existing datasets as well as the proposed UOP dataset. Consistent accuracies over 75% across all datasets could

Figure 22: Edge case 2 threat level interpretations
System threat evaluation output - Increases, Human expert opinion output - Decreases

be considered as a strong indication of the performance of the proposed system.

The proposed architecture and system enables it to identify violations of COVID-19 safety protocols, including social distancing and dyadic human interactions, consistent with the ground reality. This feature is enabled due to the fact that the proposed system takes a holistic approach towards identifying crowd behavior characteristics as well as the behaviors of individuals within a crowd in a single frame and also across multiple frames. In other words, it utilizes a spatial and temporal analysis to identify the key properties including the distance between persons, mask-wearings, handshakes, and other interactions. This feature makes the proposed methodology more robust as typical human interactions (that may lead to the spread of COVID-19 or any other contagious disease) require close encounters of humans spanning some time duration. The proposed system realizes this characteristic.

Furthermore, this unified framework allows for future incorporation of possible other future measures for curtailing the spread of COVID-19 or any other epidemics impacting the health and safety of the society. Therefore, this proposed framework may be strengthened by incorporating more and more COVID-19 specific features as well as it could be adapted and adopted for similar other scenarios which may benefit from video or CCTV based non intrusive observations.

The proposed system is suitable for many real world scenarios due to the pervasiveness of CCTV cameras in smart cities. The reduction of performance to cost ratio of computing hardware (capable of running ANN based algorithms efficiently) has made it possible for even smaller scale organizations to acquire the system. Free and open source (FOSS) release of the codebase can accelerate both third party deployment and improvement of the solution. However, privacy, bias, and fairness concerns in conducting analytics on CCTV footage of people should be handled in a case by case basis based on the rules and regulations of individual organizations and countries.

6 Acknowledgements

This work is funded by Lewis Power, Singapore and International Development Research Centre (IDRC), Canada.

References

[1] Dahai Zhao, Feifei Yao, Lijie Wang, Ling Zheng, Yongjun Gao, Jun Ye, Feng Guo, Hui Zhao, and Rongbao Gao. A comparative study on the clinical features of coronavirus 2019 (covid-19) pneumonia with other pneumonias. *Clinical Infectious Diseases*, 71(15):756--761, 2020.

[2] Brit Long, William J Brady, Alex Koyfman, and Michael Gottlieb. Cardiovascular complications in covid-19. *The American journal of emergency medicine*, 38(7):1504--1507, 2020.

[3] Mark A Ellul, Laura Benjamin, Bhagteshwar Singh, Suzannah Lant, Benedict Daniel Michael, Ava Easton, Rachel Kneen, Sylviane Defres, Jim Sejvar, and Tom Solomon. Neurological associations of covid-19. *The Lancet Neurology*, 2020.

[4] Jamie Lopez Bernal, Nick Andrews, Charlotte Gower, Eileen Gallagher, Ruth Simmons, Simon Thelwall, Julia Stowe, Elise Tessier, Natalie Groves, Gavin Dabrera, Richard Myers, Colin N.J. Campbell, Gayatri Amirthalingam, Matt Edmunds, Maria Zambon, Kevin E. Brown, Susan Hopkins, Meera Chand, and Mary Ramsay. Effectiveness of Covid-19 Vaccines against the B.1.617.2 (Delta) Variant. *New England Journal of Medicine*, 385(7):585--594, 2021.

[5] Matthew McCallum, Jessica Bassi, Anna De Marco, Alex Chen, Alexandra C Walls, Julia Di Iulio, M Alejandra Tortorici, Mary-Jane Navarro, Chiara Silacci-Fregni, Christian Saliba, et al. SARS-CoV-2 immune evasion by variant B. 1.427/B. 1.429. *Science*, 373(6555):648--654, 2021.

[6] Piero Olliaro, Els Torreele, and Michel Vaillant. Covid-19 vaccine efficacy and effectiveness—the elephant (not) in the room. *The Lancet Microbe*, 2(7):279--2809, 4 2021.

[7] Ali Pormohammad, Mohammad Zarei, Saied Ghorbani, Mehdi Mohammadi, Mohammad Hossein Razizadeh, Diana L Turner, and Raymond J Turner. Efficacy and safety of covid-19 vaccines: A systematic review and meta-analysis of randomized clinical trials. *Vaccines*, 9(5):467, 2021.

[8] Leila Abdullahi, John Joseph Onyango, Carol Mukiira, Joyce Wamicwe, Rachel Githiomi, David Kariuki, Cosmas Mugambi, Peter Wanjohi, George Githuka, Charles Nzioka, Jennifer Orwa, Rose Oronje, James Kariuki, and Lilian Mayieka. Community interventions in low—and middle-income countries to inform covid-19 control implementation decisions in kenya: A rapid systematic review. *PLOS ONE*, 15(12):1--29, 12 2020.

[9] Swati Mukerjee, Clifton M. Chow, and Mingfei Li. Mitigation strategies and compliance in the covid-19 fight; how much compliance is enough? *PLOS ONE*, 16(8):1--19, 08 2021.

[10] Swetaprovo Chaudhuri, Saptarshi Basu, Prasenjit Kabi, Vishnu R Unni, and Abhishek Saha. Modeling the role of respiratory droplets in covid-19 type pandemics. *Physics of Fluids*, 32(6):063309, 2020.

[11] Thushara Galbadage, Brent M Peterson, and Richard S Gunasekera. Does covid-19 spread through droplets alone? *Frontiers in public health*, 8:163, 2020.

[12] Trisha Greenhalgh, Jose L Jimenez, Kimberly A Prather, Zeynep Tufekci, David Fisman, and Robert Schooley. Ten scientific reasons in support of airborne transmission of sars-cov-2. *The lancet*, 397(10285):1603--1605, 2021.

[13] Derek K Chu, Elie A Akl, Stephanie Duda, Karla Solo, Sally Yaacoub, Holger J Schünemann, Amena El-harakeh, Antonio Bognanni, Tamara Lotfi, Mark Loeb, et al. Physical distancing, face masks, and eye protection to prevent person-to-person transmission of sars-cov-2 and covid-19: a systematic review and meta-analysis. *The lancet*, 395(10242):1973--1987, 2020.

[14] Nina B Masters, Shu-Fang Shih, Allen Bukoff, Kaitlyn B Akel, Lindsay C Kobayashi, Alison L Miller, Harapan Harapan, Yihan Lu, and Abram L Wagner. Social distancing in response to the novel coronavirus (covid-19) in the united states. *PloS one*, 15(9):e0239025, 2020.

[15] Linda Thunström, Stephen C Newbold, David Finnoff, Madison Ashworth, and Jason F Shogren. The benefits and costs of using social distancing to flatten the curve for covid-19. *Journal of Benefit-Cost Analysis*, 11(2):179--195, 2020.

[16] Günter Kampf. Potential role of inanimate surfaces for the spread of coronaviruses and their inactivation with disinfectant agents. *Infection Prevention in Practice*, 2(2):100044, 2020.

[17] Sarah L Warnes, Zoë R Little, and C William Keevil. Human coronavirus 229e remains infectious on common touch surface materials. *MBio*, 6(6), 2015.

[18] Steffen E Eikenberry, Marina Mancuso, Enahoro Iboi, Tin Phan, Keenan Eikenberry, Yang Kuang, Eric Kostelich, and Abba B Gumel. To mask or not to mask: Modeling the potential for face mask use by the general public to curtail the covid-19 pandemic. *Infectious Disease Modelling*, 5:293--308, 2020.

[19] Shakil Bin Kashem, Dwayne M Baker, Silvia R González, and C Aujean Lee. Exploring the nexus between social vulnerability, built environment, and the prevalence of covid-19: A case study of chicago. *Sustainable cities and society*, 75:103261, 2021.

[20] Hassan Ugail, Riya Aggarwal, Andrés Iglesias, Newton Howard, Almudena Campuzano, Patricia Suárez, Muazzam Maqsood, Farhan Aadil, Irfan Mehmood, Sarah Gleghorn, et al. Social distancing enhanced automated optimal design of physical spaces in the wake of the covid-19 pandemic. *Sustainable Cities and Society*, 68:102791, 2021.

[21] Chen Ren, Chang Xi, Junqi Wang, Zhuangbo Feng, Fuzhan Nasiri, Shi-Jie Cao, and Fariborz Haghighat. Mitigating covid-19 infection disease transmission in indoor environment using physical barriers. *Sustainable cities and society*, 74:103175, 2021.

[22] Shubham Srivastava, Xingwang Zhao, Ati Manay, and Qingyan Chen. Effective ventilation and air disinfection system for reducing coronavirus disease 2019 (covid-19) infection risk in office buildings. *Sustainable Cities and Society*, page 103408, 2021.

[23] George Grekousis and Ye Liu. Digital contact tracing, community uptake, and proximity awareness technology to fight covid-19: a systematic review. *Sustainable cities and society*, 71:102995, 2021.

[24] Sizhen Bian, Bo Zhou, Hymalai Bello, and Paul Lukowicz. A wearable magnetic field based proximity sensing system for monitoring covid-19 social distancing. In *Proceedings of the 2020 International Symposium on Wearable Computers*, pages 22--26, 2020.

[25] Maria Fazio, Alina Buzachis, Antonino Galletta, Antonio Celesti, and Massimo Villari. A proximity-based indoor navigation system tackling the COVID-19 social distancing measures. *Proceedings - IEEE Symposium on Computers and Communications*, 2020-July, 2020.

[26] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440--1448, 2015.

[27] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21--37. Springer, 2016.

[28] Mohammad Javad Shafiee, Brendan Chywl, Francis Li, and Alexander Wong. Fast yolo: A fast you only look once system for real-time embedded object detection in video. *arXiv preprint arXiv:1709.05943*, 2017.

[29] Rafael Munoz-Salinas, Eugenio Aguirre, and Miguel García-Silvente. People detection and tracking using stereo vision and color. *Image and Vision Computing*, 25(6):995--1007, 2007.

[30] Jacek Czyz, Branko Ristic, and Benoit Macq. A particle filter for joint detection and tracking of color objects. *Image and Vision Computing*, 25(8):1271--1281, 2007.

[31] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. *Proceedings - International Conference on Image Processing, ICIP*, 2017-Septe:3645--3649, 2018.

[32] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645--3649, 2017.

[33] Mohd Aquib Ansari and Dushyant Kumar Singh. Monitoring social distancing through human detection for preventing/reducing covid spread. *International Journal of Information Technology*, pages 1--10, 2021.

[34] Imran Ahmed, Misbah Ahmad, and Gwanggil Jeon. Social distance monitoring framework using deep learning architecture to control infection transmission of covid-19 pandemic. *Sustainable Cities and Society*, 69:102777, 2021.

[35] Imran Ahmed, Misbah Ahmad, Joel JPC Rodrigues, Gwanggil Jeon, and Sadia Din. A deep learning-based social distance monitoring framework for covid-19. *Sustainable Cities and Society*, 65:102571, 2021.

[36] Jingchen Qin and Ning Xu. Reaserch and implementation of social distancing monitoring technology based on ssd. *Procedia Computer Science*, 183:768--775, 2021.

[37] Sergio Saponara, Abdussalam Elhanashi, and Alessio Gagliardi. Implementing a real-time, ai-based, people detection and social distancing measuring system for covid-19. *Journal of Real-Time Image Processing*, pages 1--11, 2021.

[38] Mohammad Shorfuzzaman, M Shamim Hossain, and Mohammed F Alhamid. Towards the sustainable development of smart cities through mass video surveillance: A response to the covid-19 pandemic. *Sustainable cities and society*, 64:102582, 2021.

[39] Adina Rahim, Ayesha Maqbool, and Tauseef Rana. Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera. *Plos one*, 16(2):e0247440, 2021.

[40] Jie Su, Xiaohai He, Linbo Qing, Tong Niu, Yongqiang Cheng, and Yonghong Peng. A novel social distancing analysis in urban public space: A new online spatio-temporal trajectory approach. *Sustainable Cities and Society*, 68:102765, 2021.

[41] Mahdi Rezaei and Mohsen Azarmi. Deepsocial: Social distancing monitoring and infection risk assessment in covid-19 pandemic. *Applied Sciences (Switzerland)*, 10(21):1--29, 2020.

[42] Dongfang Yang, Ekim Yurtsever, Vishnu Renganathan, Keith A Redmill, and Ümit Özgüner. A vision-based social distancing and critical density detection system for covid-19. *Sensors*, 21(13):4608, 2021.

[43] Narinder Singh Punn, Sanjay Kumar Sonbhadra, and Sonali Agarwal. Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques, 2020.

[44] Biparnak Roy, Subhadip Nandy, Debojit Ghosh, Debarghya Dutta, Pritam Biswas, and Tamodip Das. Moxa: a deep learning based unmanned approach for real-time monitoring of people wearing medical masks. *Transactions of the Indian National Academy of Engineering*, 5(3):509--518, 2020.

[45] Puranjay Mohan, Aditya Jyoti Paul, and Abhay Chirania. A tiny cnn architecture for medical face mask detection for resource-constrained endpoints. In *Innovations in Electrical and Electronic Engineering*, pages 657--670. Springer, 2021.

[46] Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N Taha, and Nour Eldeen M Khalifa. Fighting against covid-19: A novel deep learning model based on yolo-v2 with resnet-50 for medical face mask detection. *Sustainable cities and society*, 65:102600, 2021.

[47] Shubham Shinde, Ashwin Kothari, and Vikram Gupta. Yolo based human action recognition and localization. *Procedia computer science*, 133:831--838, 2018.

[48] Yasaman S Sefidgar, Arash Vahdat, Stephen Se, and Greg Mori. Discriminative key-component models for interaction detection and recognition. *Computer Vision and Image Understanding*, 135:16--30, 2015.

[49] Coert Van Gemeren, Ronald Poppe, and Remco C Veltkamp. Hands-on: deformable pose and motion models for spatiotemporal localization of fine-grained dyadic interactions. *EURASIP Journal on Image and Video Processing*, 2018(1):1--16, 2018.

[50] AS Hassan, Suren Sritharan, Gihan Jayatilaka, Roshan I Godaliyadda, Parakrama B Ekanayake, Vijitha Herath, and Janaka B Ekanayake. Hands off: A handshake interaction detection and localization model for covid-19 threat control. *arXiv preprint arXiv:2110.09571*, 2021.

[51] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection, 2020.

[52] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679--698, 1986.

[53] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740--755. Springer, 2014.

[54] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020.

[55] Yan Jiang, Feng Gao, and Guoyan Xu. Computer vision-based multiple-lane detection on straight road and in a curve. In *2010 International Conference on Image Analysis and Signal Processing*, pages 114--117. IEEE, 2010.

[56] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395--416, 2007.

[57] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[58] Laura Leal-Taixé, Anton Milan, Ian Reid, Stefan Roth, and Konrad Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv preprint arXiv:1504.01942*, 2015.

[59] M. S. Ryoo and J. K. Aggarwal. UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA). http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html, 2010.

[60] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[61] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303--338, 2010.

[62] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018.

[63] Stephen Robertson. A new interpretation of average precision. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 689--690, 2008.

[64] R. Padilla, S. L. Netto, and E. A. B. da Silva. A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 237--242, 2020.

[65] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248--255. Ieee, 2009.