milliTRACE-IR: Contact Tracing and Temperature Screening via mm-Wave and Infrared Sensing

Marco Canil[†], *Graduate Student Member, IEEE*, Jacopo Pegoraro[†], *Graduate Student Member, IEEE* and Michele Rossi, *Senior Member, IEEE*

[†] These authors contributed equally.

***Abstract*—In this work, we present milliTRACE-IR, a joint mm-wave radar and infrared imaging sensing system performing unobtrusive and privacy preserving human body temperature screening and contact tracing in indoor spaces. Social distancing and fever detection have been widely employed to counteract the COVID-19 pandemic, sparking great interest from academia, industry and public administrations worldwide. While most solutions have dealt with the two aspects separately, milliTRACE-IR combines, via a robust sensor fusion approach, mm-wave radars and infrared thermal cameras. The system achieves fully automated measurement of distancing and body temperature, by jointly tracking the faces of the subjects in the thermal camera image plane and the human motion in the radar reference system. It achieves decimeter-level accuracy in distance estimation, inter-personal distance estimation (effective for subjects getting as close as 0.2 m), and accurate temperature monitoring (max. errors of 0.5 °C). Moreover, milliTRACE-IR performs contact tracing: a person with high body temperature is reliably detected by the thermal camera sensor and subsequently traced across a large indoor area in a non-invasive way by the radars. When entering a new room, this subject is re-identified among several other individuals with high accuracy (95%), by computing gait-related features from the radar reflections through a deep neural network and using a weighted extreme learning machine as the final re-identification tool.**

***Index Terms*—Indoor human sensing, mm-wave radars, thermal camera, temperature screening, person re-identification, extreme learning machines.**

I. Introduction

In this work, we are concerned with the design of a real-time, integrated radio and infrared sensing system to jointly perform unobtrusive elevated skin temperature screening and privacy preserving contact tracing in indoor environments.

Lately, *social distancing* has become a primary strategy to counteract the COVID-19 infection. Many research works [1], [2] have shown that it is an effective non-pharmacological approach and an important inhibitor for limiting the transmission of many contagious diseases such as H1N1, SARS, and COVID-19. Along with social distancing, *elevated skin temperature detection* and *contact tracing* have proven to be key to effectively contain the pandemic [3]. However, available methods to enforce these countermeasures often rely on RGB cameras and/or apps that need to be installed and continuously run on people's smartphones, often rising privacy concerns [4]. Moreover, currently adopted methods to screen people's temperature require individuals to stand in front of a thermal sensor, which may be impractical in heavily frequented public places.

Here, we design and validate milliTRACE-IR, a joint mm-wave radar and infrared imaging sensing system that performs unobtrusive and privacy preserving human body temperature screening and contact tracing in indoor spaces. Next, its main components are discussed, emphasizing their novel aspects and the joint processing of the acquired sensor data.

mm-Wave radar: The radar analyzes the reflections of a transmitted mm-wave signal off the individuals that move in the monitored environment, returning *sparse point-clouds* that carry information about the subjects' locations and the velocity of their body parts. A novel point-cloud clustering method is designed, combining Gaussian mixtures [5] and the density-based DBSCAN [6] algorithm, to distinguish the mm-wave radio reflections from the subjects, as they move as close as 0.2 m to one another. The so obtained point-cloud clusters are used to track the subjects' positions in the physical space by means of a Kalman Filter (KF) [7], and to obtain their gait-related features through a deep-learning based feature extractor. Finally, a novel person re-identification algorithm is proposed by exploiting weighted extreme learning machines (WELM).

Thermal camera: The infrared imaging system, or thermal camera (TC), returns images whose pixels contain information on the *temperature of the objects* in the TC field of view (FoV). To measure the subjects' temperature, at first, YOLOv3 [8] is used to perform face detection in the TC images, by bounding those areas containing a human face. Hence, the obtained bounding boxes are tracked through an Extended Kalman Filter (EKF) [9] and the subjects' temperature is estimated by accumulating readings for each EKF track, according to a dedicated estimation and correction procedure. Through the EKF, the subject's distance from the TC is also estimated from the size of the corresponding bounding box. This is achieved by considering the non-linear part of the EKF, which is approximated by fitting a function over a set of experimental data points.

Radar and thermal camera data fusion: Tracks in the radar reference systems are associated with those in the TC image plane via an original algorithm that finds optimal matches for the readings taken by the two sensors, through their *joint* analysis. This makes it possible to take temperature measurements from a subject and reliably associate them with the highly precise tracking of his/her movement performed by the radar. In addition, the joint analysis of radar and TC data allows refining the temperature estimated through the TC: to mitigate the influence of the distance on the temperature readings [10], a regression function that provides temperature correction coefficients is fit from training data. The final temperatures are obtained using such function with the accurate distances retrieved from the radar.

Hence, once a subject's temperature is measured, it is associated with the corresponding radar track and the subject's movements and contacts inside the building are accurately monitored, by re-identifying the subject as he/she moves across the FoV of different radar devices. To the best of our knowledge, milliTRACE-IR is the first system that achieves temperature screening and human tracking through the joint analysis of radar and TC signals. Furthermore, it concurrently performs body temperature screening and contact tracing, while these aspects have been previously dealt with separately. A sensible usage model for the system is as follows: the TCs shall be deployed in strategic locations to allow an effective temperature screening, such as facing the building/room entrance, to ensure that people's faces are seen frontally for a reasonable amount of time, and that their TC images are only taken when they enter or leave the building/room. On the other hand, the radar can be utilized to track the subjects while moving inside the monitored indoor space. This ensures higher privacy with respect to RGB camera systems.

The main contributions of our work are:

1) milliTRACE-IR, a joint *mm-wave radar and infrared imaging sensing system* that performs unobtrusive and privacy preserving human body temperature screening and contact tracing in indoor spaces is designed and validated through an extensive experimental campaign.
2) A novel *data association* method is put forward to robustly associate tracks obtained from the mm-wave radar and from the TC, where the radar returns the people coordinates in the physical space and the TC identifies people's faces in the thermal image space. The achieved precision and recall in the associations are as high as 97%.
3) An original *clustering algorithm for mm-wave point-clouds* is devised, making it possible to resolve the radar reflections from subjects as close as 0.2 m.
4) A new WELM based *person re-identification* procedure is presented. The WELM is trained at runtime on previously unseen subjects, achieving an accuracy of 95% over six subjects with only 3 minutes of training data.
5) A novel method is designed to perform *elevated skin temperature screening* as people move freely within the FoV of the TC, without requiring them to stop and stand in front of the thermal sensor. For this, a dedicated approach is presented to mitigate the distortion in the TC temperature readings as a function of the distance, by also leveraging the accurate distance measures from the radar. Through this method, we obtain worst-case errors of 0.5 °C.

The paper is organized as follows. In Section II, we discuss the related work. Section III introduces some basic concepts about mm-wave radars and thermal imaging systems, while in Section IV the proposed approach is thoroughly presented. Section V contains an in depth evaluation of milliTRACE-IR on a real experimentation setup and concluding remarks are provided in Section VI.

II. Related Work

In the literature, almost no work has focused on a joint approach to social distancing and people's body temperature monitoring which preserves the privacy of the users. Here, we discuss several prior works in related areas, highlighting the differences with respect to the system proposed in this paper.

Social distancing monitoring: Social distancing has been one of the most widely employed countermeasures to contagious diseases outbreaks [1]. Real-time monitoring of the distance between people in workplaces or public buildings is key for risk assessment and to prevent the formation of crowds. Existing approaches use either wireless technology like Bluetooth or WiFi [11], [12], which require the users to carry a mobile device, or camera-based systems [13], which are privacy invasive. Other approaches use the received signal strength indication (RSSI) from cellular communication protocols [1] or wearables [14], although these are often inaccurate, especially when used in crowded places [1]. A lot of effort has been put into designing person detection and tracking algorithms for crowd monitoring and people counting [15] by using fixed surveillance cameras and mobile robots [16]. The main drawbacks of these methods are the intrinsic difficulty in estimating the distance between people from images or videos, along with the fact that the users have to be continuously filmed during their daily lives, which raises privacy concerns.

Concurrently, a large body of work has focused on ultra-wideband (UWB) transmission for people tracking [17], [18], e.g., using mm-wave radars, as these naturally allow measuring distances with decimeter-level accuracy. However, none of these works has tackled the problem of estimating interpersonal distances when people are very close to one another for extended periods of time; this is especially difficult with radio signals, as the separation of the reflections from different subjects becomes challenging.

Passive temperature screening: Infrared thermography is widely adopted for non-contact temperature screening of people in public places [19]. Due to the COVID-19 pandemic, there has been a growing interest in developing screening methods to measure the temperature of multiple subjects simultaneously, without requiring them to collaborate and/or to carry dedicated devices [20]. Approaches that involve the use of RGB cameras, e.g., [21], share the aforementioned privacy-related limitations.

The authors of [10] developed a Bayesian framework to measure the body temperature of multiple users using low-cost passive infrared sensors. The distance from the sensors and the number of subjects is also obtained. However, the working range of this system is very short (around 1.5 m for precise temperature estimation), so it is deemed unapt for monitoring a large indoor area.

Radar-thermal imaging association and fusion: Sensor fusion between radars and RGB cameras has been extensively investigated, see, e.g., [22], [23], while the joint processing of mm-wave radar data and infrared thermal images was marginally treated [24]. In addition, the last paper only deals with the detection of humans using thermal imaging and does not address body temperature screening.

In the present work, we focus on the *data association* between a thermal camera and a mm-wave radar over short periods of time, using the accurate radar distance estimates to refine the temperature reading. This makes it possible to consider scenarios where the thermal camera only covers a small portion of the environment (e.g., the entrance) so as to preserve the subjects' privacy, while a mm-wave radar network can effectively monitor the whole indoor space.

mm-Wave radar person re-identification (Re-Id): Radio-frequency (RF) based person Re-Id is a recent research topic. So far, many works have focused on person identification [25], [26], where the subjects to identify have been previously seen by the system, typically via a preliminary training phase. Re-Id is more challenging, as it addresses the recognition of *unseen* subjects, for which only a few radio samples are collected during system operation. Differently from camera image based Re-Id methods [27], RF approaches need to profile the users across time intervals of a few seconds, to extract robust person specific features [28]. To the best of our knowledge, only two works have proposed solutions to this problems [28], [29]. In both cases, a deep learning method trained on a large set of users is used to extract features from the human gait. At test time, the features obtained from the subjects to be re-identified are compared against those of a set of known individuals using distance-based similarity scores. This approach entirely depends on the feature extraction process, and the classifier does not learn to refine its decisions at *runtime*, as new samples become available. This is a weakness, as the gait features extracted from mm-wave radars are known to be variable, e.g., across different days [30]. Conversely, we propose to combine deep feature extraction with fast classifiers which are continuously trained and refined as new data is collected; this improves the robustness of the identification task.

III. Preliminaries

In this section we summarize the main working principles of the sensing technologies used in this work, namely, *frequency-modulated continuous wave* (FMCW) mm-wave radars and infrared thermal cameras.

A. mm-Wave FMCW Radar

A MIMO FMCW radar allows the joint estimation of the distance, the radial velocity and the angular position of the targets with respect to the radar device [31]. It works by transmitting sequences of *chirp* signals, linearly sweeping a bandwidth B, and analyzing their copies, which are reflected back from the environment. A full chirp sequence, termed *radar frame*, is repeated with period Δ seconds.

1) Distance, velocity and angle estimation: By computing the frequency shift induced by the delay of each reflection, the radar allows obtaining the distance and velocity of the targets with high accuracy. The use of multiple receiving antennas, organized in a *planar array*, allows obtaining the angle-of-arrival (AoA) of the reflections along the azimuth and the elevation dimensions, leveraging the different frequency shifts measured by the different antenna elements. This enables the localization of the targets in the physical space.

2) Radar detection: The raw output of the radar is typically high dimensional for mm-wave devices, due to the high resolution. To sparsify the signal and perform a detection of the main reflecting points, a typical approach is the *constant false alarm rate* (CFAR) algorithm [32], which consists of applying a dynamic threshold on the power spectrum of the output signal. A further processing step is required to remove the reflections from static objects, i.e., the *clutter*. This operation is performed using a *moving target indication* (MTI) high pass filter that removes the reflections with Doppler frequency values close to zero [32].

3) Radar point-clouds: After the detection phase, a human presence in the environment typically generates a large number of detected points. This set of points, usually termed radar *point-cloud*, can be transformed into the 3-dimensional Cartesian space ($x-y-z$) using the distance, azimuth and elevation angles information of the multiple body parts. In addition, the velocity of each point is also retrieved, along with the strength of the corresponding signal reflection.

In the following, we refer to the point-cloud outputted by the radar at frame k as $\mathcal{P}_k$, containing a variable number of reflecting points. Each point, $\boldsymbol{p} \in \mathcal{P}_k$, is described by vector $\boldsymbol{p} = \left[x, y, z, v, P^{\rm RX}\right]^T$, including its coordinates x, y, z, its velocity v and reflected power $P^{\rm RX}$.

B. Infrared Thermal Cameras

Infrared thermal imaging deals with detecting radiation in the long-infrared range of the electromagnetic spectrum ($\sim 8-15$ μm) and producing images of that radiation, called *thermograms*. According to the *Planck's Law*, infrared radiation is emitted by all objects with temperature $T > 0$ K [33]. Since the radiation energy emitted by an object is positively correlated to its temperature, from the analysis of the received radiation it is possible to measure the object's temperature.

A thermographic camera, or *thermal camera*, is a device that is capable of creating images of the detected infrared radiation. The operating principle is quite similar to that of a standard camera, and the same relations described by the so-called *pinhole camera model* hold [34]. Within this approximation, the coordinates of a point $\boldsymbol{a} = [a_x, a_y, a_z]^T$ in the three-dimensional space are projected onto the image plane of an ideal pinhole camera through a very small aperture. Mathematically, this operation is described as $\boldsymbol{a}^{\rm proj} = \boldsymbol{\Psi}\boldsymbol{a}$, where $\boldsymbol{a}^{\rm proj}$ is the projected point and $\boldsymbol{\Psi}$ is the intrinsic matrix of the camera that contains information about its focal lengths, pixel dimensions and position of the image plane. However, when dealing with a real thermal camera, this approximation may be insufficient and the *radial* and/or *tangential* distortions introduced by the use of a lens and by inaccuracies in the manufacturing process may additionally have to be accounted for. On the image plane, an array of infrared detectors is responsible for measuring the received radiation, which is sampled and quantized to produce a digital information. The pixels of the final image that is returned by

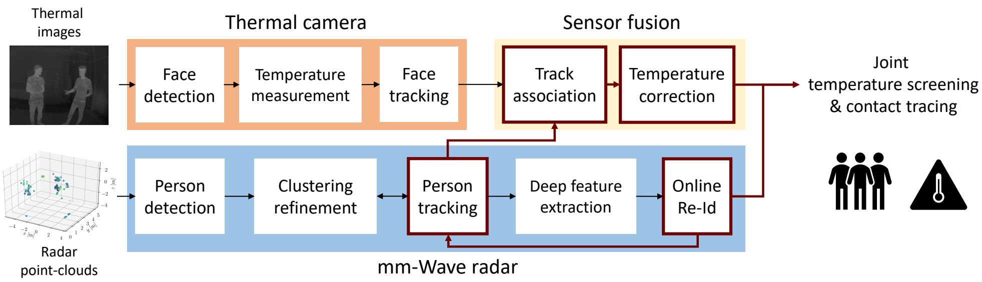


Fig. 1: milliTRACE-IR signal processing workflow.

a thermal camera contain information about the temperature of the corresponding body/object part, encoded into the pixel intensity.

IV. Proposed Approach

We consider the problem of monitoring an indoor environment covered by multiple mm-wave radar sensors, which span over different rooms and corridors. A few infrared thermal cameras are placed at strategic locations to perform accurate temperature screening of the people in the indoor space without compromising their privacy, e.g., at the building's entrance.

From a high-level perspective, milliTRACE-IR performs the following operations.

(1) **Person detection and temperature measurement:** When people enter the monitored indoor space, our system concurrently performs face detection from the infrared images captured by the thermal camera and person detection using the mm-wave radar point clouds.

1) From the thermal camera (TC) images, a face detector is used to obtain bounding boxes enclosing the faces of the detected subjects, (Section IV-B). A measure of their body temperature is obtained from the intensity of the thermal image pixels in the bounding box, see Section IV-C. While our approach is independent of the specific face detector architecture used, in the implementation we used YOLOv3 [8].
2) Concurrently, radar signal processing is used to detect and group the point-clouds from different subjects and estimate their positions (Section IV-D). A novel clustering algorithm based on DBSCAN and Gaussian Mixture models is put forward to separate the contributions of closeby subjects (Section IV-E).

(2) **Radar-TC person tracking:** Kalman filtering (KF) is independently applied to the TC images and to the radar point-clouds to respectively track the subjects' movements within the thermal images and in the indoor Cartesian space. We modify standard KF-based tracking in the thermal image plane to achieve a coarse estimation of the distance of the subjects, based on the dimension of their face bounding box Section IV-B. In this phase, each subject track is associated with a unique numerical identifier.

(3) **Radar-TC track association:** As a subject exits the FoV of the TC, his/her body temperature is associated with the corresponding trajectory from the mm-wave radar, by performing a track-to-track association between TC tracks and radar tracks. This association algorithm is based on the subjects' distances from the TC, and on the radar estimated positions of the subjects, projected onto the thermal image plane (Section IV-F). After the association, the temperature measurement is corrected accounting for the distance of each person from the TC, using the more precise distance estimates provided by the radar, Section IV-C.

(4) **Radar-based person re-identification:** During the radar tracking process, the point-cloud sequences generated by each subject are collected and fed to a deep neural network that performs gait feature extraction (Section IV-G). The resulting gait features are organized into a labeled training set, where labels are obtained from the track identifiers. When a subject exits the FoV of a radar and enters that of another radar placed in a different room or corridor, a weighted extreme learning machine (WELM) based classifier [35] is trained on-the-fly and used to re-identify the subject at runtime (Section IV-I). This robust and lightweight person Re-Id process, based on the gait features extracted from the radar point-clouds, enables contact tracing across large indoor environments.

A. Notation

The system operates at discrete time-steps, $k = 1, 2, \dots,$ each with fixed duration of Δ seconds, also referred to as *frame* in the following. Boldface, capital letters refer to matrices, e.g., $\boldsymbol{X}$, with elements X_{ij}, whereas boldface lowercase letters refer to vectors, e.g., $\boldsymbol{x}$. $\boldsymbol{X}^{-1}$ denotes the inverse of matrix $\boldsymbol{X}$, and $\boldsymbol{x}^T$ denotes the transpose of vector $\boldsymbol{x}$. $\boldsymbol{x}_k$ refers to vector $\boldsymbol{x}$ at time k, x_j refers to element j of $\boldsymbol{x}$ and $(\boldsymbol{x}_k)_j$ is element j of $\boldsymbol{x}_k$. $\mathcal{N}(\mu, \sigma^2)$ indicates a Gaussian random variable with mean μ and variance σ^2. We use the notation $||\boldsymbol{x}||_2$ to indicate the Euclidean norm of vector $\boldsymbol{x}$, while $||\boldsymbol{x}||_{\boldsymbol{\Gamma}}= \sqrt{\boldsymbol{x}^T\boldsymbol{\Gamma}\boldsymbol{x}}$ denotes the norm induced by matrix $\boldsymbol{\Gamma}$. We denote the diagonal matrix with elements $x_1, x_2, \dots, x_n$ by $\mathrm{diag}\,[x_1, x_2, \dots, x_n]$. $|\mathcal{X}|$ indicates the cardinality of set $\mathcal{X}$.

B. Thermal Camera: Face Detection and Tracking

The detection of the subjects in the thermal camera images is performed by means of a face detector that computes rectangular bounding boxes delimiting the faces of the people within the FoV. The bounding boxes are used to track the positions of the subjects in the subsequent instants and to identify a region of interest from which the temperature of the targets is obtained. Our approach is independent of the particular face detector used, provided that it outputs bounding boxes enclosing the faces of the subjects. In our implementation, we used YOLOv3 [8] due to its excellent performance in terms of accuracy and speed.

To track the faces of the subjects in the image plane, we use an extended Kalman filter (EKF) [9]. We define the *state* vector of a target subject at time k, as $\boldsymbol{x}_k = [x^c_k, y^c_k, \dot{x}^c_k, \dot{y}^c_k, h_k, d_k, \dot{d}_k]^T$, where x^c_k, y^c_k are the true coordinates of the center of his/her face in the thermal image, $\dot{x}^c_k, \dot{y}^c_k$ its velocities along the vertical and horizontal directions, h_k is the true height of the bounding box enclosing the subject's face, d_k, the distance of the target from the camera in the physical space, and $\dot{d}_k$ its time derivative (rate of variation).

The observation vector that we obtain from the YOLOv3 face detector, denoted by $\boldsymbol{z}_k = [\tilde{x}^c_k, \tilde{y}^c_k, \tilde{h}_k]^T$, contains noisy measurements of the face position and height (represented by the height of the bounding box), which are distinguished from their true values by the superscript "$\tilde{}$". We denote the observation noise by vector $\boldsymbol{r}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$, with $\boldsymbol{R} = \text{diag}\left(\sigma^2_{\tilde{x}^c}, \sigma^2_{\tilde{y}^c}, \sigma^2_{\tilde{h}}\right)$, with diagonal elements representing the (constant) observation noise variances of $\tilde{x}^c_k$, $\tilde{y}^c_k$ and $\tilde{h}_k$, respectively. We used $\sigma^2_{\tilde{x}^c} = \sigma^2_{\tilde{y}^c} = 0.01$ and $\sigma^2_{\tilde{h}} = 20$.

The EKF state transition model is defined as $\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{u}_k)$, where $f(\cdot)$ is the transition function, connecting the system state at time k, $\boldsymbol{x}_k$, to that at time $k+1$, $\boldsymbol{x}_{k+1}$, and vector $\boldsymbol{u}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q})$ represents the process noise. In our model, the process noise includes 4 independent components, representing two random accelerations of the bounding-box center coordinates, u^x_k, u^y_k, a random noise term for the bounding-box dimension, u^h_k, and a random acceleration for the subject's distance, u^d_k. Therefore, we can write $\boldsymbol{u}_k = \left[u^x_k, u^y_k, u^h_k, u^d_k\right]^T$ with covariance matrix $\boldsymbol{Q} = \text{diag}\left[\sigma^2_x, \sigma^2_y, \sigma^2_h, \sigma^2_d\right]$. In our implementation, we used $\sigma^2_x = \sigma^2_y = \sigma^2_d = 5$ and $\sigma_h = 5.148$ (see Section IV-B1).

Assuming that the target moves according to a *constant velocity* (CV) model, from the state definition we get

$$f(\boldsymbol{x}_k, \boldsymbol{u}_k) = \begin{bmatrix} x_k + \Delta\dot{x}_k + u^x_k\Delta^2/2 \\ y_k + \Delta\dot{y}_k + u^y_k\Delta^2/2 \\ \dot{x}_k + u^x_k\Delta \\ \dot{y}_k + u^y_k\Delta \\ g\left(d_k + \Delta\dot{d}_k + u^d_k\Delta^2/2\right) + u^h_k \\ d_k + \Delta\dot{d}_k + u^d_k\Delta^2/2 \\ \dot{d}_k + u^d_k\Delta \end{bmatrix}, \quad (1)$$

where the only non-linear term is function $g(\cdot)$, which relates the subject's distance extracted by the thermal camera to the height h_k of the bounding-box enclosing his/her face. Our approach consists in *(i)* obtaining an estimate for $g(\cdot)$ in an *offline* fashion using training data, and *(ii)* using such estimate in the EKF model. These two steps are detailed next.

1) Estimation of function $g(\cdot)$: Function $g(\cdot)$ maps the distance of the target from the thermal camera d_k, at time k, onto the corresponding height of the bounding box, h_k, as follows,

$$h_k = g(d_k) + u^h_k. \quad (2)$$

Using N_t training samples $\{h_i, d_i\}_{i=1}^{N_t}$ containing the true distances of the target, d_i, and the measured bounding box height, h_i, we obtain $g(\cdot)$ solving an *offline* non-linear least-squares (LS) problem of the form

$$\arg\min_g \sum_{i=1}^{N_t} \left(h_i - g(d_i)\right)^2. \quad (3)$$

From the equations of the pinhole camera model [34], we restrict $g(\cdot)$ to the family of hyperbolic functions with shape $g(d_i) = b_0/(d_i + b_1) + b_2$, reducing the problem to that of estimating the parameters b_0, b_1, and b_2, i.e.,

$$\arg\min_{b_0, b_1, b_2} \sum_{i=1}^{N_t} \left(h_i - \frac{b_0}{d_i + b_1} + b_2\right)^2. \quad (4)$$

This optimization problem is here solved using the Levenberg-Marquardt algorithm [36] for non-linear LS fitting: with our setup, we obtained $b_0 = 162.04, b_1 = 0.61, b_2 = -14.79$.

Note that the process noise acts on the bounding-box dimension in two ways, inside the function $g(\cdot)$, modeling the uncertainty in the subject's distance due to the random acceleration, and through the additive term u^h_k, modeling the imperfect estimation of $g(\cdot)$ itself. The variance of u^h_k can be estimated from the residuals, after fitting the training measurements with function $g(\cdot)$.

2) Using $g(\cdot)$ in the EKF: Due to the non-linear dependence of the state $\boldsymbol{x}_k$ on the process noise $\boldsymbol{u}_k$, in the EKF operations we use the following transformed process noise covariance matrix [37]

$$\boldsymbol{Q}'_k = \boldsymbol{L}_k \boldsymbol{Q} \boldsymbol{L}_k^T, \quad \text{with } \boldsymbol{L}_k = \left.\frac{\partial f(\boldsymbol{x}_k, \boldsymbol{u}_k)}{\partial \boldsymbol{u}_k}\right|_{\hat{\boldsymbol{x}}_{k|k}}, \quad (5)$$

where matrix $\boldsymbol{L}_k$ is the Jacobian of function $f(\cdot)$ with respect to the process noise vector, evaluated for the current state estimate. Using the above system model, we recursively obtain the system state estimate at time k, $\hat{\boldsymbol{x}}_k$, along with the corresponding error covariance matrix, $\boldsymbol{P}_k$. By definition of the EKF state, this allows us to get a coarse estimate of the distance of the subjects from the TC, which is exploited in the radar-TC data association step, see Section IV-F.

C. Thermal Camera: Subject Temperature Estimation

The body temperature is obtained from the thermal camera readings in the bounding-boxes contained in $\hat{\boldsymbol{x}}_k$, for each subject, and for all the time steps in which he/she is tracked by the EKF. At any given time k, a single (noisy) temperature measurement, $\tilde{T}_k$, is extracted by taking the maximum value across all the pixels in the current bounding box. Denoting by B_k the 2-D region of the image enclosed by the bounding box, and by B_{ki} the intensity of its pixel i, it holds $\tilde{T}_k = \max_i B_{ki}$. In line with [10], the direct reading $\tilde{T}_k$ is subject to a scaling factor, $\alpha(d_k)$, with respect to the true subject's temperature T, where $\alpha(d_k)$ depends on the distance from the TC, i.e.,

$$T = \alpha(d_k)\tilde{T}_k. \quad (6)$$

For an accurate temperature screening, the scaling factor $\alpha(d_k)$ is estimated from the training data, considering a linear model of the form

$$\alpha(d_k) = a_0 + a_1 d_k. \quad (7)$$

Using $N_t^{'}$ training measurements $\{\tilde{T}_i, d_i, T\}_{i=1}^{N_t^{'}}$, the fitting coefficients a_0, a_1 are obtained by solving the following LS problem

$$\arg\min_{a_0,a_1} \sum_{i=1}^{N_t^{'}} \left(T - \alpha(d_i)\tilde{T}_i\right)^2 . \tag{8}$$

Solving this optimization problem, we found $a_0 = 1.116$, $a_1 = 0.013$. At system operation time, denoting by M the number of time-steps for which the subject is correctly tracked by the EKF, his/her true temperature at time k is finally estimated as

$$\hat{T}_k = \frac{1}{M} \sum_{j=k-M+1}^{k} \alpha(\hat{d}_j)\tilde{T}_j, \tag{9}$$

where $\alpha(\cdot)$ is defined in Eq. (7), using the parameters obtained from Eq. (8), while $\hat{d}_j$ is an estimate of the distance obtained by the system at time-step j. milliTRACE-IR uses the distance computed by the mm-wave radar device, after the data association step has been performed (see Section IV-F), as it is significantly more accurate than that obtained from the TC.

D. mm-Wave Radar: People Detection and Tracking

The common approach to people tracking from mm-wave radar point-clouds [18], [26], [30] includes
(i) **detection**: using density-based clustering to separate the points generated by the subjects from clutter and noise;
(ii) **tracking**: applying Kalman filtering techniques [7] on each cluster centroid to track the movement trajectory of each subject in space.

Detection is typically performed using DBSCAN [6], an unsupervised density-based clustering algorithm that takes two input parameters, ε and $m_{\rm pts}$, respectively representing a radius around each point and the minimum number of other points that must be inside such radius to satisfy a certain density condition. Given the description of the radar measurements from Section III-A, the coordinates of the points in the horizontal plane $(x-y)$ are used as input to DBSCAN, which outputs a list of detected clusters and a set of points which are classified as *noise*. Typically, the centroid of each cluster is used as an *observation* of the subject's position, feeding a subsequent KF tracking algorithm [18]. DBSCAN has proven to be robust and accurate as long as the subjects do not come too close to one another [18], [26], [38], see also Fig. 2a. When this occurs (Fig. 2b), the algorithm often fails to distinguish between adjacent subjects, merging their contributions into a single cluster [39].

In the KF tracker, the *state* of each subject at time k is defined as $\boldsymbol{s}_k = [x_k, y_k, \dot{x}_k, \dot{y}_k]^T$, containing the $x-y$ subject's coordinates and the corresponding velocities. The state evolution is assumed to obey $\boldsymbol{s}_k = \boldsymbol{A}\boldsymbol{s}_{k-1}$, where the transition matrix $\boldsymbol{A}$ represents a constant-velocity (CV) model [38]. The KF computes an estimate of the state for a target subject at time k, denoted by $\hat{\boldsymbol{s}}_k$, by sequentially updating the predictions from the CV model with the new observations. The association between the new observations (time k) and the previous states (time $k-1$) exploits the nearest-neighbors joint probabilistic data association algorithm (NN-JPDA) [38], [40]. In this work, we also use the just described DBSCAN and KF based signal processing pipeline, by developing a novel clustering procedure to better resolve the point-clouds of subjects that are close to one another. The solution that we designed to enhance the tracking accuracy in such cases is a major contribution of our work and is detailed next.

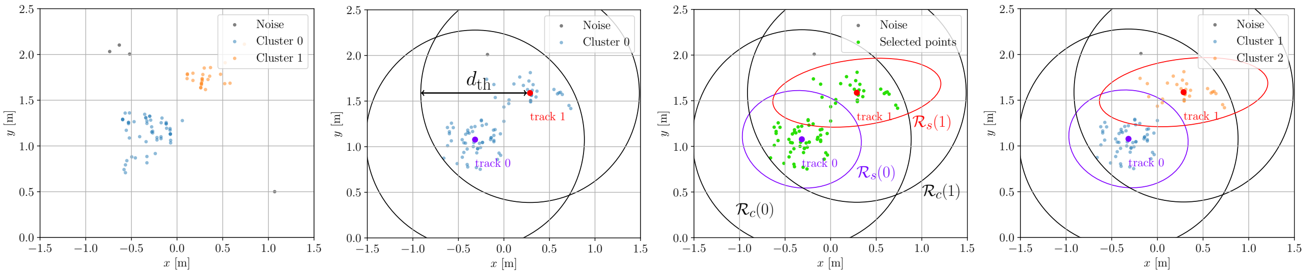
E. mm-Wave Radar: Highly Accurate Clustering

As a possible solution to DBSCAN drawbacks, one may adjust the parameters ε and $m_{\rm pts}$ so as to correctly resolve the clustering ambiguity, even for closely spaced targets. However, ε and $m_{\rm pts}$ interact in a complex and often unpredictable way, making the design of such adaptation rule difficult.

In milliTRACE-IR, we adopt a different approach, which combines *(i)* the standard DBSCAN algorithm with fixed ε and $m_{\rm pts}$, *(ii)* the spatial locations of the subjects, available from the tracking procedure, and *(iii)* the Gaussian mixture (GM) clustering algorithm [5]. Our algorithm, reported in Alg. 1 and exemplified in Fig. 2, proceeds as follows. At first, the DBSCAN algorithm is applied to obtain an estimate of the clusters and a reasonable separation between the noise points and those belonging to actual subjects, using $\varepsilon = 0.4$ m and $m_{\rm pts} = 10$. DBSCAN outputs a cluster label for each point $\boldsymbol{p} \in \mathcal{P}_k$, denoted by $\ell_{\boldsymbol{p}}$. Clusters are denoted by $\mathcal{C}_n$, and their centroids by $\bar{\boldsymbol{c}}_n$, with $n = 1, \dots, n_k$.

The next step is to identify which of the tracked subjects get closer than a critical distance $d_{\rm th}$ from one another. We expect that the clusters provided by standard DBSCAN for these subjects will be incorrect, as the point-cloud data from these would be merged into a single cluster. To pinpoint these subjects, we leverage their KF state, which corresponds to a filtered representation of their trajectories. Let us consider track t at time k, its coordinates are predicted as $\hat{\boldsymbol{s}}_k^t = \boldsymbol{A}\hat{\boldsymbol{s}}_{k-1}^t$ (see line $2-3$ in Alg. 1). For any two subjects with associated tracks t and t', we check whether $||\hat{\boldsymbol{s}}_k^t - \hat{\boldsymbol{s}}_k^{t'}||_2 < d_{\rm th}$. If this occurs, as shown in the example of Fig. 2b for tracks $t = 0$ and $t' = 1$, we say that t and t' are *nearby subjects*. Hence, we define $\mathcal{G}$ as the set of subjects that are mutually within a radius of $d_{\rm th}$ from one another. A group $\mathcal{G}$ can be constructed starting from any subject and recursively adding all the subjects who are closer than $d_{\rm th}$ from any of the set members. If a subject has no other subjects within distance $d_{\rm th}$, it will be the only member of his group. Collecting all the disjoint groups, constructed from the maintained tracks at time k, we obtain set $\boldsymbol{\mathcal{G}}_k(d_{\rm th})$ containing all the nearby subjects groups. Once the nearby groups are identified, we resolve the ambiguities inside each group $\mathcal{G}$ containing more than one member, recomputing the clustering labels as follows. Consider a single group $\mathcal{G}$. To delimit the region where the clustering has to be refined, we define the following additional regions. With $\boldsymbol{\Sigma}_n^t$ we refer to the sample covariance matrix of the *last* cluster associated with track t, containing information about the shape of the subject's cluster. We consider the regions of the plane containing the points that are within a radius of $d_{\rm th}$ from $\hat{\boldsymbol{s}}_k^t$, as

$$\mathcal{R}_c(t) = \left\{\boldsymbol{x} \in \mathbb{R}^2 \text{ s.t. } \left|\left|\boldsymbol{x} - \hat{\boldsymbol{s}}_k^t\right|\right|_2 < d_{\rm th}\right\}, \tag{10}$$



(a) (b) (c) (d)

Fig. 2: Illustration of the proposed clustering method. In (a) the point-clouds belonging to 2 subjects are well separated and DBSCAN outputs the correct clustering. In the next time-step, (b), DBSCAN fails and merges the two clusters into one. Our method selects the points to re-cluster using the tracks positions together with Eq. (10) and Eq. (11), as shown in (c), and outputs the correct result using GM on the selected points with $n_{\mathcal{G}} = 2$, see (d).

Algorithm 1 Clustering refinement method.

```
Input: States of the targets at time $k-1$, observed point-cloud at time $k$, $\mathcal{P}_k$.
Output: Labels $\ell_{\boldsymbol{p}}$, $\forall \boldsymbol{p} \in \mathcal{P}_k$.
 1: $\{\ell_{\boldsymbol{p}}\}_{\boldsymbol{p}\in\mathcal{P}_k}, \{\mathcal{C}_n\}_{n=1}^{n_k} \leftarrow \text{DBSCAN}(\varepsilon, m_{\text{pts}}, \mathcal{P}_k)$
 2: $\hat{\boldsymbol{s}}_k^t \leftarrow \boldsymbol{A}\hat{\boldsymbol{s}}_{k-1}^t$ all maintained tracks $t$
 3: Find groups of nearby subjects $\boldsymbol{\mathcal{G}}_k(d_{\text{th}})$
 4: for each $\mathcal{G} \in \boldsymbol{\mathcal{G}}_k(d_{\text{th}})$ do
 5:     $n_{\mathcal{G}} \leftarrow |\mathcal{G}|$
 6:     if $n_{\mathcal{G}} > 1$ then
 7:         $\mathcal{R}(\mathcal{G}) \leftarrow \bigcup_{t\in\mathcal{G}}(\mathcal{R}_c(t) \cap \mathcal{R}_s(t))$
 8:         $\mathcal{S} \leftarrow \{\boldsymbol{p} \in \mathcal{C}_n$ such that $\bar{\boldsymbol{c}}_n \in \mathcal{R}(\mathcal{G})\}$
 9:         discard $\ell_{\boldsymbol{p}}, \forall \boldsymbol{p} \in \mathcal{S}$
10:         $\{\ell_{\boldsymbol{p}}\}_{\boldsymbol{p}\in\mathcal{S}}, \{\pi_q\}_{q=1}^{n_{\mathcal{G}}} \leftarrow \text{GM}(n_{\mathcal{G}}, \mathcal{S})$
11:         discard cluster $q$ if $\pi_q < \pi_{\text{thr}}$
12:     end if
13: end for
```

and the regions of points with a squared Mahalanobis distance smaller than γ,

$$\mathcal{R}_s(t) = \left\{ \boldsymbol{x} \in \mathbb{R}^2 \text{ s.t. } \left|\left|\boldsymbol{x} - \hat{\boldsymbol{s}}_k^t\right|\right|^2_{(\boldsymbol{\Sigma}_n^t)^{-1}} < \gamma \right\}. \tag{11}$$

For our results, we selected $d_{\text{th}} = 1.2$ m and $\gamma = 9.21$.[1] Then, the labels assigned by DBSCAN to all the points belonging to a cluster whose centroid falls inside region $\mathcal{R}(\mathcal{G}) = \cup_{t\in\mathcal{G}}(\mathcal{R}_c(t) \cap \mathcal{R}_s(t))$, are discarded (lines $7-9$ in Alg. 1).[2] We denote this set of points by $\mathcal{S}$.

Then, the GM algorithm is applied to the points belonging to set $\mathcal{S}$ to refine the clusters within this region, see the green points in Fig. 2c. As GM requires the number of clusters to be specified in advance, we set it equal to the number of subjects in the group, i.e., $n_{\mathcal{G}} = |\mathcal{G}|$. The GM algorithm outputs the labels $\ell_{\boldsymbol{p}}$ for each point $\boldsymbol{p} \in \mathcal{S}$ and the weight of the Gaussian component associated with each GM cluster, $\pi_q \in [0, 1], q = 1, \dots, n_{\mathcal{G}}$, with $\sum_q \pi_q = 1$. We use the new labels to replace the ones previously found by DBSCAN (Fig. 2d), unless the GM clusters have very small weights, i.e., the new clusters having $\pi_q < \pi_{\text{thr}}$ are discarded and treated as noise points. We set $\pi_{\text{thr}} = 0.1/n_{\mathcal{G}}$.

The proposed method effectively solves the problem faced by DBSCAN in resolving subjects close to one another. The cost of this improvement is that an additional GM algorithm has to be applied to a subset of the point-cloud, however, at each time k the number of points in this subset is typically much smaller than that in the full point-cloud $\mathcal{P}_k$.

F. Radar and Thermal Camera Data Association

Upon tracking the subjects in the TC image plane and in the physical space, respectively using the measurements from the TC and from the mm-wave radar sensor, we apply a track-to-track association method to link the movement trajectory of each person to his/her body temperature.

Assume that, at time k, the system has access to N_k^{rad} tracks from the radar sensor and N_k^{tc} tracks from the thermal camera, indicized by i and j, respectively. Our data association strategy consists in *(i)* computing a *cost* for each association $(i \leftrightarrow j)$, and *(ii)* solving the resulting combinatorial cost minimization problem to associate the best matching track pairs. The main challenge in the association of radar and thermal camera tracks is the design of a cost function that grants robustness in the presence of multiple targets, which may enter the monitored area in unpredictable ways. The key point is to gauge the similarity of the tracks by comparing them in terms of common quantities, which can be estimated from both devices.

Assume also that the two sensors are located in the same position and with the same orientation (co-located). In this setup, *(i)* the *distance* between the subjects and the sensors is the same, so its estimate should match for tracks representing the same subjects, and *(ii)* the radar KF states containing the coordinates of the subjects' positions can be projected onto the TC image plane; after this operation, the horizontal component of the radar projections and the horizontal component of the TC bounding boxes position should match for correctly associated tracks. We denote by d_k^i the estimated distance of radar track i, and by d_k^j the estimated distance of TC track j. With x_k^i and x_k^j we indicate the horizontal component of the radar state projection and the horizontal component of the TC bounding box center, respectively.

Note that the radar positions provided by the KF state have only two dimensions, that is, x and y (the first and

[1] This value corresponds to a probability of 99% of falling inside the region, assuming that the points in the cluster are distributed on the plane according to a Gaussian distribution around $\hat{\boldsymbol{s}}_k^t$.

[2] Discarding a label corresponds to setting it equal to that used by DBSCAN to represent noise points.

second components of the state vector). However, we need three-dimensional vectors for their proper projection onto the TC image plane. For this reason, we artificially add a 0-valued z component and we assume to track the subjects' position at height 0. Denoting by $\boldsymbol{a}_k^i = [(\hat{\boldsymbol{s}}_k^i)_1, (\hat{\boldsymbol{s}}_k^i)_2, 0]^T$ the vector identifying subject i's position at time k, d_k^i is computed using Pythagora's theorem as $d_k^i = \sqrt{(\hat{\boldsymbol{s}}_k^i)_1^2 + (\hat{\boldsymbol{s}}_k^i)_2^2}$. Distance d_k^j is retrieved directly from the tracking state of the TC. x_k^i is computed by projecting the radar coordinates $\boldsymbol{a}_k^i$ onto the TC image plane, as $\boldsymbol{a}_k^{i,\mathrm{proj}} = \boldsymbol{\Psi}\boldsymbol{a}_k^i$ (see Section III-B), applying to it a radial distortion based on the estimated distortion coefficients and retaining only the x-axis component. Projection x_k^j corresponds to the x coordinate of the TC tracked state.

We define the association cost $A(i,j)$ for the tracks pair (i,j) as the sum of two terms, representing how well the tracks match in terms of their estimated distance across time, $A_d(i,j)$, and of the estimated position on the horizontal axis of the projections on the TC image plane, $A_x(i,j)$. Formally, considering K subsequent time steps in which radar track i and TC track j are both available, we define

$$A(i,j) = A_d(i,j) + A_x(i,j) \tag{12}$$

$$= \frac{\rho(K)}{K}\left(\sum_{k=1}^{K}\frac{(d_k^i - d_k^j)^2}{\sigma^2_{d_k^i} + \sigma^2_{d_k^j}} + \sum_{k=1}^{K}\frac{(x_k^i - x_k^j)^2}{\sigma^2_{x_k^i} + \sigma^2_{x_k^j}}\right) \tag{13}$$

where each term in the sums is weighted according to its estimated variance and, recalling that Δ is the sampling interval of the system,

$$\rho(K) = \frac{1}{\ln(K/\Delta)} \tag{14}$$

is a term that favours longer tracks with respect to shorter ones. Costs $A(i,j)$, $i = 1, \ldots, N_k^{\mathrm{rad}}$, $j = 1, \ldots, N_k^{\mathrm{tc}}$, are arranged into a $N_k^{\mathrm{rad}} \times N_k^{\mathrm{tc}}$ cost matrix, and the optimal association of tracks is obtained by minimizing the overall cost, computed through the Hungarian algorithm [41]. The Hungarian algorithm uses the cost matrix as input and solves the problem of pairing each track with only one other track while minimizing the total cost, entailing an overall complexity $O((N_k^{\mathrm{rad}}N_k^{\mathrm{tc}})^3)$.

In a more general setting, the two devices will be deployed at different locations in space. However, knowing their relative position and orientation, it is possible to compute a roto-translation matrix $\boldsymbol{\Phi}$ to geometrically transform the data into a new coordinate system where the TC and the radar sensors are co-located, as described above. To do so, we select the TC position and orientation as a reference, and transform the positions estimated from the radar sensor into the TC coordinate system, see also Section V-A.

G. *Extraction of Feature Vectors from mm-Wave Point-Clouds*

To extract the gait features of the subjects, we adapt the neural network (NN) proposed in [30], which was developed for person identification. The network uses a point-cloud feature extraction block inspired by PointNet [42], and followed by temporal dilated convolutions [43] to capture features related to the movement evolution in time. The proposed NN takes as input a radar point-cloud sequence, denoted by $\boldsymbol{Z}$, and outputs the corresponding feature vector $\boldsymbol{v} = \mathcal{F}(\boldsymbol{Z})$. In Fig. 3, we show the block diagram of the NN. First, we expand the network with respect to [30], using augmented point-cloud features of size 196×1 and 2 temporal convolution blocks containing 3 convolutional layers each, with $(32, 64, 128)$ and $(256, 128, 32)$ filters, respectively, for the two block. Then, we introduce a fully connected layer [44] before the classification output, which produces a vector $\tilde{\boldsymbol{v}}$ of dimension 32. The final feature vector is obtained using L_2-normalization on $\tilde{\boldsymbol{v}}$, i.e., $\boldsymbol{v} = \tilde{\boldsymbol{v}}/||\tilde{\boldsymbol{v}}||_2$.

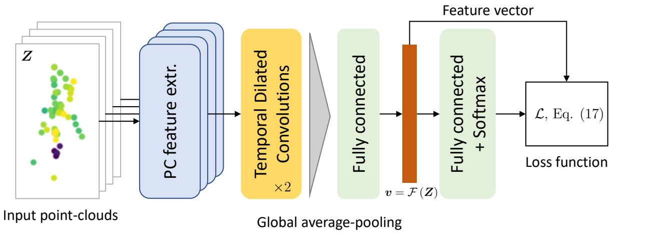


Fig. 3: Block diagram of the NN feature extractor.

1) Training: the NN is trained to produce representative feature vectors, $\boldsymbol{v}$, containing information on the way of walking of the subjects. This requires that the network generalizes well to subjects *not seen* at training time, as the performance of the re-identification mechanism strongly depends on the quality of the extracted features. To this end, we propose to train the NN using a weighted combination of the *cross-entropy loss* [44], which we denote by $\mathcal{L}_{\mathrm{ce}}$, the *center loss* [45], $\mathcal{L}_{\mathrm{cnt}}$, and the *triplet loss* [46], $\mathcal{L}_{\mathrm{tri}}$.

The cross-entropy is the most widely used loss for classification purposes in deep learning, and here it is used to train the network to distinguish among the different subjects [44]. However, just training the NN on a classification problem does not lead to sufficiently discriminative features for the re-identification mechanism. We adopt the center loss to additionally force the feature representations belonging to the same class to be close in the feature space, in terms of Euclidean distance. Specifically, denoting by $\boldsymbol{c}_l$ the centroid of the feature vectors belonging to class l, we have

$$\mathcal{L}_{\mathrm{cen}}(\boldsymbol{v}, l) = ||\boldsymbol{v} - \boldsymbol{c}_l||_2^2, \tag{15}$$

where the centroids are learned as part of the training process via the back-propagation algorithm [45].

The triplet loss is used to push apart the feature representations of inputs belonging to different classes. For this, triplets of input samples are used, containing two vectors from the same class, denoted by $\boldsymbol{v}_a$ and $\boldsymbol{v}_b$, and a vector belonging to a different class, $\boldsymbol{v}_c$. The triplet loss can be computed as

$$\mathcal{L}_{\mathrm{tri}}(\boldsymbol{v}_a, \boldsymbol{v}_b, \boldsymbol{v}_c) = \max\left\{||\boldsymbol{v}_a - \boldsymbol{v}_b||_2^2 - ||\boldsymbol{v}_a - \boldsymbol{v}_c||_2^2 + \mu, 0\right\}, \tag{16}$$

where μ is a margin hyperparameter. Hence, we train the feature extractor with the following total loss function

$$\mathcal{L} = \mathcal{L}_{\mathrm{ce}} + \mathcal{L}_{\mathrm{tri}} + \omega\mathcal{L}_{\mathrm{cen}}, \tag{17}$$

where the parameter $\omega = 0.5$ is used to tune the relative importance of the center loss. In our implementation, we used a training dataset containing mm-wave radar point-clouds from 16 subjects, collected in different indoor environments to increase the generalization capabilities of the network. We used the Adam optimizer [44] with learning rate 10^{-4} for 250 epochs.

2) Feature extraction: at inference time, i.e., during the system operation, the NN is used to compute feature vectors that are representative of the subjects' gait. Specifically, we collect 45 steps (3 seconds) long sequences of radar point-clouds, denoted by $\boldsymbol{Z}$, for each tracked subject. The inner representation $\boldsymbol{v} = \mathcal{F}(\boldsymbol{Z})$, after L_2-normalization, is used as the feature vector.

H. Weighted Extreme Learning Machine (WELM)

The weighted extreme learning machine (WELM) [35] is a particular kind of single-layer feedforward neural network in which the weights of the hidden nodes are chosen randomly, while the parameters of the output layer are computed analytically. Consider an n_{cls}-class classification problem, a training set $\mathcal{V} = \cup_{n=1}^{n_{\text{cls}}} \mathcal{V}_n$ of input *feature vectors* $\boldsymbol{v}$ (see Section IV-G), each with an associated one-hot encoded label $\boldsymbol{y} \in \{0,1\}^{n_{\text{cls}}}$, where $\mathcal{V}_n$ is the set containing the vectors from class $n = 1, \dots, n_{\text{cls}}$. For any $\boldsymbol{v} \in \mathcal{V}$, the WELM computes the matrix of hidden feature vectors $\boldsymbol{H} \in \mathbb{R}^{|\mathcal{V}| \times L}$, with rows $\boldsymbol{h}(\boldsymbol{v})$, where L is the number of WELM hidden units and $\boldsymbol{h}(\cdot)$ is a non-linear activation function. We use $\boldsymbol{h}(\boldsymbol{v}) = \text{ReLU}(\boldsymbol{W}\boldsymbol{v} + \boldsymbol{b})$ where ReLU is the rectified linear unit [44] ($\text{ReLU}(x) = \max(x, 0)$) and $\boldsymbol{W}, \boldsymbol{b}$ are the weights and biases of the ELM hidden layer, respectively. The elements of $\boldsymbol{W}$ and $\boldsymbol{b}$ are here generated from $\mathcal{N}(0, 0.1)$. The WELM learning process amounts to computing, for each class n, the optimal values of an output weight vector $\boldsymbol{\beta}_n$ that minimizes the *weighted* LS L_2-regularized quadratic cost function $||\boldsymbol{H}\boldsymbol{\beta}_n - y_n||^2_{\boldsymbol{\Omega}} + \lambda||\boldsymbol{\beta}_n||^2_2$, where λ is a regularization parameter and $\boldsymbol{\Omega}$ is a diagonal weighting matrix used to boost the importance of those samples belonging to under-represented classes. This compensates for the tendency of the standard ELM to favor over-represented classes at inference time [35]. In our scenario, the individuals move freely in the environment across different rooms, so the number of feature vectors collected from each of them is not only unknown in advance, but highly variable. Hence, the training set usually contains unbalanced classes, and we use

$$\Omega_{i,i} = 1/|\mathcal{V}_{n_i}|, \; i = 1, \dots, |\mathcal{V}|, \tag{18}$$

where we denoted by $n_i = \arg\max_n (\boldsymbol{y}_i)_n$ the class of the i-th vector. Stacking all the $\boldsymbol{\beta}_n$ into a single matrix $\boldsymbol{B} \in \mathbb{R}^{L \times n_{\text{cls}}}$ and the labels into matrix $\boldsymbol{Y} \in \{0,1\}^{|\mathcal{V}| \times n_{\text{cls}}}$, the WELM output weights $\boldsymbol{B}$ can be computed in closed-form using one of the following equivalent expressions

$$\boldsymbol{B} = \boldsymbol{H}^T \left(\lambda \boldsymbol{I} + \boldsymbol{\Omega}\boldsymbol{H}\boldsymbol{H}^T\right)^{-1} \boldsymbol{\Omega}\boldsymbol{Y}, \text{ or} \tag{19}$$

$$\boldsymbol{B} = \left(\lambda \boldsymbol{I} + \boldsymbol{H}^T\boldsymbol{\Omega}\boldsymbol{H}\right)^{-1} \boldsymbol{H}^T\boldsymbol{\Omega}\boldsymbol{Y}. \tag{20}$$

Due to the dimension of the matrix to be inverted, if $|\mathcal{V}|> L$, it is more convenient to use Eq. (20), while if $|\mathcal{V}|\leq L$ Eq. (19) has to be preferred. The output classification for a vector $\boldsymbol{v}$ is then computed as $\arg\max_i \left(\boldsymbol{h}(\boldsymbol{v})^T\boldsymbol{B}\right)_i$, where $\boldsymbol{h}(\boldsymbol{v})^T\boldsymbol{B}$ is a vector of WELM scores for each class.

Algorithm 2 Re-identification mechanism at time k.

```
Input: Training set $\mathcal{V}$, track to be re-identified $t^{\text{id}}$.
Output: Re-id label of $t^{\text{id}}$.
 1: $\boldsymbol{H} \leftarrow \left[\boldsymbol{h}^T(\boldsymbol{v}), \forall \boldsymbol{v} \in \mathcal{V}\right]$
 2: $\boldsymbol{Y} \leftarrow$ labels of $\mathcal{V}$
 3: $\boldsymbol{\Omega} \leftarrow$ Eq. (18)
 4: $\boldsymbol{B} \leftarrow$ Eq. (20) or Eq. (19) depending on $|\mathcal{V}| \lesseqgtr L$
 5: $\boldsymbol{\xi}_0 \leftarrow \boldsymbol{0}$
 6: for $j = 1, \dots, W$ do
 7:     $\boldsymbol{v}_j^{\text{id}} \leftarrow \mathcal{F}(\boldsymbol{Z}_j)$
 8:     $\boldsymbol{\xi}_j \leftarrow \left[\boldsymbol{h}^T(\boldsymbol{v}_j^{\text{id}})\boldsymbol{B} + j\boldsymbol{\xi}_{j-1}\right]/(j+1)$
 9: end for
10: label $\leftarrow \arg\max_i (\boldsymbol{\xi}_W)_i$
```

I. WELM based Person Re-Identification

To enable person re-identification based on the feature vectors $\boldsymbol{v}$ extracted by the NN, we use the WELM multiclass classifier of Section IV-H, which is *trained at runtime* only when the system has to re-identify a previously seen subject. This is done by sequentially collecting feature vectors from all the subjects seen by the system at operation time, and storing them into the training set $\mathcal{V}$.

Note that, although an online sequential version of the ELM training process has been proposed in [47], we choose to train the WELM every time a person has to be re-identified using a batch implementation and including in the training set $\mathcal{V}$ all the subjects seen up to the current time-step k. This is because in the online training procedure of [47] the number of classes has to be *fixed* in advance, while in our setup the number of subjects seen by the system may change in time and the Re-Id procedure must be flexible to the addition of new individuals to the training set $\mathcal{V}$. The WELM training and re-identification phases are detailed in Alg. 2 and explained in the following.

1) Training: the training process is performed at runtime as explained in Section IV-H, using $L = 1,024$ and $\lambda = 0.1$. During the normal system operation, the feature vectors obtained from each track are continuously added to set $\mathcal{V}$, storing the corresponding one-hot encoded vectors containing the subjects' identities into matrix $\boldsymbol{Y}$. To reduce the computational burden, the feature extraction step is executed every 5 time-steps. This is reasonable, as the input sequences to the NN contain 45 time-steps overall and extracting the features at every time-step would lead to highly correlated, and therefore less informative feature vectors, in addition to entailing a higher computation cost. At time-step k, if a subject has to be re-identified, the training procedure of Section IV-H is executed (lines $1-4$): the WELM feature vectors $\boldsymbol{H}$ are computed by applying the activation function $\boldsymbol{h}(\cdot)$ to each training vector and the weight matrix $\boldsymbol{\Omega}$ is obtained from Eq. (18) (lines $1-3$). The WELM output matrix $\boldsymbol{B}$, is computed using Eq. (19) or Eq. (20) depending on $|\mathcal{V}|$ (line 4).

2) Re-identification: the Re-Id procedure is used to recognize subjects that have been seen by the system and associate them with their temperature measurement and their past movement history in the monitored area. Denoting by t^{id} the track to be re-identified, the trained WELM processes the NN features of this user, $\boldsymbol{v}^{\text{id}}$, as follows: $\boldsymbol{h}(\boldsymbol{v}^{\text{id}})^T\boldsymbol{B}$. Due to the

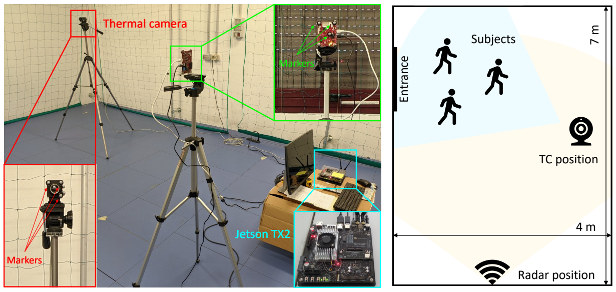


Fig. 4: Experimental setup for the data association.

high variability of human movement, rather than considering a single feature vector, milliTRACE-IR computes the cumulative average WELM scores over a time window of length W, where the average score at time $j = 1, \dots, W$ is referred to as $\boldsymbol{\xi}_j$ (lines $6 - 9$). The identity label corresponds to the index of the largest element of $\boldsymbol{\xi}_W$ (line 10).

V. Experimental Results

milliTRACE-IR has been implemented on an NVIDIA Jetson TX2 edge computing device[3], connected to a Texas Instruments IWR1843BOOST mm-wave radar[4], operating in the $77 - 81$ GHz band, and to a FLIR A65 thermal camera[5], as shown in Fig. 4, operating in real-time at a frame rate of $1/\Delta = 15$ Hz. In this section, we present the experimental results obtained by testing the system in different indoor environments.

A. TC and Radar Tracks Association

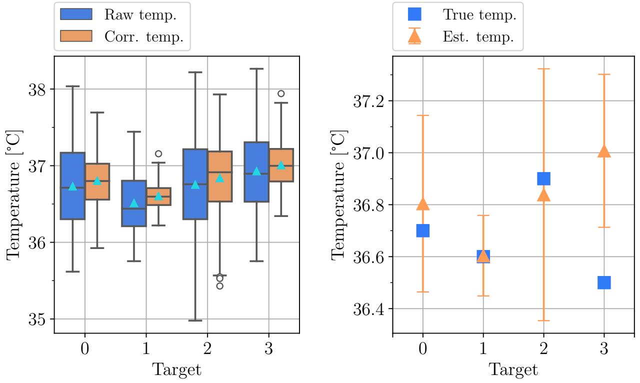
To assess the performance of the radar-TC track association method, we conducted tests in a 7×4 m research laboratory. A motion tracking system including 10 cameras was used to gather ground-truth (GT) data about the locations of the subjects, by placing markers atop their heads. This camera based tracking system provides 3D localization with millimiter-level precision, for all markers, at a rate of 100 Hz. The radar and the TC were placed as shown in Fig. 4. We collected 5 measurement sequences with 2 subjects and 9 sequences with 3 subjects, all freely entering the room. The roto-translation matrix $\boldsymbol{\Phi}$ was estimated using a set of markers applied to the devices, while the TC intrinsic matrix $\boldsymbol{\Psi}$ (see Section IV-F) and the radial distortion coefficients were obtained through the Zhang's method [48], using a sun-heated checkerboard pattern.

We define an *association* as a specific pairing $i \leftrightarrow j$ of a track i from the radar with a track j from the TC, and a *correct association* as an association for which the two tracks correspond to the same subject. Given a set of tracks, we define the set of all the correct associations performed by the algorithm as $\mathcal{A}_{\rm TP}$ (*true positives*), the set of all the associations performed by the algorithm as $\mathcal{A}_{\rm P}$ (*positives*), and the set of all the associations that the algorithm should have performed, based on the GT, as $\mathcal{A}_{\rm R}$ (*relevant*).

To quantify the association performance of our system, we define the *precision*, $\text{Pr} = |\mathcal{A}_{\rm TP}|/|\mathcal{A}_{\rm P}|$, and the *recall*, $\text{Rec} = |\mathcal{A}_{\rm TP}|/|\mathcal{A}_{\rm R}|$. Using these metrics, the proposed track association method is evaluated by assessing the contribution of each cost component in $A(i, j)$ (see Eq. (12)). The results are reported in Tab. 1, where the row labels A_x, A_d, and $A_x + A_d$ indicate the cost function used. In the table, we also show the impact of adding the correction coefficient $\rho(K)$ (see Eq. (14)): for the case "Without $\rho(K)$", we set $\rho(K) = 1$.

	With $\rho(K)$		Without $\rho(K)$	
	Pr [%]	Rec [%]	Pr [%]	Rec [%]
$A_x + A_d$	**97.3**	**97.3**	91.9	91.9
A_x only	91.9	89.2	89.7	94.6
A_d only	92.1	94.6	86.8	89.2

TABLE 1: Impact of the components of the cost function. Row labels A_x, A_d, and $A_x + A_d$ indicate that we used, respectively, only costs A_x, A_d or the sum of the two, in the evaluation. Label "With $\rho(K)$" indicate that we used the corrective term, while label "Without $\rho(K)$" means that we used $\rho(K) = 1$.



(a) Comparison between with (*Corr. temp.*) and without (*Raw temp.*) distance-based correction. The triangles show the mean values.

(b) Comparison between the estimated temperatures and the true temperatures. The error bars represent the standard deviations.

Fig. 5: Results of the temperature screening.

As shown, our track association method reliably associates the radar and TC tracks, reaching precision and recall both higher than 97%. The joint use of A_x, A_d and $\rho(K)$ leads to improvements of up to 11% and 8% for the precision and recall metrics, respectively.

B. Temperature Screening

Remarkably, the proposed temperature screening method does not require people to stand in front of the TC sensor, but estimates their temperature as they move within the FoV of the TC sensor. However, in order for the method to return accurate temperature measurements, the subject' frontal face should be captured by the TC for a minimum time duration. For this reason, we suggest that the TC is placed near a point of passage, e.g., in proximity of an entrance. The performance of the temperature screening method was tested on sequences where this condition is met, that is, where the frontal face was seen from the TC for a reasonable amount of time. In particular, we measured $4 - 7$ sequences of ~ 10 s each from 4 different individuals moving within 3.5 m from the TC. Each subject was tested at a different time of the day, to gauge the effects of the changing (thermal) environmental

[3] https://developer.nvidia.com/embedded/jetson-tx2
[4] https://www.ti.com/tool/IWR1843BOOST
[5] https://www.flir.it/products/a65/

	Mean [°C]	± std [°C]	True temp. [°C]	Error [°C]
Target 0	36.8	0.340	36.7	0.104
Target 1	36.6	0.155	36.6	0.004
Target 2	36.8	**0.485**	36.9	−0.062
Target 3	37.0	0.294	36.5	**0.507**

TABLE 2: Results of the temperature estimation and comparison with respect to the true values for the 4 targets. The worst cases are highlighted.

conditions, and of a possible concept drift (e.g., heating) of the TC after a long period of operation. Furthermore, as explained in Section IV-C, a linear function $\alpha(\cdot)$ was fit to compensate for the influence of the distance on the measures.

To evaluate the benefit brought by the correction based on the targets' distance, in Fig. 5a, we compare the results obtained with (*Corr. temp.*) and without (*Raw temp.*) the correction. Since the TC is intrinsically subject to a bias, to facilitate the comparison of the measures, in the *Raw temp.* case we only correct for this bias, assuming a constant target distance of 2 m and multiplying each measured temperature by $\alpha(2) = a_0 + 2a_1$. The full method (*Corr. temp.*), instead, uses the rescaled average estimate, as per Eq. (9). The box-plot shows that the range of the corrected temperatures is significantly reduced (for these experiments, the true temperature is constant), demonstrating the efficacy of the proposed correction plus averaging approach.

In Fig. 5b, we compare our temperature estimates and the true temperatures measured by means of a contact thermometer. The numerical results are reported in Tab. 2, where the worst cases are reported in bold fonts. Mean temperatures are estimated with a maximum standard deviation smaller than 0.5 °C, and the maximum absolute error with respect to the true temperature is about 0.5 °C.

However, we noticed that the environmental conditions and the heating of the thermal camera do affect the measures in an unpredictable way. Specifically, the *slope* of the fitting function does not change significantly, but a *bias* is introduced (see Target 3 in Fig. 5b) that should be corrected using an external reference, such as a reference piece of material instrumented with a contact thermometer, in the field of view of the TC. Another possible solution is to monitor the people's temperature statistics and to detect deviations from that distribution, so that the system would continuously and automatically adapt to the new and different operating conditions. For instance, if at time k the people mean temperature and its standard deviation are, respectively, μ and σ, it is possible to raise an alarm for a subject whose estimated temperature is $\hat{T} > \mu + c \cdot \sigma$, for a user-defined parameter c.

C. Positioning and Social Distance Monitoring

To evaluate the performance of the radar tracking system in estimating the position of the targets and the inter-subject's distance, we conducted tests in the 7×4 m research laboratory described in Section V-A. We collected 7 sequences of duration $10 - 15$ s, each with 3 subjects moving freely in the room, along with their GT locations obtained from the motion tracking system. We computed the root mean squared error (RMSE) between the mm-wave radar estimated locations and the GT. Moreover, we measured the inter-subject distances considering all the possible combinations of the three subjects, leading to a total of 21 inter-subject distances across all the recorded sequences.

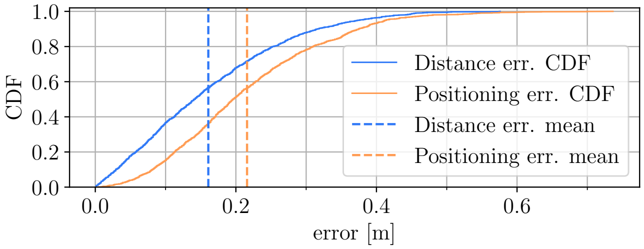


Fig. 6: CDF of the absolute error between the true (ground truth) and the estimated subject's position / inter-subject's distance, as measured by the radar tracking system. The dashed lines denote the mean error.

	Mean [m]	± std [m]	Frames	Time [s]
Position RMSE	0.216	0.115	1448	97
Subj. distance RMSE	0.161	0.112	1153	77

TABLE 3: RMSE of the subject's position and of the inter-subject's distance estimated by the radar sensor, computed against the GT.

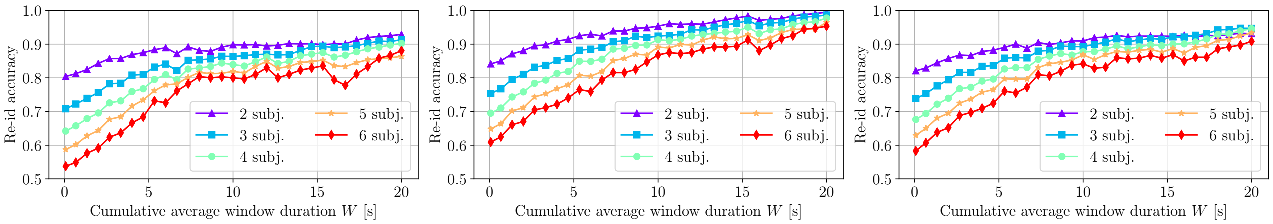
	Ours		**DBSCAN**	
	r_{cl} [%]	corr. tracked	r_{cl} [%]	corr. tracked
2 sub. parallel	90.7	✓	46.5	×
2 sub. crossing	87.9	✓	59.6	×
2 sub. close	89.9	✓	69.7	✓
3 sub. parallel	92.3	✓	65.3	✓
3 sub. crossing	83.7	✓	73.5	×

TABLE 4: Ratio r_{cl} between the number of frames in which the different subjects are correctly separated and the total number of frames, using our method and DBSCAN. With the symbols "✓" and "×" we denote success and failure of the tracking step, respectively.

The cumulative distribution functions (CDF) of the absolute error between the ground truth and the estimated subject's position/inter-subject distance, as measured by the radar tracking system, is shown in Fig. 6, along with the corresponding mean values. The numerical results are provided in Tab. 3. The radar system achieves an absolute *positioning* error within 0.3 m in 80% of the cases. For the inter-subject *distance*, the error remains within 0.25 m in 80% of the cases.

D. Effectiveness of the Improved Clustering Technique

To evaluate the improvement brought by the proposed clustering method over the standard DBSCAN, both algorithms were tested on specific measurement sequences with subjects moving within 1 m from one another. To quantify the clustering performance, we introduce the correct clustering ratio, r_{cl}, representing the fraction of frames in which the clusters belonging to the different subjects are correctly separated. The results of this evaluation are summarized in Tab. 4. We used sequences with 2 and 3 individuals *(i)* walking along parallel paths with the same velocity and at a distance between 0.5 m and 0.8 m (*parallel*), *(ii)* walking along crossing paths, with subjects coming as close as 0.2 m from one another



(a) 1 min. training data.

(b) 3 min. training data.

(c) Imbalanced training data.

Fig. 7: Re-identification accuracy results. In (a) and (b) we used 1 and 3 minutes of training data per subject, respectively. In (c), we used 1 minute of training data for a randomly selected subset containing half of the subjects and 4 minutes for the remaining half.

(*crossing*) and *(iii)* staying still and moving arms at an inter-subject distance of approximately 0.8 m (*close*). The proposed clustering algorithm led to a large improvement (up to 44 %) in terms of $r_{\rm cl}$ metric with respect to DBSCAN. In addition, for 3 of the 5 test sequences, DBSCAN led to failures in the tracking process, either merging the tracks of different subjects, or failing to detect some of them, while our new method correctly tracked all the subjects in all cases.

E. Person Re-Identification

The proposed WELM based Re-Id algorithm was evaluated on a set of mm-wave radar measurements from 6 individuals who were *not* included among the 16 subjects used to train the feature extraction NN. The tests were conducted in a 12×3 m research lab, with furniture that made the evaluation rather challenging. We collected 4 minutes ($3,600$ radar frames) of training data and slightly over 1 minute ($1,000$ frames) of test data per subject, where the individuals walked freely in the room. The radar position was changed for each test to gauge the impact of varying the radar point-of-view.

1) Re-Id accuracy: The Re-Id accuracy as a function of W (see Alg. 2) is shown in Fig. 7a and Fig. 7b. The curves of these plots are obtained averaging the results of 20 different WELM initializations, and all the possible combinations of the considered number of subjects (from 2 to 6) over the 6 total individuals. As expected, the Re-Id performance increases with an increasing inference time (larger W) and with the length of the training sequences: the accuracy gain is about 10% by going from 1-minute (Fig. 7a) to 3 minutes (Fig. 7b) long training sequences. Also, milliTRACE-IR reaches high Re-Id accuracy using $W \geq 15$ s and the detrimental effect of increasing number of subjects to be classified is greatly reduced using larger values of W, as accumulating the WELM scores over longer time windows increases the robustness of the WELM decision. Overall, the accuracy of the proposed method is higher than 95% in all cases, only using 3 minutes of training data per subject and $W = 20$ s, which are reasonable in practice. The worst-case (3 minutes of training data for 6 subjects) WELM training time, on the ARM Cortex-A57 processor of the Jetson TX2 device, took 2.98 ± 0.015 s.

2) Impact of imbalanced training data: As shown in Fig. 7c, the effect of imbalanced training data is successfully mitigated by the sample weighting strategy of Eq. (18). In this evaluation, we trained the WELM with 1 minute of training data for a randomly selected subset containing half of the subjects and 4 minutes for the remaining half.

	WELM			**CS baseline**		
	1 min.	4 min.	imb.	1 min.	4 min.	imb.
$W = 0$ s	53.8	60.9	58.3	44.6	49.5	51.6
$W = 10$ s	80.0	86.8	84.2	63.9	77.7	80.6
$W = 20$ s	88.6	95.3	90.8	72.2	88.8	86.9

TABLE 5: Re-Id accuracies obtained by the WELM and the CS baseline on 6 subjects using 1 and 4 minutes balanced training sets, and an imbalanced training set. The cumulative average window W is set to 0 s (a single test feature vector is used), 10 s or 20 s.

3) Improvement over a baseline: In Tab. 5, we compare the WELM to a baseline classification method widely used in camera-based person Re-Id [27] that, unlike milliTRACE-IR, does not learn a similarity score based on the actual distribution of the feature vectors at operation time. The baseline algorithm collects the training feature vectors along with the corresponding labels and computes the *centroid* of each class m in the NN feature space, denoted by $\boldsymbol{c}_m$. To re-identify a subject, the *cosine similarity* (CS) between his/her feature vectors, $\boldsymbol{v}$, and the centroid of each class m is computed, obtaining a similarity score $s_m = \boldsymbol{c}_m^T \boldsymbol{v}/(||\boldsymbol{c}_m||_2 \times ||\boldsymbol{v}||_2)$, and the classification is performed taking $\arg\max_m s_m$.

The WELM performs better than the baseline scheme across all the tests, see Tab. 5. The performance gap is significant for little training data (up to 16% improvement), small windows and imbalanced training sets.

VI. Concluding Remarks

In this work, we have designed and implemented milliTRACE-IR, the first system combining high resolution mm-wave radar devices and infrared cameras to perform non-invasive joint temperature screening and contact tracing in indoor spaces. This system uses thermal cameras to infer the temperature of the subjects, achieving measurement errors within 0.5 °C, and mm-wave radars to infer their spatial coordinates, by successfully locating and tracking subjects that are as close as 0.2 m apart. This is possible thanks to improvements along several lines, such as the association of the thermal camera and radar tracks from the same subject, along with a novel clustering algorithm combining density-based and Gaussian mixture methods to separate the radar reflections coming from different subjects as they move close to one another. Moreover, milliTRACE-IR performs contact tracing: a person with high body temperature is reliably detected by the thermal camera sensor and subsequently traced across a large indoor area in a non-invasive way by the radars. When entering a new room, this subject is re-identified among several other individuals with high accuracy (95%), by computing gait-related features from the radar reflections through a deep neural network and using a weighted extreme learning machine as the final re-identification tool.

References

[1] C. T. Nguyen, Y. M. Saputra, N. Van Huynh, N.-T. Nguyen, T. V. Khoa, B. M. Tuan, D. N. Nguyen, D. T. Hoang, T. X. Vu, E. Dutkiewicz, *et al.*, "Enabling and emerging technologies for social distancing: A comprehensive survey," *arXiv preprint arXiv:2005.02816*, 2020.

[2] T. P. B. Thu, P. N. H. Ngoc, N. M. Hai, *et al.*, "Effect of the social distancing measures on the spread of COVID-19 in 10 highly infected countries," *Science of The Total Environment*, vol. 742, p. 140430, 2020.

[3] F. de Laval, A. Grosset-Janin, F. Delon, A. Allonneau, C. Tong, F. Letois, A. Couderc, M.-A. Sanchez, C. Destanque, F. Biot, *et al.*, "Lessons learned from the investigation of a COVID-19 cluster in Creil, France: effectiveness of targeting symptomatic cases and conducting contact tracing around them," *BMC Infectious Diseases*, vol. 21, no. 1, pp. 1–9, 2021.

[4] A. S. Ali and Z. F. Zaaba, "A study on contact tracing apps for covid-19: Privacy and security perspective," *Webology*, vol. 18, no. 1, 2021.

[5] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006.

[6] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *2nd International Conference on Knowledge Discovery and Data Mining*, (Portland, Oregon, USA), Aug 1996.

[7] R. E. Kalman, "A new approach to linear filtering and prediction problems," *ASME Transactions, Journal of Basic Engineering*, vol. 82, (Series D), no. 1, pp. 35–45, 1960.

[8] A. Farhadi and J. Redmon, "Yolov3: An incremental improvement," *Computer Vision and Pattern Recognition*, 2018.

[9] M. I. Ribeiro, "Kalman and extended kalman filters: Concept, derivation and properties," *Institute for Systems and Robotics*, vol. 43, p. 46, 2004.

[10] S. Savazzi, V. Rampa, L. Costa, S. Kianoush, and D. Tolochenko, "Processing of body-induced thermal signatures for physical distancing and temperature screening," *IEEE Sensors Journal*, Early Access 2020.

[11] R. Faragher and R. Harle, "Location fingerprinting with bluetooth low energy beacons," *IEEE journal on Selected Areas in Communications*, vol. 33, no. 11, pp. 2418–2428, 2015.

[12] I. Galvan-Tejada, E. I. Sandoval, R. Brena, *et al.*, "Wifi bluetooth based combined positioning algorithm," *Procedia Engineering*, vol. 35, pp. 101–108, 2012.

[13] M. Cristani, A. Del Bue, V. Murino, F. Setti, and A. Vinciarelli, "The visual social distancing problem," *IEEE Access*, vol. 8, pp. 126876–126886, 2020.

[14] S. Bian, B. Zhou, H. Bello, and P. 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*, pp. 22–26, 2020.

[15] M. Rezaei and M. Azarmi, "Deepsocial: Social distancing monitoring and infection risk assessment in covid-19 pandemic," *Applied Sciences*, vol. 10, no. 21, p. 7514, 2020.

[16] A. J. Sathyamoorthy, U. Patel, Y. A. Savle, M. Paul, and D. Manocha, "Covid-robot: Monitoring social distancing constraints in crowded scenarios," *arXiv preprint arXiv:2008.06585*, 2020.

[17] N. Knudde, B. Vandersmissen, K. Parashar, I. Couckuyt, A. Jalalvand, A. Bourdoux, W. De Neve, and T. Dhaene, "Indoor tracking of multiple persons with a 77 GHz MIMO FMCW radar," in *European Radar Conference (EURAD)*, (Nuremberg, Germany), Oct 2017.

[18] P. Zhao, C. X. Lu, J. Wang, C. Chen, W. Wang, N. Trigoni, and A. Markham, "mID: Tracking and Identifying People with Millimeter Wave Radar," in *15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, (Santorini Island, Greece), May 2019.

[19] G. B. Dell'Isola, E. Cosentini, L. Canale, G. Ficco, and M. Dell'Isola, "Noncontact Body Temperature Measurement: Uncertainty Evaluation and Screening Decision Rule to Prevent the Spread of COVID-19," *Sensors*, vol. 21, no. 2, p. 346, 2021.

[20] C. Ferrari, L. Berlincioni, M. Bertini, and A. Del Bimbo, "Inner eye canthus localization for human body temperature screening," in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 8833–8840, IEEE, 2021.

[21] T. Lewicki and K. Liu, "AI thermometer for temperature screening: demo abstract," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pp. 597–598, 2020.

[22] R. Zhang and S. Cao, "Extending reliability of mmwave radar tracking and detection via fusion with camera," *IEEE Access*, vol. 7, pp. 137065–137079, Sep 2019.

[23] F. Nobis, M. Geisslinger, M. Weber, J. Betz, and M. Lienkamp, "A deep learning-based radar and camera sensor fusion architecture for object detection," in *2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pp. 1–7, IEEE, 2019.

[24] M. Ulrich, T. Hess, S. Abdulatif, and B. Yang, "Person recognition based on micro-doppler and thermal infrared camera fusion for firefighting," in *21st International Conference on Information Fusion (FUSION)*, pp. 919–926, IEEE, 2018.

[25] B. Vandersmissen, N. Knudde, A. Jalalvand, I. Couckuyt, A. Bourdoux, W. De Neve, and T. Dhaene, "Indoor person identification using a low-power FMCW radar," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, pp. 3941–3952, Jul 2018.

[26] Z. Meng, S. Fu, J. Yan, H. Liang, A. Zhou, S. Zhu, H. Ma, J. Liu, and N. Yang, "Gait Recognition for Co-Existing Multiple People Using Millimeter Wave Sensing," in *AAAI Conference on Artificial Intelligence*, (New York, New York, USA), Feb 2020.

[27] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, "Deep learning for person re-identification: A survey and outlook," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Early Access 2021.

[28] L. Fan, T. Li, R. Fang, R. Hristov, Y. Yuan, and D. Katabi, "Learning longterm representations for person re-identification using radio signals," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Seattle, Washington, USA), Jun 2020.

[29] Y. Cheng and Y. Liu, "Person reidentification based on automotive radar point clouds," *IEEE Transactions on Geoscience and Remote Sensing*, Early Access 2021.

[30] J. Pegoraro and M. Rossi, "Real-time People Tracking and Identification from Sparse mm-Wave Radar Point-clouds," 2021.

[31] S. M. Patole, M. Torlak, D. Wang, and M. Ali, "Automotive radars: A review of signal processing techniques," *IEEE Signal Processing Magazine*, vol. 34, pp. 22–35, Mar 2017.

[32] M. A. Richards, J. Scheer, W. A. Holm, and W. L. Melvin, *Principles of modern radar*. Raleigh, NC, USA: Scitech Publishing Inc., 2010.

[33] "How to find the right Thermal imaging camera," *DIAS Infrared GmbH*, https://www.dias-infrared.de/pdf/How-to-find-the-right-thermal-imaging-camera_DIAS-Infrared.pdf, 2020.

[34] S. Prince, *Computer Vision: Models Learning and Inference*. Cambridge University Press, 2012.

[35] W. Zong, G.-B. Huang, and Y. Chen, "Weighted extreme learning machine for imbalance learning," *Neurocomputing*, vol. 101, pp. 229–242, Feb 2013.

[36] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[37] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[38] T. Wagner, R. Feger, and A. Stelzer, "Radar signal processing for jointly estimating tracks and micro-Doppler signatures," *IEEE Access*, vol. 5, pp. 1220–1238, Feb 2017.

[39] L. Feng, S. Du, Z. Meng, A. Zhou, and H. Ma, "Evaluating mmWave Sensing Ability of Recognizing Multi-people Under Practical Scenarios," in *International Conference on Green, Pervasive, and Cloud Computing*, pp. 61–74, Springer, Dec 2020.

[40] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, 2009.

[41] Kuhn, Harold W, "The Hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[42] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Honolulu, Hawaii, USA), Jul 2017.

[43] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," in *The 9th ISCA Speech Synthesis Workshop*, (Sunnyvale, California, USA), Sep 2016.

[44] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[45] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European conference on computer vision*, (Amsterdam, Netherlands), Springer, Oct 2016.

[46] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in *IEEE conference on computer vision and pattern recognition (CVPR)*, (Boston, Massachussetts, USA), 2015.

[47] H. T. Huynh and Y. Won, “Regularized online sequential learning algorithm for single-hidden layer feedforward neural networks,” *Pattern Recognition Letters*, vol. 32, pp. 1930–1935, Oct 2011.

[48] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.