Beyond Social Distancing: Application of real-world coordinates in a multi-camera system with privacy protection

Frances Ryan, Feiyan Hu, Julia Dietlmeier, Noel E. O'Connor, Kevin McGuinness

Insight SFI Research Centre for Data Analytics, Dublin City University, Dublin, Ireland

Abstract

In this paper, we develop a privacy-preserving framework to detect and track pedestrians and project to their real-world coordinates facilitating social distancing detection. The transform is calculated using social distancing markers or floor tiles visible in the camera view, without an extensive calibration process. We select a lightweight detection model to process CCTV videos and perform tracking within-camera. The features collected during within-camera tracking are then used to associate passenger trajectories across multiple cameras. We demonstrate and analyze results qualitatively for both social distancing detection and multi-camera tracking on real-world data captured in a busy airport in Dublin, Ireland.

Keywords: Social distancing, people detection, cross-camera tracking, coordinate calibration

1 Introduction

During pandemics, social distancing helps slow the spread of the virus and as Covid-19 has proven will be an important tool in the fight against future pandemics. Additionally, identifying and managing areas with regular overcrowding is vital in crowded public places to ensure pedestrian safety and monitor wait or queue times.

A key element of social distancing detection is transforming pedestrian bounding boxes to real-world coordinates. This usually requires a precomputed/learnt transform matrix. Researchers have proposed different ways to infer this transform in the past year. The most popular way is to find three or more pairs of points in both image and real-world coordinates; a perspective matrix is computed using these pairs of points. However, most of the time it is challenging to specify real-world 2D coordinates. [Yang et al., 2021] tried to find real-world reference points using the floor plan of public buildings such as a train station, while others have tried using information about detected pedestrians height to estimate size of reference objects and real-world coordinates [Cong et al., 2020]. There are also some researchers that do not compute the distance between every pedestrian pair. Instead, for each pedestrian, a circular violation zone is established, and pedestrians appearing within the zone are marked as a violating group. [Punn et al., 2020] used the estimated depth from the camera to estimate the violation area for each pedestrian. [Aghaei et al., 2021] assume camera roll and pan angle to be 0, and project the image to the real-world and then use torso size instead of whole body height to compute the area of the violation zone.

In our work, we compute the perspective transform matrix by taking advantage of the commonly used social distancing markers on the floor. There are some benefits to this choice of reference points: 1) all stickers are on the same real-world plane; 2) the relative distance between stickers is largely consistent; 3) markers such as these have become commonplace during the pandemic.

The main contributions of this paper are as follows:

- A proof of concept using social distancing markers/floor tiles to compute the perspective matrix.
- Extension of a single camera to a multi-camera system by trajectory matching for cross-camera tracking.
- Demonstration and evaluation of both functionalities on challenging data from a busy airport.

Figure 1 gives an overview of the proposed system. Frames from multiple views are passed through the face and pedestrian detection module. The face boxes can be blurred, hence anonymizing the image data. Pedestrian boxes are input into the tracking module. The bottom-centre points(approximate foot location) are transformed to a top-down view, where person distances can be calculated. Points can further be transformed to world coordinates to plot in the existing GIS system. Within-camera tracks and appearance features can be passed to the cross-camera trajectory association module to assign final identities. To achieve a balance between speed and accuracy, we used the YOLOv5x[1] detector, trained on the CrowdHuman dataset [Shao et al., 2018]). DeepSort [Wojke et al., 2017] is used for within-camera tracking. To extract good ReID features, we use the method proposed in [Jia et al., 2019] to tackle domain shift using instance normalization in early layers and feature normalization in deep layers.

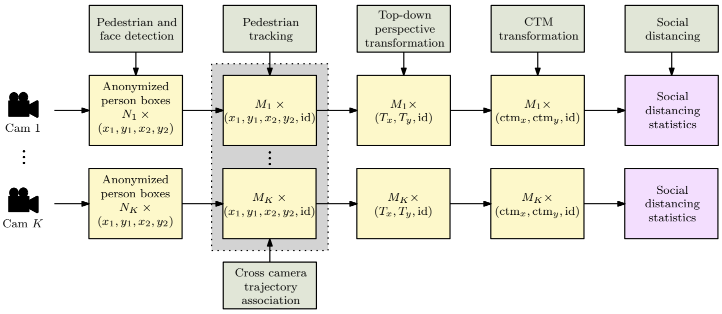

Pedestrian and face detection
Pedestrian tracking
Top-down perspective transformation
CTM transformation
Social distancing
Cam 1
Anonymized person boxes $N_1 \times$ (x_1, y_1, x_2, y_2)
$M_1 \times$ $(x_1, y_1, x_2, y_2, \mathrm{id})$
$M_1 \times$ (T_x, T_y, id)
$M_1 \times$ $(\mathrm{ctm}_x, \mathrm{ctm}_y, \mathrm{id})$
Social distancing statistics
Cam K
Anonymized person boxes $N_K \times$ (x_1, y_1, x_2, y_2)
$M_K \times$ $(x_1, y_1, x_2, y_2, \mathrm{id})$
$M_K \times$ (T_x, T_y, id)
$M_K \times$ $(\mathrm{ctm}_x, \mathrm{ctm}_y, \mathrm{id})$
Social distancing statistics
Cross camera trajectory association

Figure 1: System overview.

2 System architecture

2.1 Transformation to Real-World Coordinates

The perspective transform matrix to top-down view is computed from 4 pairs of points. Source points – image coordinates describing the location of 4 distancing markers (or other fixed points, e.g. the corners of floor tiles) that form a square or rectangle – and target points. Transforming between image coordinates and top-down view with multiple reference points forms a system of linear equations that is solved by Gaussian elimination with the optimal pivot element to estimate the 3×3 perspective transform matrix. The transformation from top-down to world coordinates is a problem similar to rigid body movement, thus the Customised Transverse Mercator (CTM) transformation matrix can be computed by solving an Orthogonal Procrustes problem [Schönemann, 1966]. The problem can be formulated as the following optimization with the constraint that the rotation matrix R is orthonormal:

$$\begin{aligned} \min_{\sigma, R, \vec{t}} \|RA - \hat{B}\|^2, &\quad \sigma\hat{B} = B - \vec{t} \cdot \mathbb{1} \\ \text{s.t.} \quad R^\top R = RR^\top &= \mathbb{I} \in \mathbb{R}^{3\times3}, \end{aligned} \tag{1}$$

where σ is a scaling factor, R is a rotation matrix, and $\vec{t}$ is a translation vector. A and B are matrices whose columns contain the corresponding points in top-down and CTM coordinates. $\hat{B}$ is shifted and scaled back from B such that the scale of $\hat{B}$ is the same as A and a pair of corresponding points in A and $\hat{B}$ are aligned as origin points. To solve the optimization, σ is estimated first as average scale factors between corresponding edges. The edge between two nodes is calculated using the L^2 norm. $\vec{t} = B[:,1] - \sigma A[:,1]$ is estimated using σ as a vector formed from a pair of corresponding points in B to A. Finally, we solve the Orthogonal Procrustes problem by singular vector decomposition as $UDV^\top = \mathrm{svd}(\hat{B}A^\top)$. The rotation matrix is calculated as $R = UJV^\top$, where J is an identity matrix. In the case of $\det(UV^\top) = -1$, the value on diagonal of J that corresponds to the smallest value of D is set to -1.

2.2 Multi-Target Cross-Camera Re-Identification

In the airport, cameras can be selected such that the following conditions are satisfied: pedestrians 1) first appear in a set of non-overlapping ‘query’ cameras at various times; 2) travel through a known topology of subsequent

[1] https://github.com/ultralytics/yolov5

'gallery' cameras; 3) can only reappear in the next camera after having left the previous camera. Within-camera trajectories are post-processed to exclude those that are very short, that matched with low-confidence or very small bounding boxes. The trajectories from both query and gallery cameras are then input into the cross-camera trajectory association module. The within-camera appearance features collected for each trajectory are averaged such that there is one feature vector per identity from all cameras. Initially, all trajectories from the query cameras are associated with the next camera according to the network topology. This is done by calculating the cosine distance between all averaged feature vectors and associated using the Hungarian algorithm [Kuhn, 1955]. Time constraints [Dietlmeier et al., 2022] are applied to remove the possibility of matching with target identities appearing outside of pre-configured time constraints, e.g. before a query identity left the query camera. A threshold is set such that if the distance exceeds this, identities should not be associated. Unmatched query trajectories are still carried forward for potential association with a trajectory in the next camera. If, on the other hand, the match distance is within the threshold, the trajectories are appended and carried forward for association in further cameras.

3 Experiments and Results

Social distancing. We demonstrate the system performing the social-distance monitoring function across various areas within the airport. Detected people far from the camera are removed by post-processing based on bounding-box location, due to inaccuracy caused by the perspective transform in those locations. The Euclidean distances between pairs of people are calculated and assessed against the configurable 'social distance.'. Figure 3a shows social-distance monitoring can be effective for sparse and medium crowd density.

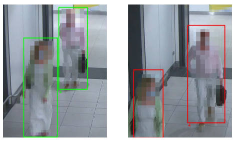
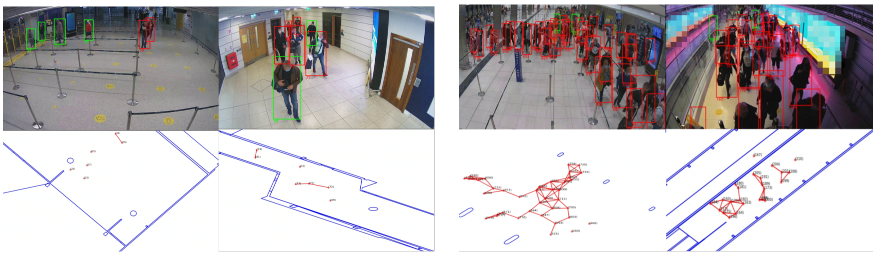


Figure 2: (left) distances are correctly detected (right) the pair are falsely found to be within $2m$.

Overestimation of social distance. Figure 2 demonstrates failures when the bounding box is cut-off due to a person exiting a scene, since this results in inaccurate estimations of foot location for a person and, hence, the transformed position is incorrect. This could result in overestimation of social distancing violations in this area. However, this can be prevented by assigning a specific region of interest within certain camera views to carry out the analysis, thereby excluding areas where the bounding box may be unreliable.

Underestimation of social distance. Figure 3b shows failures may occur in areas with high crowd density in the far background. Single person bounding boxes may be detected among a crowd and the person is mistakenly marked as exceeding a $2m$ distance, resulting in underestimation of violations in these areas. In some areas false positive detection is caused by reflections or people moving on the other side of glass panels, these can easily be filtered out based on location since the cameras are fixed. Nonetheless, we have found it is feasible to use floor markers to estimate the distance between pedestrians.

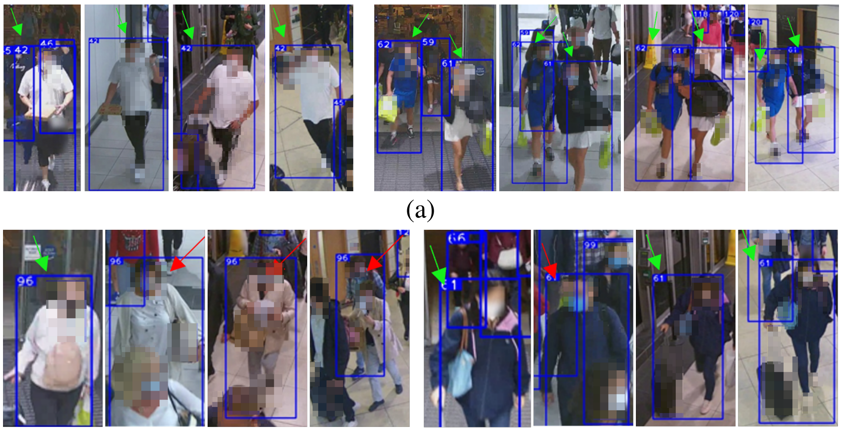


(a)

(b)

Figure 3: a) Successful examples; b) Failure cases of social distancing detection. For each, the top image shows the camera view with bounding boxes in red indicating people within $2m$ of another while the bottom image plots people in the CTM coordinate space – blue lines outline airport walls.

Multi-camera tracking. Figure 4 shows examples of tracking passengers cross-camera – disembarking the aircraft, entering and later exiting the immigration hall. The green arrow in the leftmost image in each group indicates the initial query and successful tracking through the remaining cameras. Incorrect matches are indicated with a red arrow. In general and as expected, people wearing distinctive clothing or bags are easily tracked across cameras even in the presence of reasonably high crowd density, as demonstrated by examples in Figure 4a.



(a)

(b)

Figure 4: a) Successful examples b) failure cases of tracking passengers.

Identity switches may occur between people wearing similar clothing as shown in the examples in Figure 4b. These mismatches can be due to people changing items of clothing, e.g. putting on a jacket, between query camera and subsequent cameras, or in cases where dense crowds cause occlusions entering an area. In certain cases, the match might be incorrect in a camera where there are dense crowds, but the correct match is found in other cameras where the crowd is more dispersed. Such failures could be minimized by selecting cameras where crowds are more likely to be dispersed or using overlapping cameras, where available, to tackle severe occlusion. The convolutional neural network features generalize well to the real-world data and our colleagues in the airport confirm that the tracking in world coordinates is helpful for operations. For social distancing detection, quantitative evaluation is challenging because it is time consuming and error-prone to obtain accurate ground-truth and annotation for multi-camera tracking data.

4 Conclusion

We have proposed an approach to computer vision based social distancing detection for multi-camera system in a challenging real-world environment. We showed an effective way to calculate perspective transform using distancing markers present in the image as reference points. Furthermore, we demonstrated the transformation from top-down coordinates to CTM coordinates by using a matrix that is computed by solving an Orthogonal Procrustes problem, extending the system to use trajectory-based association for cross-camera tracking. In future work, we hope to use the developed framework to assist in annotating airport data to further evaluate the system and investigate the scaling of a cross-camera tracking system across a larger area.

References

[Aghaei et al., 2021] Aghaei, M., Bustreo, M., Wang, Y., Bailo, G., Morerio, P., and Del Bue, A. (2021). Single image human proxemics estimation for visual social distancing. In *WACV*, pages 2785–2795.

[Cong et al., 2020] Cong, C., Yang, Z., Song, Y., and Pagnucco, M. (2020). Towards enforcing social distancing regulations with occlusion-aware crowd detection. In *ICARCV*, pages 297–302. IEEE.

[Dietlmeier et al., 2022] Dietlmeier, J., Hu, F., Ryan, F., O'Connor, N. E., and McGuinness, K. (2022). Improving person re-identification with temporal constraints. In *WACV Workshop*, pages 540–549.

[Jia et al., 2019] Jia, J., Ruan, Q., and Hospedales, T. (2019). Frustratingly easy person re-identification: Generalizing person Re-ID in practice. In *BMVC*, pages 141.1–141.14.

[Kuhn, 1955] Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

[Punn et al., 2020] Punn, N. S., Sonbhadra, S. K., Agarwal, S., and Rai, G. (2020). Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and deepsort techniques. *arXiv preprint arXiv:2005.01385*.

[Schönemann, 1966] Schönemann, P. H. (1966). A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31:1–10.

[Shao et al., 2018] Shao, S., Zhao, Z., Li, B., Xiao, T., Yu, G., Zhang, X., and Sun, J. (2018). CrowdHuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*.

[Wojke et al., 2017] Wojke, N., Bewley, A., and Paulus, D. (2017). Simple online and realtime tracking with a deep association metric. In *ICIP*.

[Yang et al., 2021] Yang, D., Yurtsever, E., Renganathan, V., Redmill, K. A., and Özgüner, Ü. (2021). A vision-based social distancing and critical density detection system for COVID-19. *Sensors*, 21:4608.