Enhancing a Telemedicine Platform with Global Navigation Satellite System Technology and Clustering Algorithms for Supporting Epidemiological Analysis

Silvia Panicacci
Dept. of Information Engineering
University of Pisa
silvia.panicacci@phd.unipi.it

Gianluca Giuffrida
Dept. of Information Engineering
University of Pisa
gianluca.giuffrida@phd.unipi.it

Massimiliano Donati
Dept. of Information Engineering
University of Pisa
massimiliano.donati@unipi.it

Alberto Lubrano
Dept. of Information Engineering
University of Pisa
alberto.lubrano@dii.unipi.it

Martina Olivelli
Dept. of Information Engineering
University of Pisa
m.olivelli@studenti.unipi.it

Alessio Ruiu
IngeniArs S.r.l.
Pisa, Italy
alessio.ruiu@ingeniars.com

Luca Fanucci
Dept. of Information Engineering
University of Pisa
luca.fanucci@unipi.it

***Abstract*— Telemedicine platforms have been largely used to manage multiple problems during the Covid-19 pandemic. In fact, they have given the possibility of remotely monitoring infected and high-risk patients, reducing hospitalisations. Telemonitoring systems with Global Navigation Satellite System technology allow to geo-localise all patients' measurements and enable the tracking of positions. These data can be used for contact tracing or to support doctors in epidemiological analysis. This paper presents the integration of satellite technologies in an existing telemedicine system (E@syCare), during the current outbreak. In particular, the platform has been enhanced with GPS, to geo-tag all vital parameters collected by the tablet gateway and the smartwatch. Geographical data are processed, after a request through the improved web-based medical interface based on some filters (e.g., vital parameters and their thresholds, considered period of time, and maximum cluster radius), with two sequential clustering algorithms. Agglomerative Clustering is used to find the optimal number of clusters given a maximum radius, and K-Means to effectively generate the predefined number of clusters. Resulting clusters are shown on an interactive epidemiological map in the web-based medical interface. This additional feature gives the possibility to healthcare authorities to correlate the spread of a disease or a virus with specific geographical areas or environmental conditions, to monitor fitness/movement habits of patients (also when the pandemic is over), and to track contact among patients.**

Keywords—covid-19, telemedicine, clustering, epidemiological map

I. INTRODUCTION

During the current pandemic, digital technologies have provided a great help to solve multiple issues, for communication, working, studying, as well as in public health [1],[2]. Multiple management problems have been faced by health administrators and authorities, such as how to monitor the clinical status of people treated at home, minimise the exposure of the medical staff, and track contacts and position of infected people subjected to quarantine regulations, early discharged patients, and people subjected to preventive quarantine [3]-[5].

Technology, and in particular telemedicine platforms can really help in facing the issues caused by the Covid-19 outbreak, as highlighted in multiple research and pilot studies, both for infected people and for people with previous chronic diseases [6]-[11]. More in detail, home monitoring systems allow to continuously remotely monitor the clinical status of people, thus enabling the adoption of a therapy similar to the one used during a hospitalisation, while reducing hospital admissions and mortality [12].

The idea of this project is to empower an existing telemedicine platform, called E@syCare [13] with Global Navigation Satellite System (GNSS) technology to provide enhanced functionalities for the final users (both healthcare personnel and patients) and a commercial smartwatch, featuring as minimum a GNSS receiver, a heart rate (HR) sensor and an accelerometer, to monitor physical activity and track patient positions [14]. The use of satellite technologies allows to geo-tag all measurements and data, giving the possibility to healthcare authorities to perform analysis of correlation between the spread of a particular disease or virus and a specific geographical area, identifying whether specific environmental condition favour the spread of certain diseases or virus. In addition, the geo-localisation of all observations of vital parameters collected by the medical sensors, exploiting the GNSS receiver on board of the smartwatch, enables the contact tracing, greatly reducing the infection spread of the virus. Finally, the availability of positioning information allows violation detection of movement restrictions and reconstruction of potential dangerous personal contacts during the virus emergency. Geo-localisation will be still important in the future, when the pandemic is over, to track movement habits of chronic patients (e.g., to study the correlation of chronic disease spread in specific areas).

This paper presents how GNSS technology has been integrated in the telemedicine platform and how all modules of the system have been improved to support the functionality of geo-tagging vital parameters at the moment of acquisition via the tablet gateway and the smartwatch. In addition, it describes the use of non-supervised algorithms to process these geographical data, tracking epidemic clusters, and the visualisation of the results in the web interface, by means of epidemiological maps for supporting doctors in contact tracing and eventually correlate the spread of the virus with particular geographical areas. After this introduction, Section II describes the general architecture of the telemedicine platform, underlying the geo-localisation additional feature. Section III is focused on the implementation of clustering algorithms and shows their integration in the telemedicine system. Finally, conclusions are drawn in Section 0.

II. The System

The E@syCare telemedicine platform is composed by three elements (Fig. 1): I) the cloud (server application), that is the brain of the entire system, II) the web interface, dedicated to general practitioners (GPs) and specialists for remote monitoring, and III) the data collector kit (composed by a tablet and a set of sensors), that acquires the vital parameters from the users. The entire system has a client-server architecture, where the cloud platform receives the information acquired by the tablets, and sends notifications or events to the users/GPs.

The patients data are fed into the cloud platform through the use of tablets and medical sensors. In particular, the tablet acts as gateway between the sensors and the cloud: each tablet is responsible to remind the user when to acquire a vital parameter (according to the personalised monitoring plan), communicate via Bluetooth (BT) or Bluetooth Low Energy (BLE) with the sensors to collect the measurement, and synchronise data with the cloud, while geo-tagging them.

Two versions of the mobile application have been developed: the Home application dedicated to home monitoring of patients, and the Professional application dedicated to the physicians to manage multiple patients.

More in detail, when patients are enrolled, they receive the Home kit, which includes a tablet with the Home application installed and set of sensors, such as thermometer and pulsoximeter. Since the monitoring of the physical activity

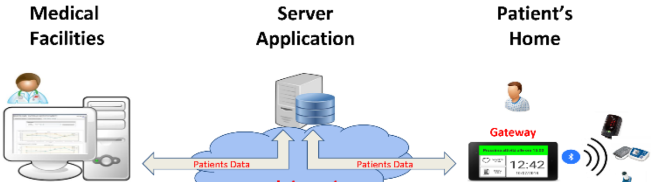

Fig. 1. System architecture.

represents a valuable information for the GP, the Home application supports also a smartwatch device. It can monitor indoor and outdoor activities, measuring HR, steps and the distance covered during the fitness. The added-value provided by the smartwatch is its integrated GPS sensor. This sensor returns the patient's position during the outdoor physical activity, and allows fine control if the tracking option is activated. The tracking option is designed to ensure respect for local restrictions, randomly taking users' position during the day, while checking whether the smartwatch is actually worn.

To increase the productivity of GPs and specialists, they receive a kit with the same sensors of the patients but the tablet contains the Professional version of the software, that is dedicated to the management of several patients at time.

The two versions of application geo-tag the patients' measurement, allowing to make some epidemiological analysis, such as maps, to contain and monitor the spread of Covid-19.

Through the web interface, which is composed by a dashboard dedicated to simplify the GPs and specialists operations, the doctors can define the personalised measurement plan for the patients (i.e., schedule the measurements, e.g., oxygen saturation, temperature and blood pressure, and set critical events according to the collected vital parameters, e.g., temperature higher than 38°C), and verify the measurements acquired by the patients and stored in the cloud.

The cloud is responsible to collect the patients data from the tablets and present them to the doctors in the web interface. In addition to the standard functionalities provided by a telemonitoring system, this improved version of cloud allows to:

- receive data from the smartwatch containing positions and activity tracking of the enrolled patients;
- generate an event in case the patient with movement restriction in pandemic periods has violated such a prescription. In case the alarm is generated, the position of the patient is traced to allow the so-called "contact tracing" which prevents the spread of the infection;
- generate an event in case the patient with movement restrictions is not wearing the smartwatch;
- receive geo-tagged cross-validated measurements from Home and Professional tablet applications;
- provide advanced functionalities for healthcare authorities including high resolution geo-maps of people affected by Covid-19 (in pandemic periods) or chronic diseases (in non-pandemic periods) to analyse the correlation between specific areas and its incidence in terms of virus infection or chronic diseases.

III. Data Analysis

The analysis of the data stored in the cloud may provide a big help to healthcare authorities, both in pandemic and non-pandemic periods, for example to analyse the correlation of the virus or a chronic disease with the environment. In fact, as already stated in Section II, each time a patient or a nurse executes a measurement, the position of the collector is acquired and saved at the same time. Moreover, exploiting the smartwatch GNSS receiver, it is possible to detect the position of the users during their daily life in random moments. The random position samplings guarantee the non-predictability of the measurements and provide a mitigation for the potential issue of patients wearing the smartwatch only for taking the measure.

The geo-tagging information can be used for the creation of an interactive map where illness concentrations, outbreaks or multiple violations are shown. The techniques used to perform these analysis are common clustering algorithms. Then, clustered data are correlated by the position or a chosen pathology. The results can be consulted by the doctors through the web-based medical user interface.

This represents a great tool to categorise illnesses due to factors related to geographic areas, such as pollution or allergens, or to diagnose a common pathology much more easier than before.

A. Identification of epidemic Clusters

The idea for the identification of clusters from geo-tagged data is to generate a non-a-priori known number of clusters in two dimensions (i.e., latitude and longitude of the patients' positions), with centroids and radii based on distance.

To the best of our knowledge, a clustering algorithm with these requirements does not exist. For this reason, we have split the problem in two phases, with the following goals:

1. find a pseudo-optimal number of clusters based on the distance among the samples;
2. generate the previous defined number of clusters with centroids that are not necessarily points of the input dataset, and with each sample member of only one cluster.

For the first phase, hierarchical clustering algorithms were the best candidates, especially Feature Agglomerative Clustering (FAC) [15] and Agglomerative Clustering (AC) [16]. Both algorithms exploit the agglomerative approach to combine the elements together and define clusters. The system to define the clusters is based on 4 different measures of distance:

1. single linkage, corresponding to the minimisation of the similarity between values belonging to different clusters (Eq. 1, where X_A belongs to cluster A and Y_B belongs to cluster B);
2. complete linkage, corresponding to the maximisation of the distance between the values in the clusters (Eq. 2);
3. average linkage, corresponding to the average distance among all clusters and all elements (Eq. 3);
4. ward linkage, corresponding to the analysis of the variance of the clustering (Eq. 4, where n_A and n_B are respectively the number of elements in cluster A and B).

$$d_{A,B} = \min d(X_A, Y_B) \quad (1)$$

$$d_{A,B} = \max d(X_A, Y_B) \quad (2)$$

$$d_{A,B} = \frac{1}{kl} \sum_{i=1}^{k} \sum_{j=1}^{l} d(X_A, Y_B) \quad (3)$$

$$d_{A,B} = \frac{n_A n_B}{n_A + n_B} ||X_A - Y_B||^2 \quad (4)$$

However, FAC was excluded because it tries to cluster features instead of samples and it is a dimensionality reduction tool, to be used with more than two dimensions. On the contrary, AC recursively merges the pair of clusters that minimally increases a given linkage distance: it starts with N groups, each containing one element, and then the most similar groups are merged recursively, until the maximum distance is reached. In this way, the user could decide the maximum radius of the clusters. The output of this algorithm is the optimal number of clusters to generate with the selected radius. Because of these characteristics, AC with ward distance was definitively selected to calculate the number of clusters, taking in input longitudes (x), latitudes (y) and distance (threshold).

Once the number of clusters is generated by the AC algorithm, another clustering algorithm is needed to effectively cluster the samples. In this case, the choice was between Affinity Propagation (AP) [17] and K-Means [18]. AP was excluded because centroids always coincide with an input sample, which is not a requirement for our application. Conversely, K-Means tries to separate samples in groups with respect to their variance, generating a number of clusters provided in input, with centroids usually not coincident with an input sample. The output of the K-Means algorithm is an array of centroids coordinates. Its objective function is shown in Eq. 5, where x_i represents an element, μ_k represents a cluster, and w_{ik} denotes the membership of the x_i element in the cluster μ_k (w_{ik} is equal to 1 if xi belongs to μ_k, otherwise it is equal to 0). Then, K-Means was selected for the second phase of clustering, taking in input longitudes (x), latitudes (y) and the number of clusters previously calculated.

$$J = \sum_{i=1}^{m} \sum_{k=1}^{K} w_{ik} ||x_i - \mu_k||^2 \quad (5)$$

Finally, for each cluster, the radius is computed, taking the distance between the centroid and the farthest sample belonging to the cluster itself.

Fig. 2 shows some examples of clustering results, starting from 300 geo-localised random points (Fig. 2(a)), obtained launching a Python script which sequentially executes AC and K-Means, as described above. Longitude is on the x-axis, while latitude is on the y-axis. Of course, a smaller input maximum radius generates more clusters (Table 1) and requires more overall computational time. As shown in Table 1, K-Means execution time varies with the number of input

TABLE I. EXECUTION TIME VS 4 DIFFERENT INPUT RADII AND 300 GEO-LOCALISED POINTS (FIG. 2(A)).

Input Radius (km)	Number of Clusters	AC Execution Time (ms)	K-Means Execution Time (ms)
5	98	27	454
20	29	28	185
50	11	29	135
100	6	27	121

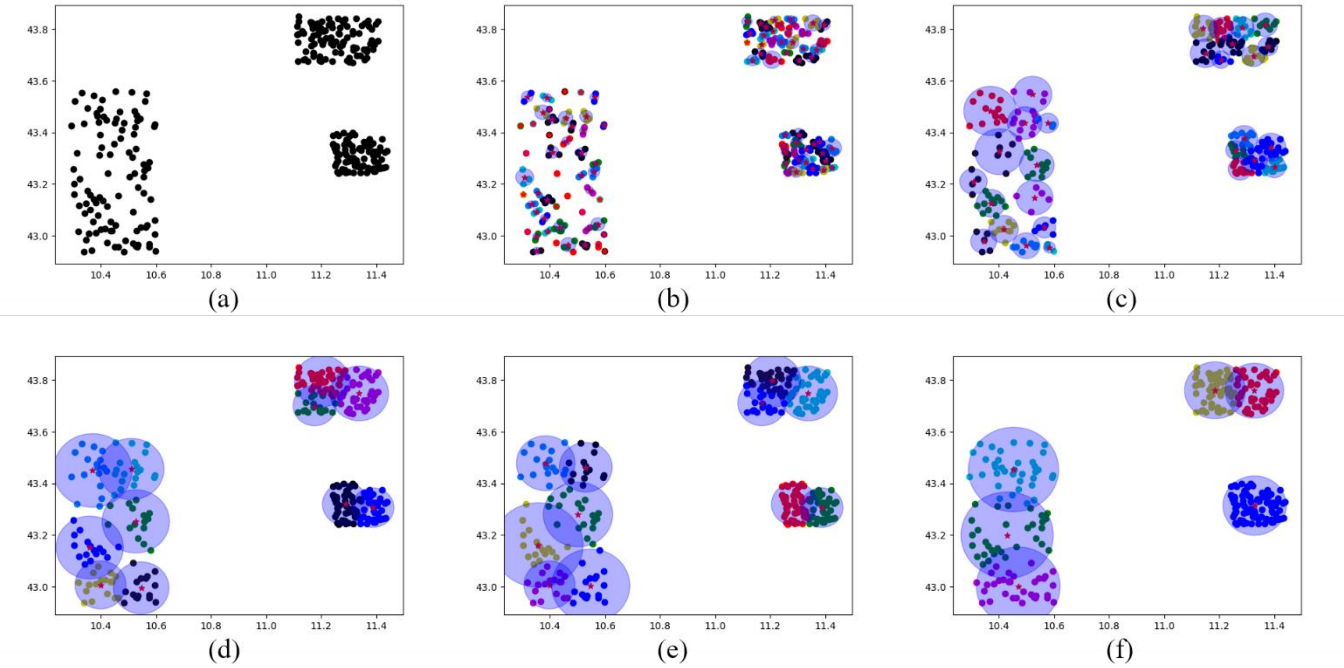

Fig. 2. Example of results with AC and K-Means on 300 random geo-localised points (x is longitude, y is latitude): (a) initial random points; (b) clustering with maximum radius 5 km; (c) clustering with maximum radius 20 km; (d)-(e) clustering with maximum radius 50 km; (f) clustering with maximum radius 100 km.

clusters to generate, while AC spends more or less the same time regardless of input radius (about 28 ms for the examples reported in Fig. 2), since the complete tree is always computed. Finally, from Fig. 2(d) and Fig. 2(e) it is possible to note that, even if the number of clusters generated with the same input radius is the same (in this case, with radius equal to 50 km, 11 clusters are generated), and so AC is a deterministic algorithm, K-Means achieves different results (i.e., centroids and radii) with the same input, given by the initial random choice of the centroids.

B. *Integration in the Telemedicine System*

To integrate the Python script which executes the two clustering algorithms in the cloud module of the telemedicine platform, we implemented a Java wrapper, whose architecture is shown in Fig. 3.

Since data protection and privacy issues may be raised by the use of this script, the wrapper acts as an intermediator, by extracting the relevant patients' information and providing only fake ids and positions. This solution allows to lose the dependence between data and patients, while maintaining the functionality to draw the epidemiological map.

The patients to be considered for clustering and their positions are filtered directly in the web interface, where the user (i.e., a doctor) can choose the vital parameters and their thresholds (e.g., temperature greater than 37°C), the reference period (e.g., the last month), and the maximum radius for clusters. After analysis, clustered data are visualised in an epidemiological map, drawn by exploiting Google Maps

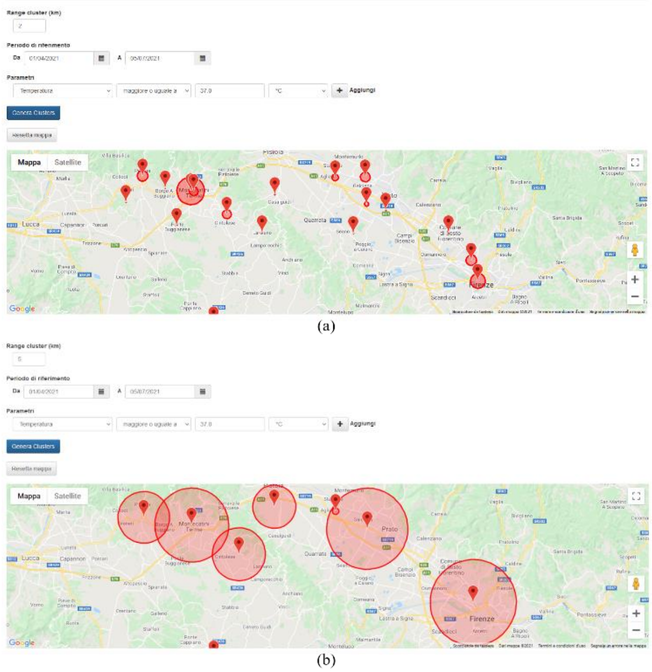

Fig. 3. Example of epidemiological maps, selecting patients with temperature greater than 37°C from 1st April 2021 to 5th July 2021 with maximum radius equal to (a) 2 km and (b) 5 km.

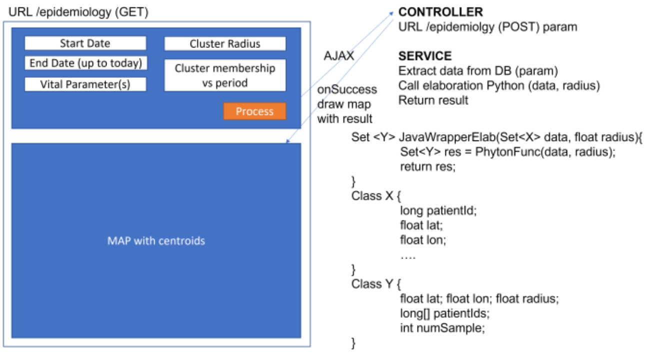

Fig. 4. Architecture of the Java wrapper for the integration of Python clustering script.

APIs, generating an interactive map with pop-ups. In particular, within the map several position markers are drawn, each one representing a cluster centroid, and circles around them, with radius equal to the radius of the considered cluster, calculated as the distance between the centroid and the farthest sample belonging to the cluster.

Examples of interactive maps integrated in the cloud of the telemedicine system are shown in Fig. 4: the patients selected have some measurements of temperature greater than or equal to 37°C in the period from 1st April 2021 to 5th July 2021 with different radii (2 and 5 km, respectively).

IV. Conclusion

Informatisation has provided a great help in multiple fields, and especially in public health, during the Covid-19 pandemic. Telemedicine platforms have been extensively used for remote monitoring, allowing treatment and monitoring of clinical status of people treated at home while minimising the exposure of the medical staff and reducing the impact on the healthcare system. The use of satellite technologies, e.g., Global Navigation Satellite System, enables the geo-localisation of patients' measurement, with the possibility of further analysis, such as contact tracing or correlation between the spread of the virus and a geographical area.

This paper presents the integration of this satellite technology in the E@syCare telemedicine platform, with the aim of geo-tagging vital parameters and analysing data with clustering algorithms to draw epidemiological maps to support doctors. The clustering algorithms are in two dimensions (longitude and latitude), avoiding the use of patients information for data protection and privacy: Agglomerative Clustering is used to generate the pseudo-optimum number of clusters given the maximum cluster radius; K-Means starts from the previously calculated number of clusters and effectively generates clusters, with centroids and radii. Through the web-based medical interface, doctors can filter patients, with vital parameters and thresholds and period of time, select the maximum cluster radius, and visualise clustering results in an epidemiological map, drawn by exploiting Google Maps APIs.

Acknowledgment

This research was partially funded by European Space Agency (ESA), in the framework of the Invitation To Tender "Space in response to Covid-19 outbreak".

References

[1] G. Goggin, "COVID-19 apps in Singapore and Australia: reimagining health nations with digital technology," *Media International Australia*, vol. 177, no. 1, pp 61-75, 2020, doi: 10.1177/1329878X20949770.

[2] S. Whitelaw, M. A. Mamas, E. Topol, and H. G. C. Van Spall, "Applications of digital technology in COVID-19 pandemic planning and response," *The Lancet Digital Health*, vol. 2, no. 8, pp. e435-e440, 2020, doi: 10.1016/S2589-7500(20)30142-4.

[3] L. Lai *et al.*, "Digital triage: Novel strategies for population health management in response to the COVID-19 pandemic," *Healthcare*, vol. 8, no. 4. Elsevier, 2020, p. 100493, doi: 10.1016/j.hjdsi.2020.100493.

[4] J. Amann, J. Sleigh, and E. Vayena, "Digital contact-tracing during the Covid-19 pandemic: An analysis of newspaper coverage in Germany, Austria, and Switzerland," *PLOS ONE*, vol. 16, no. 2, pp. 1-16, 2021, doi: 10.1371/journal.pone.0246524.

[5] I. Braithwaite, T. Callender, M. Bullock, and R. W. Aldridge, "Automated and partly automated contact tracing: a systematic review to inform the control of COVID-19," *The Lancet Digital Health*, vol. 2, no. 11, pp. e607-e621, 2020, doi: 10.1016/S2589-7500(20)30184-9.

[6] E. Monaghesh and A. Hajizadeh, "The role of telehealth during COVID-19outbreak: a systematic review based on current evidence," *BMC Public Health*, vol. 20, no. 1, pp. 1–9, 2020, doi: 10.1186/s12889-020-09301-4.

[7] S. Panicacci, M. Donati, A. Lubrano, A. Vianello, A. Ruiu, L. Melani, A. Tomei, and L. Fanucci, "Telemonitoring in the Covid-19 Era: the Tuscany Region Experience," *Healthcare*, vol. 9, no. 5, pp. 516, 2021, doi: 10.3390/healthcare9050516.

[8] J. G. Cleland, R. A. Clark, P. Pellicori, and S. C. Inglis, "Caring for people with heart failure and many other medical problems through and beyond the COVID-19 pandemic: the advantages of universal access to home telemonitoring," *European journal of heart failure*, vol. 22, no. 6, pp. 995-998, 2020, doi: 10.1002/ejhf.1864.

[9] G. Furlanis, M. Ajčević, M. Naccarato, P. Caruso, I. Scali, C. Lugnan, A. B. Stella, and P. Manganotti, "e-Health vs COVID-19: home patient telemonitoring to maintain TIA continuum of care," *Neurological Sciences*, vol. 41, no. 8, pp. 2023-2024, 2020, doi: 10.1007/s10072-020-04524-0.

[10] S. Panicacci, M. Donati, F. Profili, P. Francesconi, and L. Fanucci, "Trading-off Machine Learning Algorithms towards Data-Driven Administrative-Socio-Economic Population Health Management," *Computers*, vol. 10, no. 1, pp. 1-21, 2021, doi: 10.3390/computers10010004.

[11] A. V. Silven *et al.*, "Telemonitoring for Patients With COVID-19: Recommendations for Design and Implementation," *Journal of Medical Internet Research*, vol. 22, no. 9, p. e20953, 2020, doi: 10.2196/20953.

[12] M. Donati, A. Celli, A. Ruiu, S. Saponara, and L. Fanucci, "A Telemedicine Service Platform exploiting BT/BLE Wearable Sensors forRemote Monitoring of Chronic patients," *2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, 2018, pp. 1–4, doi: 10.1109/MOCAST.2018.8376643.

[13] M. Donati, A. Celli, A. Ruiu, S. Saponara, and L. Fanucci, "A Telemedicine Service System exploting BT/BLE Wireless Sensors for Remote Management of Chronic Patients," *Technologies*, vol. 7, no. 1, 2019, doi: 10.3390/technologies7010013.

[14] S. Panicacci, G. Giuffrida, M. Donati, A. Lubrano, A. Ruiu, and L. Fanucci, "Empowering Home Health Monitoring of Covid-19 Patients with Smartwatch Position and Fitness Tracking," *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)*, 2021, pp 349-353, doi: 10.1109/CBMS52027.2021.00109.

[15] "Feature agglomeration," "https://scikit-learn.org/stable/modules/generated/sklearn.cluster.FeatureAgglomeration. html".

[16] "Agglomerative clustering," "https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering. html".

[17] "Affinity propagation," "https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AffinityPropagation. html".

[18] "K-means," "https://scikit-learn.org/stable/modules/generated/sklearn.cluster. KMeans.html".