Poster: A Real-time Social Distance Measurement and Record System for COVID-19

Weijun Wang[1] Tingting Yuan[1] Minghao Han[2]
Meng Li[2] Han Zhang[2] Yu Ma[2] Sripriya Srikant Adhatarao[1] Xiaoming Fu[1]*

[1]University of Goettingen [2]Nanjing University

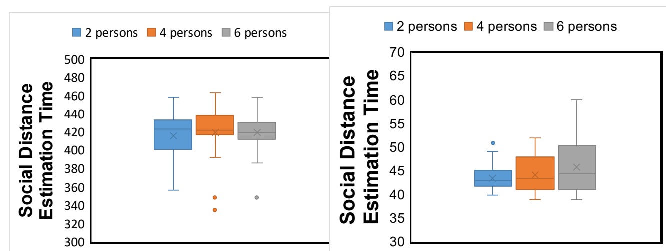
Abstract

COVID-19 coronavirus is spreading all over the world for almost one year. Keeping Social distance has been proven is an efficient policy against the spread of coronavirus. In this paper, we present a system for indoor monitoring. This system not only measures real-time social distance but also records the chain of potential inflection. In real-time social distance measurement, we use stereo cameras to guarantee high accurate measurement and utilize Edge node to ensure real-time requirements. In the chain of potential inflection record, we present an architecture including object tracking and Handshake Chain database, *i.e.*, Chain of potential inflection. To the best of our knowledge, we are the first presenting a system aim to accelerate query for COVID-19. Finally, we offer several on-going work to advance our system.

1 Introduction

COronaVIrus Disease 2019 (COVID-19) is highly pandemic around the world, facing severeness due to failure of maintaining social distancing and rapid detection of infectious contact chain. As they are time-consuming and high risk tasks for humans, staff-based warning and manual chain exploration have very limited efficacy. Recently, various mobile COVID-19 apps [1] have been developed, which utilize Bluetooth on smartphone to identify persons who may have been in contact, *e.g.*, break social distance for a duration, with an infected individual. But these apps need to collect users' information and this potential privacy issue leads the small user scale. Some papers try to utilize vision-based methods [2, 3] for social distance measurement. But all of them only provide low accurate results because they only use images from one viewing angle [2] and utilize the limited cameras' computational capability. [3].

In this poster, we present a system with stereo cameras and edge nodes to provide high accurate real-time social distance measurement. In addition, our system also includes a dataset to construct and record each occurred person's Handshake Chain (defined as the chain of his contact persons whose social distance is break for a specific duration, *i.e.*, a chain of potential inflection.) The reason to build this record system is easy for the epidemic prevention department to query the chain of specific person who test positive.[1]



(a) Social distancing on Camera (b) Social distancing on Edge

Figure 1. ***Comparison of social distance estimation time on Camera and Edge. Estimation cost* 450 *ms on cameras but only* 43 *ms on Edge node.***

There are two main challenges to build our system. The first challenge is that the measurement time of social distance is quite large which can not execute on cameras real-time. The second challenge is that how to accelerate the query of the Handshake Chain on the database.

To tackle the first challenge, we utilize edge nodes to help real-time measurement. Fig. 1 illustrates the experiment results of social distance measurement time on a camera and en edge, respectively. According to the experiment results, we observe that measurement time (including the data transmission time) on edge is 10× faster than on camera and the number of objects will not impact measurement time a lot.Thus, we address the first challenge. To tackle the second challenge, we are inspired by the idea of video query system [4]. Server first runs a deeper NN-based object detector to obtain higher accurate results. Then, calculate the distance among customers and execute cross-camera object tracking simultaneously. After that, it will construct the Handshake Chain database by the tacking route. Thus, we address the second challenge. In the following of this poster, we will discuss our architecture and on-going work.

[1]Note that how to identify people and notice them is out of this poster's scope.

International Conference on Embedded Wireless Systems and Networks (EWSN) 2021
17–19 February, Delft, The Netherlands
ISBN: 978-0-9949886-5-2

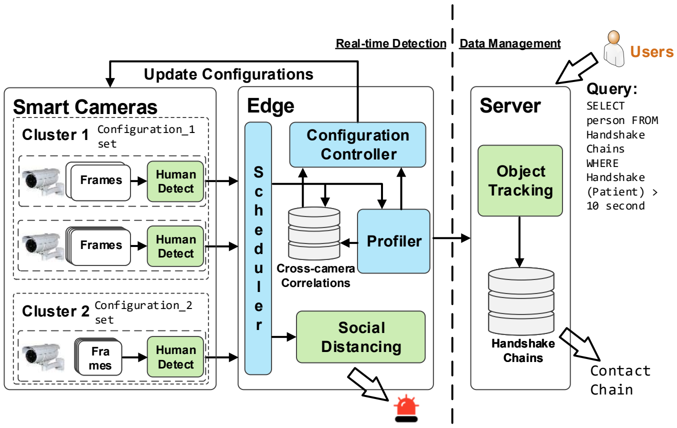


Figure 2. Architecture

2 Architecture

Our system contains two parts, *Real-time Detection* and *Data Management*, as shown in Fig. 2. Cameras and edge participate in the Real-time Detection part. In this part, cameras capture images (*Frames*) then execute *Human Detect* on each frame by CNN-based algorithm, *e.g.*, YOLO [5]. After that, cameras transmit the processed frames to edge for *Social Distance* measurement. If our system detects the social distance break event, it will deliver a message to related people, *e.g.*, let shop staff stop new customer entrance. One may think why not push the human detect model to more powerful edge. We will present the reason in the end of this section.

In Data Management part, edge and server track and organize the handshake chains for each detected people. Edge first transmits all frames to the server. Then, sever executes *Object Tracking* on all received frames to extract every people's route and delivers these routes to a handshake chain database. *Handshake Chain* database organizes and records all the information, *e.g.*, duration and location, of people who break social distance policy. Once some people are tested positive, epidemic prevention staff can query the database at first time for the contacted people.

After present main body of our system, let us have a look at the mechanism to further accelerate the social distance measurement time — *frame filtering*. In short, this mechanism accelerates the measurement time by avoid processing ineffective frames. The ineffective can be predefined as specific situations, *e.g.* all the detected people stay at their previous locations for a while, in which the whole information during this duration can be represented by one frame. To implement this mechanism, we put the human detect on the camera for frame filtering and periodically profile the accuracy-cost relation on edge as well as update configuration, *i.e.*, resolution and frame rate, for each camera. The reason to periodically profile and update cameras' configurations is that the content in cameras' images varies temporally and spatially and dynamic configurations (see the frame size and number in Fig. 2) can hold high accuracy as well as low cost. For example, there will not be many customers go shopping in early morning or late evening but many customers in the afternoon. Moreover, the distribution of people in one shop is not uniform, thus updating corresponding configurations for different camera clusters is necessary. On edge node, we add four modules to implement frame filtering mechanism together. In Fig. 2, *Scheduler* is in charge of sorting the order of frames according to each frame's priority (omit here for saving space) and deliver them to profiler and social distancing. *Profiler* profiles accuracy-cost relation periodically and delivers its profiling results to cross-camera correlations database and configuration controller. *Cross-camera Correlations* database comprehensively considers the profiling results and cameras' positions, then updates the cross-camera correlations and delivers them to the configuration controller. *Configuration Controller* fusions the current profiling results and historical cross-camera correlations information, then transmits updated configurations to cameras as their next perfiod's configurations.

3 On-going work

3.1 Privacy

Preserving privacy is important under COVID-19 because the positive test people are possibly leading to discrimination. To address this issue, we try to use depth camera instead of regular RGB camera. Yet, depth cameras bring three challenges we are working on them. First, Depth images only provide objects' rough sketch information leading significant challenge for cross-camera object tracking. Second, most public image sets are standard RGB images. Thus, how to construct depth image sets efficiently is our second challenge. Third, most object detection models are designed for RGB images. Thus, how to utilize pre-trained object detection under raw depth image is our third challenge.

3.2 Trade-off between accuracy and time cost

As we mentioned in Section II, updating cameras' configurations is a good way to accelerate the measurement speed. To further improve the efficiency of our system, we are working on the following scheme. Based on the learning knowledge from video content and stored in the cross-camera correlations database, cameras can be divided into clusters. In each cluster, cameras can vote and select the header camera. After election, header camera can trigger other in-cluster cameras when necessary.

3.3 Query-driven data management

Handshake chain database is the critical module in data management. Its data organization method dominates the query speed. Nowadays, some works on database use the NN-based method learning index to accelerate query speed [6]. Thus, we are trying to use this method to learn index from output routines of object tracking.

4 References

[1] "Covid-19 apps." [Online]. Available: https://en.wikipedia.org/wiki/COVID-19_apps

[2] D. Yang *et al.*, "A Vision-based Social Distancing and Critical Density Detection System for COVID-19," *arXiv:2007.03578*, 2020.

[3] N. Petrovic *et al.*, "IoT-based System for COVID-19 Indoor Safety Monitoring," p. 7.

[4] F. Bastani *et al.*, "MIRIS: Fast Object Track Queries in Video," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, Portland OR USA, 2020, pp. 1907–1921.

[5] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.

[6] J. Ding *et al.*, "ALEX: An Updatable Adaptive Learned Index," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 969–984.