Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
COVID-19 contact tracking by group activity trajectory recovery over camera networks

Chao Wang\textsuperscript{a,b}, XiaoChen Wang\textsuperscript{a,∗}, Zhongyuan Wang\textsuperscript{a,∗}, WenQian Zhu\textsuperscript{a,b}, Ruimin Hu\textsuperscript{a,b,c}

\textsuperscript{a} National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, Wuhan 430072, China
\textsuperscript{b} Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University, Wuhan 430072, China
\textsuperscript{c} Collaborative Innovation Center of Geospatial Technology, Wuhan 430079, China

A R T I C L E   I N F O

Article history:
Received 12 July 2021
Revised 14 July 2022
Accepted 16 July 2022
Available online 18 July 2022

Keywords:
Contact tracking
COVID-19
Group activity
Trajectory recovery

A B S T R A C T

Contact tracking plays an important role in the epidemiological investigation of COVID-19, which can effectively reduce the spread of the epidemic. As an excellent alternative method for contact tracking, mobile phone location-based methods are widely used for locating and tracking contacts. However, current inaccurate positioning algorithms that are widely used in contact tracking lead to the inaccurate follow-up of contacts. Aiming to achieve accurate contact tracking for the COVID-19 contact group, we extend the analysis of the GPS data to combine GPS data with video surveillance data and address a novel task named group activity trajectory recovery. Meanwhile, a new dataset called GATR-GPS is constructed to simulate a realistic scenario of COVID-19 contact tracking, and a coordinated optimization algorithm with a spatio-temporal constraint table is further proposed to realize efficient trajectory recovery of pedestrian trajectories. Extensive experiments on the novel collected dataset and commonly used two existing person re-identification datasets are performed, and the results evidently demonstrate that our method achieves competitive results compared to the state-of-the-art methods.

© 2022 Elsevier Ltd. All rights reserved.

1. Introduction

Over the last year, the COVID-19 pandemic has transformed the world, compelling almost all the governments around the world to take preventative measures against the virus spread [1]. Due to the fact that it can be transmitted without symptoms, COVID-19 is more infectious but has a lower mortality rate than other known viruses such as SARS and MERS. As a result, by the time the virus carrier are tested positive may have already spread the virus to a large number of others through close contact. The spread of the COVID-19 virus and the blockade had a significant negative effect on the global economy, which has developed into a concerning situation.

In addition to developing vaccines for healthcare systems [2] and automatically detecting COVID-19 cases [3–5], contact tracking is also a quite vital strategy for preventing the spread of COVID-19 that has become an urgent priority in the progress of COVID-19 epidemiological survey. As an effective way of viral prevention, contact monitoring tries to identify individuals who had intimate touch with the positive carriers [6]. Many governments employ technology to track and control the movements of anyone who may have come into contact with COVID-19.

Contact tracking is generally achieved through a manual interview of the sick individuals, conducted by the health authorities. The purpose of the interview is to obtain information regarding contacts the infected individual had with other individuals in the past 14–21 days (defined as the incubation period for COVID-19). Health officials can then utilize that information to evaluate a risk-score for each connections, based on the context (e.g., indoors/outdoors), duration, and proximity (distance between the contacts). Since most of the people they contact on a daily basis are not people they know, it is difficult to get them to recall exactly who they have met in the prior three weeks. Besides, an infected individual might have infected numerous others whom they cannot identify. Moreover, many subsequent interviews require a substantial staff of health professionals trained in the technique of manual contact tracing.

Researchers have been focusing on technological solutions to automate the contact tracing process with the target of quickly and reliably identifying contacts that might be at significant infection risk recently. The ubiquity of smartphones and their ability to keep track of their location (e.g., via GPS and WiFi), along with their in-built Bluetooth interface that allows communication and proximity detection with nearby smartphones, makes them ideal devices for

\footnote{Corresponding authors.}

\footnote{E-mail addresses: clowang@whu.edu.cn (X. Wang), wzy_hope@163.com (Z. Wang).}

https://doi.org/10.1016/j.patcog.2022.108908
0031-3203/© 2022 Elsevier Ltd. All rights reserved.
automated and reliable contact tracing. As a result, many smartphone contact tracing apps have been proposed [7], with some already deployed. Using the Bluetooth interface, these tracing apps automatically collect the contact data of their users-data to be frequently used in the event of a user being identified as infected with COVID-19.

The sparsity of GPS data and current inaccurate positioning algorithms lead to the inaccurate follow-up of confirmed COVID-19 patients and their contacts. To this end, we combine mobile GPS data with video monitoring data and propose a new task that group activity trajectory recovery to track COVID-19 contact. In this task, people move in groups through multiple cameras in crowded public environments. Fig. 1 illustrates the task and its challenges. People in a group often change their locations under different camera views, and the number of people in the group changes over time. We aim to associate every member of the group over the camera network with GPS information, so as to find people who have contact with the confirmed. The red circle represents COVID-19 confirmed, the yellow circle represents COVID-19 contact, and the green circle represents a healthy pedestrian.

To facilitate group activity trajectory recovery over camera network task, we propose a new dataset named GATR-GPS to simulate the real scene of COVID-19 contact tracking. Then, a novel coordinated optimization algorithm with spatio-temporal constraint table is proposed to efficient trajectory recovery of pedestrian trajectories.

The major contributions of this paper are summarized as follows:

- In this paper, we address a novel task named "group activity trajectory recovery" and screen suspected high-risk patients in the population by combining GPS data and video data to track COVID-19 contacts in an epidemiological survey. It combines efficiency and accuracy compared to existing methods, and the actual operation is more reliable and efficient.
- In order to deal with the task in the new scenario, we propose a high-quality dataset based on pedestrian spatio-temporal GPS information (GATR-GPS) to simulate the real-world scene of COVID-19 contact tracking. The most important characteristic of the dataset is that it has GPS information on the people in the group compared with the existing datasets.
- We propose a collaborative optimization method that takes both the spatio-temporal constraint and the person's image features into consideration. In particular, it extends the analysis of similarity measures between two pairs of objects to the combinatorial optimization of group behavioral trajectories under spatio-temporal behavioral constraints, where a spatio-temporal constraint table is used to constrain the state of pedestrians in the camera. Extensive evaluations demonstrate better performance than the state-of-the-art on our collected dataset as well as two public datasets.

The remainder of the paper is organized as follows. We review the related work in Section 2 and cover the proposed algorithm in Section 3. We present experimental results in Section 4. Finally, conclusion and future work are outlined in Section 5.

2. Related work

2.1. COVID-19 Contact tracking technology

Existing COVID-19 contact tracking techniques [8–10] can be divided into two types: manual interviewing and automatic tracking techniques. The manual survey of COVID-19 patients approach focuses on interviewing identified patients to investigate where they have been and who they have been in contact with at what time. The manual approach is laborious, and there are cases where patients forget or deliberately conceal the locations they have visited. The current COVID-19 automatic tracking method can be divided into two categories: tracking based on GPS data [11,12] and tracking based on Bluetooth data [13–15].

Manual interview surveys are time-consuming and inaccurate. Manual interview surveys are usually done through manual interviews with infected people by health authorities. The purpose of the interview is tantamount to collecting contact with other people during the past 1421 days. The contact information can be used to assess the risk of each contract. However, it is challenging for people to accurately recall each person that they may met in the last three weeks. In this context, we have been focusing on technological solutions to automate the contact tracing process with the target of quickly and reliably identifying contacts that might be at significant infection risk. Manual interview surveys are time-consuming and inaccurate.

At present, there is also research on using mobile apps to locate and track COVID-19 contacts [11,16]. However, the scarcity of GPS data and current inaccurate positioning algorithms make it impossible to effectively track contacts. The potential for contact tracking is radically changed by the advent of mobile technology, and it is
likewise a very effective method to track contacts in epidemic prevention and control [17–19].

The following comparisons were made for existing COVID-19 Contact tracking technology methods:

Manual method epidemiological survey approach: Interviews were conducted with confirmed patients to investigate where they had been and who they had contacted at what time, but the approach had stranger blindness and could not accurately recall everyone they might have met in the past three weeks.

Spatio-temporal concomitant approach [16]: Spatio-temporal GPS data analysis was performed on confirmed patients, and the method has a large error, with false alarms by analyzing GPS data errors at 600m.

Bluetooth data analysis method [20,21]: Spatio-temporal Bluetooth data analysis for confirmed patients, the crowd is complicated, the anti-interference ability is weak, and we need to make sure that the cell phone of confirmed patients is turned on Bluetooth.

Our method: Data analysis of confirmed patients by combining GPS data and video data, compared with the spatio-temporal accompanying way algorithm accuracy has improved, while video surveillance is more common in daily public places, the actual operation is more reliable and efficient.

2.2. Trajectory recovery

Widely adopted location-based services have accumulated large-scale human mobility data, which has great potential to benefit a wide range of applications, from personalized location recommendations to urban transportation planning. Since users may not allow the service provider to collect their locations continuously, individual trajectory records in such data are extremely sparse and unevenly distributed in time, which inevitably harms the performance of downstream applications, even when it has a notable user penetration rate and covers a long period of time [12,22,23].

Trajectory recovery [24,25] is still challenging for the following reasons: first of all, the high sparsity of both targeted and historical trajectories hinders us from inferring the missing locations by spatial-temporal constraints, i.e., how far a user can move in the unobserved periods. Because of the high uncertainty between two consecutive records in a sparse trajectory, a framework that can better model the mobility pattern and reduce the number of potentially visited locations is needed. The second point is how to distill periodic features from huge historical data effectively [13], considering that real-world historical data includes a great deal of noise.

2.3. Person re-identification

Computer vision work related to video surveillance has mainly focused on models for isolated targets, such as person re-identification (re-ID). re-ID is a cross camera tracking technology aiming at retrieving specific pedestrians [26,27]. It remains a challenging task due to the large number of unknown variations across camera views in human pose, illumination, viewing angle, lighting, background clutter, etc. Its difficulties are summarized as follows: First, a person’s appearance can change significantly due to alteration in lighting, position, view, and occlusion; Second, different people wearing similar clothing are difficult to distinguish in a realistic monitoring situation. The above challenges can lead to large intra-class variation, and subtle inter-class variation, affecting the final re-ID results. re-ID can be seen as a cross-camera retrieval problem, with the assumption that retrieved pedestrians exist in the gallery. However, in real-world surveillance scenario, pedestrians don’t always appear in the camera. Most of the existing methods [28–30] employ CNN to learn discriminative features or design various metric distances for better measuring the similarities between person image pairs. A network consistent re-identification method [31,32] considers explicitly maintaining the consistency of the results across the network to obtain the maximal correct matches for the whole camera network. The end-to-end consistency-aware deep learning method (CADL) [33] for person re-identification inconsistency problem was proposed by taking into account the hardware variability of different cameras, training images under different cameras in batches, adding person re-identification consistency information to the loss function construction, and iteratively optimizing the similarity matrix and data association information to obtain the global optimal match for the whole camera network.

Although the above methods are effective for single person re-identification, they all generate a robust description by mining useful clues from the given single image nor group. Motivated by this, we propose a new task that group activity trajectory recovery to recover the trajectory of contacts in a large-scale open scenario. In large-scale open scenarios, pedestrians are not always present in the camera and person’s appearance may change immensely. There are many pedestrians in urban surveillance that look similar to the target pedestrian. To this end, we propose a novel algorithm with spatio-temporal constraint table and coordinated optimization strategies that leads to efficient trajectory recovery of contact trajectories. Our approach has two key differences from such works: i) The method is to solve the problem that associating all the individual pedestrians under the camera networks. ii) The knowledge gained from the spatio-temporal constraints can be used to handle the case where the pedestrian does not appear in the camera.

3. Methodology

We present the overall framework of our approach in Fig. 2. In the following, we first define how the deep combined feature is modeled. Then, we introduce a combined optimization method based on spatio-temporal constraint (CO-SC) that allows us to exploit knowledge available about the entire pedestrian to better address the person re-identification. Finally, we describe how group matching is performed.

3.1. Problem formulation

In our scene, we consider a new task that group activity trajectory recovery over camera networks to track COVID-19 contacts. Assume there are m cameras in the camera network, and n persons walking in groups, continuously passing through multiple non-overlapping cameras. People come in and leave during the journey and the layout and membership changes. The number of possible camera pairs is C_m^n. We considered a general case in which the activity of a group person (the number of people is more than three) crosses multiple cameras uninterruptedly on urban roads and is a single walk. Group person re-identification is a problem of multi-pedestrian retrieval in the range of multi-camera. The purpose of our framework is to find the self-similar image data of the group person who appeared in the camera networks. There are COVID-19 infected people in the group and pass through all cameras along the path. The people who have close contact with the infected people are COVID-19 contact.

As shown in the Fig. 3. Blue circle is the entire set of combinations, green circle is set A, and yellow circle is set D. We define the following:

Possible combinations of trajectories: Combination of trajectories that do not contain pedestrians in their unpresented state (A),
Fig. 2. Illustration the framework of the combinatorial optimization based on spatio-temporal constraint for group activity trajectory recovery. The red circle represents the confirmed COVID-19, the yellow circle represents the COVID-19 contact, and the green circle represents the healthy pedestrian.

Table 1

| route | number of person | number of camera |
|-------|------------------|------------------|
| 1     | 8                | 30               |
| 2     | 10               | 10               |
| 3     | 4                | 10               |
| 4     | 4                | 10               |

Table 2

| Data name       | Size                                           |
|-----------------|------------------------------------------------|
| Video data      | 45 hours of video data in total of 30 cameras   |
| Training data   | 27939 images                                   |
| Test data       | 1500 images volunteers on 30 cameras           |
| Single image data | 11631 images                                   |
| GPS data        | 20643 GPS data of 20 volunteers walking         |

False visual object trajectory combinations: Trajectory combinations that do not contain pedestrians in their non-occurring state \( B = A - D \).

3.2. Spatio-temporal Constraint table

In our created datasete, 20 persons go through the course on 4 different routes with mobile devices in the GATR-GPS datasete, and the persons walk together on the same road. The route distribution shown in table 2. Route 2, 3 and 4 refer to the front, middle and behind the course, respectively, of the 10 in 30 cameras. We collect the persons’ positioning data information every 5 seconds and a total of 20,643 trajectories. As the camera coordinates use WGS-84 coordinates (international standard coordinate system) and the person coordinates use GCJ-02 coordinates, we unify longitude and latitude to the GCJ-02 coordinate system first. Then, we compare the collected persons’ position information. The Haversine formula [34] to calculate the shortest distance between two points on the surface of a sphere. For example, the person’s longitude and latitude at t times are \( x_1 \) and \( y_1 \) and the camera’s longitude and latitude are \( x_2 \) and \( y_2 \). The distance between the pedestrian and the
camera is $D$ at $t$ time. We introduce the Haverson sine formula below, where the long axis $R$ is 6378.137 km.

If the pedestrian’s $D$ under the camera computer is less than or equal to $d$, the value is 1, otherwise the value is 0. The state of pedestrian $n$ under all cameras is $f(n)$ When comparing to the size of thresholds $D'$ and $D$, we can obtain the status information $f(n)$ of whether the person passes the camera. Using $M$ to represent spatio-temporal constraint, the $f(n)$ in $M$ denotes the binary status information of the person. The number 1 denotes that the person appears under the camera, and 0 denotes that the person does not appear under the camera. If the pedestrian’s $D$ under the camera computer is less than or equal to $D'$, the value is 1, otherwise the value is 0. The state of pedestrian $n$ under all cameras is $f(n)$.

We can see in Fig. 3 that the images left by each person under each camera can form a class and the class of each person can form a combination. According to the person constraint matrix, the possible combination of persons can be obtained.

\[
\frac{\text{haversin}(\frac{D}{R})}{\text{haversin}(x_2 - x_1) + \cos x_2 \cos x_1 \text{haversin}(y_2 - y_1)} = \text{haversin}(\theta) = 1 - \cos(\theta)/2.
\]

(1)

(2)

3.3. Matrix learning

Take a subset of person images from different cameras as training samples and feed them into the CNN network for training. Then, take the output of $F_i^a$ as the features and obtain a distance matrix $d$ for each image pair in different cameras. More specifically, we use the cosine metric to calculate the distance between the features $F_i^a$ and $F_j^b$:

\[
\cos = \frac{F_i^a F_j^b}{\|F_i^a\| \|F_j^b\|}.
\]

(3)

3.4. Similarity distance matrix representation

There are $m$ cameras and $n$ persons in the camera networks, the person’s image in the camera is represented by $p_i^n$, the superscript refers to the camera ID, and the subscript refers to the person ID. CNNs are used to extract the features of any input images in different cameras and the cosine similarity distance is used to obtain the distance matrix $d$, with the row index representing the persons from camera a, and the column index representing the persons from camera b. Then, the $(i, j)$th cell in $d$ denotes the similarity distance between person $p_i^n$ and $p_j^n$. In the experiment, we use a common CNN network in person re-identification to conduct comparative experiments, such as Resnet50, a backbone network in PCBDenseNet-121.

\[
d_{i,j}^a = \begin{bmatrix}
d_{i1}^a & d_{i2}^a & \cdots & d_{ij}^a \\
d_{i1}^b & d_{i2}^b & \cdots & d_{ij}^b \\
\vdots & \vdots & \ddots & \vdots \\
d_{i1} & d_{i1} & \cdots & d_{ij}
\end{bmatrix}
\]

(4)

3.5. Deep combined features

Fig. 5 shows that the images left by each person under each camera can form a class, and the classes of all people can form a combination. The gray rectangular box represents a class, and the circles represent the images of pedestrians in that class. The different colored circles represent different pedestrians, and the cross represents that the person in that line does not appear under the camera. Intra-class represents the calculation of distances between images inside the gray rectangle, and inter-class represents the calculation of distances between images in different gray rectangles.

The deep combined features contain four factors. The mean value and the variance of the combined within-class features represent the distribution of features of the combined within-class features. The mean value and the variance of the combined between-class features represent the distribution of features in the combined between-class features. Calculating the distance between images is the basis for calculating the four features, and the yellow line between images represents the distance between images, denoted by $d$. By using the GPS data obtained from pedestrian collection to construct a spatio-temporal constraint table for travelers, if there is no GPS data, all combinations need to be traversed, represented by blue circles. Using a spatio-temporal constraint table can constrain away impossible combinations, and only the set A needs to be calculated, represented by green circles, which can improve the efficiency of retrieval.

3.5.1. Mean value in the combined within-class features

We present the mean value in the combined within-class features using $\theta(i)$, which can be calculated by the median distance value of all classes $\theta_i$ according to the similarity distance matrix representation $d_{i,j}^a$, $n$ is the number of person. The blue line indicates the within-class distance in Fig. 3 for deep combination features calculate. It calculates the distance between adjacent images in the within-class.

\[
\theta(i) = \frac{\text{med}(\theta_i)}{\theta_i} = \frac{1}{n} \sum_{j=1}^{n} d_{ij}^{(1)}.
\]

(5)

where $i$ indicates query image and $j$ indicates gallery image.

3.5.2. Variance in combined within-class features

We present the variance in the combined within-class feature using $\sigma(i)$, which represents the measures of within-class data dispersion and can be obtained as:

\[
\sigma(i) = \frac{1}{n} \sum_{j=1}^{n} (d_{ij}^{(1)} - \theta(i))^2.
\]

(6)

3.5.3. Mean value in combined between-class

We present the mean value in the combined between-class feature using $\theta(k)$, which can be calculated by the median distance value of all the between-class distances $\theta_k$ according to the similarity distance matrix representation $d_{i,j}^{(2)}(k, (k+1), m$ represents the number of cameras. The yellow line indicates the distance between classes in Fig. 5 for deep combined feature calculations. It calculates the distance of the images between adjacent persons.

\[
\theta(k) = \text{med}(\theta_k) = \frac{\sum_{k=1}^{m-1} \frac{1}{n^2} X_k}{m - 1}.
\]

(7)
where
\[ X = \sum_{i=1}^{n} \sum_{j=1}^{n} d^{(2)}(k_i, (k + 1)j). \]  \hfill (8)

### 3.5.4. Variance in the combined between-class feature

We present the variance in the combined between-class feature using \( \sigma^{(2)} \) to present, which represents the measures of between-class data dispersion and can be obtained as:
\[ \sigma^{(2)} = \frac{1}{n} \sum_{i=1}^{n} \left( d^{(2)}(k_i, (k + 1)j) - \theta^{(2)} \right)^2. \]  \hfill (9)

where \( k \) denotes the number of inter-combined sequences in the combination.

### 3.6. Dichotomous model

According to the deep combined features, we take the group person re-identification as two classification problems and train an classification model to find the correct match according to the deep combined features. The input of the classification model is \( t \) sample points, and \( h_i \) contains the deep combined features. The constraint functions are as follows, where \( w \) is an \( n \)-dimensional vector, \( b \) is offset, and \( z_j \) represents two different classes. The classification model we choose Decision tree, SVM, knn and Random forest to train different model.

We transform the problem of solving the optimal combination into a binary classification problem using the SVM method. The learning goal of the linear classifier is to find a classification hyperplane in the data space, formulated as:
\[ f(x) = w^T x + b. \]  \hfill (10)

Then the minimum value of the interval of the hyperplane \((w, b)\) as a function of all sample points \((x_i, y_i)\) is then the interval of the hyperplane as a function of the training data set \(T\):
\[ \min \{ y_i(f(x_i))(i = 1, \ldots, n) \}. \]  \hfill (11)

For all training data points, the following conditions must be satisfied:
\[ y_i(w^T h_i + b) \geq 1(i = 1, \ldots, t). \]  \hfill (12)

\( i \) is the sample, and \( T \) is the training sample and is expressed by:
\[ T = \{ (h_1, z_1), \ldots, (h_t, z_t), \ldots, (h_i, z_i) \} \in \mathbb{R}^{m} \times \mathbb{Y}. \]  \hfill (13)

where \( \mathbb{R}^m \) represents the set of samples from \( h_1 \) to \( h_t \) in the training set \( T \).

\[ h_i \in \mathbb{R}^m, z_j = y = \{ 1, -1 \}. \]  \hfill (14)

The \( M \) denotes the set of all \( M \)-dimensional vectors. \( R \) stands for representative set of real numbers, \( t \) indicates the number of samples, and \( y \) refers to the sample label 1 or 0.

Due to the large gap between the number of true and false combinations, one true combination and one hard-to-sample false combination are extracted as a training sample for each combination calculated.

Thus the minimum distance from each support vector to the hyperplane can be expressed as:
\[ \min \frac{1}{2} \| w \|^2. \]  \hfill (15)

The SVM is used to classify the combination features and extract the probabilistic ranking results of the real combination.
\[ L(p, C) = [C_1, C_2, \ldots, C_k]. \]  \hfill (16)

where \( C \) represents the combined result, and \( k \) represents the top \( k \) sorted results.

### 4. Experiments

#### 4.1. Datasets

Due to the privacy problems of mobile phone data during the epidemic situation, We propose a high-quality group re-identification dataset based on pedestrian spatio-temporal GPS information (GATR-GPS) to simulate real scene. To validate the effectiveness of our approach, we built a GATR-GPS dataset, and the public benchmark subset data in DukeMTMC-reID and Market1501 needs to be satisfied with the persons appearing under all cameras and persons working as a group.

##### 4.1.1. Market-1501

The Market-1501 dataset [36] has 32,668 bounding boxes of 1501 persons captured from 6 cameras in front of a supermarket at Tsinghua University. We choose 99 persons who appear in six cameras. A total of 2,548 bounding boxes were used for training, and 594 bounding boxes were used in testing.

##### 4.1.2. DukeMTMC-reID

The DukeMTMC-reID dataset [39,40] has 36,441 bounding boxes of 1,812 persons captured from 8 cameras at Duke University. As no person passed all 8 cameras, we chose 14 persons who passed through five cameras. A total of 280 bounding boxes were used for training, and 70 bounding boxes were used in testing.

##### 4.1.3. GATR-GPS dataset

Due to the need for uncertainty in the process of dense infection during epidemiological investigation, as well as the unavailability of data collection in real scenarios of COVID-19 infection process. There are also no international published data sets available. while, a new dataset called GATR-GPS is constructed to simulate a realistic scenario of COVID-19 contact tracking. The GATR-GPS dataset consists of a single person re-id dataset and a group re-id datasets. It contains 20 identities and each identity has several images from 30 disjoint cameras on the Wuhan University campus and has 20,643 bounding boxes. It covers 3 hours in the morning of the day, and YOLOv3 [42] is utilized for pedestrian detection. Table 3 shows the comparison between the GATR-GPS dataset and the existing person re-identification dataset and the group re-id dataset. The most important characteristic of the dataset is that it has GPS information of the person and the person walking as a group.

We also conducted experiments on the standard protocol where rank-1 accuracy and mean average precision (mAP) were measured to show that validate the effectiveness of our method in the single person re-id dataset. To verify the complexity of the algorithm the running time metric is used for evaluation. Based on the test subset of the public data set, the data of pedestrians monitored by video in the public data set is selected to simulate the crowd multi camera scene. Compared with the existing person re-identification

| Dataset | Cameras and Identities | GPS data |
|---------|------------------------|----------|
| VIPeR [35] | 2 and 632 | No |
| Market1501 [36] | 6 and 1501 | No |
| WARD [37] | 3 and 70 | No |
| CUHK03 [38] | 10 and 1467 | No |
| DukeMTMC-reID [39,40] | 8 and 1812 | No |
| MSMT17 [41] | 15 and 4101 | No |
| GATR-GPS | 30 and 20 | Yes |
data set, the self built crowd pedestrian recognition data set has the biggest feature of spatio-temporal information and the application scene is crowd multi camera. As shown in Table 3, A total of fifty video frames were selected from each video, for a total of 30 cameras, for a total of 1500 test group images. Although the test video frame image data is only 1500, there are multiple pedestrians in one image, and the image data of a single pedestrian is 11631 images.

4.2. Experimental settings

Due to the fact that the images in the dataset were collected from different locations with, the intensity and quality of images vary considerably. Nevertheless, we avoid extensive pre-processing of our images in the dataset to gain improved generalization ability of our proposed network model. This in turn makes our model further robust to artifacts and noises present in the images while extracting feature embeddings from the input images.

Due to the limitation of GPU memory, it is impossible to feed all person images for all cameras into the CNN at once and calculate deep combined features, so in our experiments, we considered 5 cameras and 5 person images to feed into our CNN each time. We set the initial learning rate as 0.01 and reduced by a factor of 0.1 every 59 batches. Used the ResNet50, DenseNet-121 and PCB networks as the pretrained models to extract image features for comparison.

4.3. Experimental results

4.3.1. Regularity discovery

Since the D threshold is much larger than the camera range view, it is found that there is data drift in the latitude and longitude coordinates obtained by using the Gaode API through data collection. Therefore, the obtained longitude coordinates are corrected by subtracting (0.0065, 0.0060) from the obtained results. The final modified distance threshold was obtained at 600m

![Fig. 6. Illustration the histogram of the variation of accuracy with distance threshold.](image1)

![Fig. 7. represents the distribution of combination optimization features after extracting features using ResNet50 network on the GATR-GPS dataset.](image2)
with a maximum accuracy of 82.6%. The experimental results are shown in the Fig. 6.

The horizontal coordinates have the number of the combination, the vertical coordinates indicate the distance feature values, the distance feature data are projected to 1-100 after analysis. (1) in Figs. 8–10 represents the mean comparison plot within the true/false combination feature sequence; (2) represents the mean comparison plot between the true/false combination feature sequence; (3) indicates the mean comparison graph between real/false combination feature sequences; (4) indicates the variance comparison graph between real/false combination feature sequences. From the graphs, we can see that the variance within the real combination sequence is lower than that of the spurious combination, and its feature distribution is relatively tight. The variance within the real combination sequence is greater than that of the spurious combination, and the feature distribution is relatively sparse. Meanwhile, a p-value of 0.03 (less than 0.05) was obtained by analyzing the real/false sequence combination feature data according to the Student’s t test method, which indicates that there is a significant difference in the distribution of the real/false sequence combination feature data.

4.3.2. Comparison with contact tracking methods

Because tracking based on Bluetooth data requires the user to open the Bluetooth device, tracking based on GPS data is more widely used. Therefore, we chose the GPS data-based tracking method for comparison. As shown in the following table, randomly take the GPS data and video data at the same moment, calculate their errors, traverse them 10 times, and average their errors, as shown in the Table 4. The error of GPS data is greater than 200m, while the video-based data error is less than 1m. Fig. 10 shows the visualization of the distance between adjacent pedestrians calculated in the image.

| dataset     | Range of error |
|-------------|---------------|
| GPS data    | ≥ 200m        |
| Video data  | ≤ 1m          |

Table 4: Comparison of the average error of two kinds of data.

4.3.3. Comparison with person re-identification methods

The GATR-GPS dataset contains GPS data and the main purpose of GPS data is to determine whether pedestrians appear under the camera or not. Since traditional pedestrian re-identification methods all assume that pedestrians appear under the camera, the results in Table 4 are experiments with both GPS data, which are relatively competitive results compared to the benchmark CADL method, with a 7.35% improvement on Rank 1 and a 5.81% improvement on mAP. If the comparison method is not experimented on the premise of GPS, the experimental results can be improved by another 50% over the existing results, and 50% of the camera network in the case of pedestrians do not appear. In the comparison of the running times of state-of-the art algorithms

![Fig. 8. represents the distribution of combination optimization features after extracting features using PCB network on the Market-1501 dataset.](image-url)
in the GATR-GPS dataset (single), the shortest running time was 4.481s with ResNet50 + Cosine distance, and the longest running time was 8.861s with ResNet50 + CO-SC (S). Although the running time is long (4.374s) with our method, the accuracy is increased mAP/Rank1 by 11%/18.29% compared to other methods.

Tables 6 and 7 show some subjective results of the proposed method on the Market-1501 and DukeMTMC-reID datasets. There is no GPS data in either of the public datasets, which simulate pedestrians passing through all camera scenes. For each dataset,

**Table 5**
Performance comparison of state-of-the-art algorithms for the GATR-GPS dataset (single).

| Methods                  | rank1   | mAP    | Time   |
|--------------------------|---------|--------|--------|
| ResNet50 + Cosine distance | 71.51   | 69.15  | 4.481s |
| PCB [43] + Cosine distance | 72.69   | 70.34  | 4.527s |
| DenseNet-121 + Cosine distance | 73.56   | 63.54  | 4.467s |
| CADL [33]                | 82.45   | 74.34  | 5.216s |
| PCB [43] + CO-SC (S)     | 74.31   | 73.25  | 8.736s |
| DenseNet-121 + CO-SC (S) | 78.12   | 76.72  | 8.783s |
| ResNet50 + CO-SC         | 89.80   | 80.15  | 8.861s |

**Table 6**
Performance comparison of state-of-the-art algorithms for the Market1501 dataset.

| Methods                  | rank1   | mAP    | Time   |
|--------------------------|---------|--------|--------|
| ResNet50 + Cosine distance | 83.65   | 74.53  | 2.145s |
| PCB [43] + Cosine distance | 84.82   | 76.57  | 2.436s |
| DenseNet-121 + Cosine distance | 76.65   | 67.78  | 2.367s |
| CADL [33]                | 84.45   | 77.65  | 4.671s |
| PCB [43] + CO-SC (S)     | 89.23   | 82.35  | 6.612s |
| DenseNet-121 + CO-SC (S) | 85.56   | 80.43  | 6.346s |
| ResNet50 + CO-SC         | 88.51   | 81.56  | 6.781s |
we give one example and its top 1 results evaluated on three kinds of networks to extract image features, and the distance metric uses cosine distance. Compared to the benchmark method, the best results were obtained for the PCB trained network on two publicly available datasets. Compared to the CADL method, a 4.70%/4.78% improvement in the mAP/Rank1 was achieved on the Market-1501 dataset and a 10.4%/5.95% improvement in the mAP/Rank1 was achieved on the DukeMTMC-reID dataset. Experiments show that our method is effective when considering the information carried by neighboring pedestrians. From Table 6, in the comparison of the running times of state-of-the-art algorithms in the Market-1501 dataset, compared with the running times of different classification methods, the shortest running time was 2.145s with ResNet50 + Cosine distance, and the longest running time was 6.781s with ResNet50 + CO-SC (S). Although the running time is long (4.636s) with our method, the accuracy is increased mAP/Rank1 by 7.03%/4.86% compared to other methods. In the comparison of the running times of state-of-the-art algorithms in the DukeMTMC-reID dataset, compared with the running times of different classification methods, the shortest running time was 1.892s with ResNet50 + Cosine distance, and the longest running time was 6.891s with ResNet50 + CO-SC (S). Although the running time is long (4.999s) with our method, the accuracy is increased mAP/Rank1 by 5.65%/6.56% compared to other methods.

4.3.4. Performance Analysis with Different Parameters

For the classification models, we chose decision tree (D), KNN (K), random forest (R) and SVM (S) to train different models. From Table 8, it can be seen that when using the ResNet50 network to extract features and calculate the combined features, the best performance can be achieved by using SVM as a classifier to mAP/Rank1 by 75.31%/76.24%. In the classification network using SVM for classification and the training network using ResNet50, DenseNet-121, and PCB networks for training comparison, experiments can be found using PCB network training compared to the ResNet50 network in mAP/Rank1 improved by 1.23%/2.44%. In the comparison of the running times of different classification methods in the GATR-GPS dataset (single), the shortest running time was 7.273s with ResNet50 + CO-SC (R), and the longest running time was 8.861s with ResNet50 + CO-SC (S). Although the running time is long with the SVM method, the accuracy is higher compared to other methods.

4.3.5. Trajectory visualization analysis

This paper mainly recovers pedestrian trajectories by combining GPS data with video data. The visual result diagram after visual trajectory recovery is shown in Fig. 11. The numbers in the picture represent camera numbers. While green numbers represent correct matches, red numbers represent incorrect matches. The figure shows the retrieval results of rank1 for 4 pedestrians under 30 cameras, and the cameras not listed are those where the pedestrian does not appear.

5. Conclusions and future work

This paper addresses the problem of inaccurate contact tracking of close patients using cell phone GPS during the epidemiological investigation of the COVID-19 outbreak, while surveillance cameras are deployed in the city and pedestrians can be easily identified. Therefore, this paper proposes a new problem based on group activity trajectory recovery by combining GPS data and video data to achieve the tracking of close contact patients, and constructs a high-quality GATR-GPS dataset to simulate the tracking of close contact patients in a realistic scenario. A Combinatorial Optimization method (CO-SC) is also proposed to use GPS data to locate whether pedestrians appear under the camera, and to construct a spatio-temporal constraint table to reflect the status of whether all pedestrians appear under the camera under the video, while removing impossible combinations to save computational complexity. Also, we use video image data to identify close patients by traversing all possible combinations and using a second classifier to find the true combination. Both pedestrians are in the correct matching relationship under the camera network. Combined with the combination constraint table can recover the trajectory of pedestrians, when the confirmed patients appear under the camera at the same time with other pedestrians in a certain time interval, the pedestrian is considered as to be close contact. Experimental results on both the GATR-GPS dataset (GPS data exists) and the publicly available dataset (GPS data is not provided) validate the effectiveness of the proposed method.

Although the proposed method provides a new idea and an improved way to track close contacts, it has some limitations. On the one hand, due to the limited resources of collecting GATR-GPS datasets, it does not fully reflect the complex environment of real scenes. On the other hand, the complexity of calculating the num-

---

Table 7

| Methods          | rank1 | mAP  | Time |
|------------------|-------|------|------|
| ResNet50 + Cosine distance | 79.67 | 74.73 | 1.892s |
| PCB [43] + Cosine distance | 85.43 | 76.52 | 1.923s |
| DenseNet-121 + Cosine distance | 80.53 | 72.26 | 1.678s |
| CADL [43] | 82.32 | 71.25 | 4.212s |
| PCB [43] + CO-SC(S) | 88.27 | 81.65 | 6.234s |
| DenseNet-121 + CO-SC(S) | 81.78 | 79.61 | 6.452s |
| ResNet50 + CO-SC | 86.23 | 80.38 | 6.891s |

Table 8

| Methods          | rank1 | mAP  | Time |
|------------------|-------|------|------|
| ResNet50 + CO-SC (D) | 58.13 | 51.46 | 8.194s |
| ResNet50 + CO-SC (K) | 60.61 | 56.36 | 8.851s |
| ResNet50 + CO-SC (R) | 64.52 | 62.44 | 7.273s |
| ResNet50 + CO-SC (S) | 76.24 | 75.31 | 8.861s |
| DenseNet-121 + CO-SC (S) | 70.47 | 69.12 | 8.783s |
| PCB + CO-SC (S) | 78.68 | 76.54 | 8.736s |
ber of combinations is relatively high, and while it is increasing with the number of people and cameras, there are still gaps in practical applications.

In order to improve this work, we will focus on reducing the complexity of the calculation of the combination number in order to improve the efficiency of the actual application. In addition, we will further explore dense joint tracking on complex paths to improve the level of application in dealing with actual complex scenes.

Declaration of Competing Interest

No conflict of interest exists.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the National Science Foundation of China U1803262 and U1903214.

References

[1] P.L. Curjel, A.D. Coman, A. Panchenko, O.C. Fodor, L. Rajtjo, Death anxiety, death reflection and interpersonal communication as predictors of social dis- tance towards people infected with covid 19, Current Psychology (2021) 1–14.
[2] S.-C. Wu, Progress and concept for covid-19 vaccine development, Biotechnology journal (2020).
[3] A. Ondrus, S. Aqil, T. Trongtrirakul, A.K. Laour, Automatic covid-19 lung infected region segmentation and measurement using ct-scans images, Pattern recognition 114 (2020) 107747.
[4] M. Shorufazzaman, M.S. Hossain, Metacovid: A siamese neural network framework with contrastive loss for n-shot diagnosis of covid-19 patients, Pattern Recognition 113 (2020) 107700.
[5] Z. Wang, Y. Xiao, Y. Li, J. Zhang, F. Lu, M. Hou, X. Liu, Automatically discrimi-

nating and localizing covid-19 from community-acquired pneumonia on chest x-rays, Pattern recognition 110 (2020) 107613.
[6] A. Urbaszewski, Y.J. Lee, Information technology and the pandemic: a prelimi-

nary multinational analysis of the impact of mobile tracking technology on the covid-19 contagion control, European Journal of Information Systems 29 (4) (2020) 405–414.
[7] J.A. del Peral-Rosado, R. Raulefs, J.A. López-Salcedo, G. Seco-Granados, Survey of cellular mobile radio localization methods: From 1g to 5g, IEEE Commu-

nications Surveys & Tutorials 20 (2) (2017) 1124–1148.
[8] B. Amen, S. Faiz, T.-T. Do, Big data directed acyclic graph model for real-time

covid-19 twitter stream detection, Pattern Recognition 123 (2022) 108404.
[9] W.M. Shaban, A.H. Rabie, A.I. Saleh, M. Abou-Elouid, Accurate detection of
covid-19 patients based on distance biased naive bayes (dbnb) classification

strategy, Pattern Recognition 119 (2021) 108110.
[10] K. He, W. Zhao, X. Xie, W. Ji, M. Liu, Z. Tang, Y. Shi, F. Shi, Y. Gao, J. Liu et al., Synergistic learning of lung lobe segmentation and hierarchical multi-instance classification for automated severity assessment of covid-19 in ct images, Pat-

tern recognition 113 (2021) 107782.
[11] Y. Sweeney, Tracking the debate on covid-19 surveillance tools, Nature

Machine Intelligence 2 (6) (2020) 301–304.
[12] C. Chen, A.C. Viana, M. Fiore, C. Sarraute, Complete trajectory recreation from

sparse mobile phone data, EPJ Data Science 8 (1) (2019) 1–24.
[13] E. Hernández-Orallo, C.T. Calafate, J.-C. Cano, P. Manzi, Evaluating the effec-

tiveness of covid-19 bluetooth-based smartphone contact tracing applications,

Applied Sciences 10 (2) (2020) 7113.
[14] Z. Shao, G. Cheng, J. Ma, Z. Wang, J. Wang, D. Li, Real-time and accurate uav

pedestrian detection for social distancing monitoring in covid-19 pandemic,

IEEE transactions on multimedia 24 (2021) 2069–2083.
[15] V. Dentauro, P. Giglio, D. Impedovo, L. Moretti, G. Pirlo, Aucos resnet: an end-
to-end network for covid-19 pre-screening from cough and breath, Pattern Recognition 127 (2022) 108656.
[16] J. Abele, M. Bäcker, U. Buermeyer, H. Zillesen, Covid-19 contact tracing and data protection can go together, JMRI mHealth and ultraHealth 8 (4) (2020) e19359.
[17] R. Gupta, M. Bedi, P. Goyal, S. Wadhiera, V. Verma, Analysis of covid-19 tracking

tool in india: Case study of aarogya setu mobile application, Digital Govern-

ment: Research and Practice 1 (4) (2020) 1–9.
[18] M. Zens, A. Brammertz, J. Herpich, N. Südkamp, M. Hinterseer, App-based

tracking of self-reported covid-19 symptoms: analysis of questionnaire data, Journal of medical Internet research 22 (9) (2020) e21956.
[19] M. Shorufazzaman, M.S. Hossain, Metacovid: A siamese neural network frame-

work with contrastive loss for n-shot diagnosis of covid-19 patients, Pattern recognition 113 (2021) 107700.

Chao Wang, received his B.S. degree in Computer Sci-

ence and Technology from East China University of Tech-

nology in 2017. His currently pursuing his Ph.D. degree in

National Engineering Research Center for Multime-

time Software, School of Computer Science, Wuhan Uni-

versity, China. His research interests include person re-

identification, multi-target tracking, and pedestrian trajec-

try recovery. Contact him at snipercwang@whu.edu.cn.
Xiaochen Wang, received his B.S. degree in cartography and geography information system from Wuhan University in 2003, and Ph.D. degree in communication and information system from Wuhan University in 2011. Since 2011, he has worked at the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University. His research interests include audio compression and audio signal processing. He is the corresponding author of this article. Contact him at clowang@163.com.

Zhongyuan Wang, received the Ph.D. degree in communication and information system from Wuhan University, Wuhan, China, in 2008. He is also directing four projects funded by the National Natural Science Foundation Program of China. He received the Ph.D. degree in communication and information systems from Wuhan University, Wuhan, in 2008. He is a Member of the IEEE. He is the corresponding author of this article. Contact him at wzy_hope@163.com.

Wenqian Zhu, received the M.S. degree from the Wuhan University of Technology, Wuhan, China, in 2016. He is currently working toward the Ph.D. degree in pattern recognition and image retrieval with the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, Wuhan. His current research interests include vehicle and person reidentification, pattern recognition and machine learning, and intelligent video surveillance. Contact him at zwq_pi@whu.edu.cn.

Ruimin Hu, received the B.S. and M.S. degrees from the Nanjing University of Posts and Telecommunications, in 1984 and 1990, respectively, and the Ph.D. degree from Huazhong University of Science and Technology, Wuhan, China, in 1994. He is currently a professor at the School of Computer Science, Wuhan University, China, and before that was the Dean of the School of Computer Science and the Director of the National Engineering Research Center for Multimedia Software at Wuhan University, China. His current research interests include multimedia network communication, security and surveillance technology, and big data analytics. He is a Senior Member of the IEEE. Contact him at hrm@whu.edu.cn.