INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2020, VOL. 66, NO. 1, PP. 1–6
Manuscript received October 10, 2019; revised January, 2020. DOI: 10.24425/ijet.2019.126xxx

Position Estimation in Mixed Indoor-Outdoor Environment Using Deep Learning and Signals of Opportunity

Sebastian Urwan, Dominika R. Wysocka and Alicja Pietrzak, Krzysztof K. Cwalina

***Abstract*—Abstract - To improve the user's location in indoor and outdoor environment a novel radiolocalization system is proposed. It is based on the radio sygnatures method using radio signals from LTE an WiFi networks and is assisted with deep learning algorithm. First, the measurements of RSSI, RSRP and RSRQ from LTE network and RSSI from WiFi network are taken. Then after data preprocessing the neural network input vectors are created and the neural network system's models are trained. And finally the user's position is calculated with a trained neural network system's models. Additionaly the influence of various number of measurements from LTE and WiFi networks in the input vector on the positioning accuracy was examined. From the results we can see that using deep learning algorithms with a radio signatures method can can result in less positioning error comparing to the GPS system in indoor environment. What is more, the combination of LTE and WiFi signals measurement in an input vector results in better indoor and outdoor as well as floor classification accuracy and less positioning error comparing to the input vector consisting measurements from only LTE network or from only WiFi network.**

***Keywords*—radiolocalization, deep neural network, DNN, hybrid localization**

I. Introduction

A demand for indoor and outdoor localization is constantly growing due to the need of increasing people's safety and comfort. Indoor localization is used in order to navigate users in shopping centers, hospitals or in underground parking lots [1], [2] and outdoor localization can be used to navigate autonomous cars [3]. Position estimation in both indoor and outdoor environment will be crucial in creating smart cities in the future. What is more, it is needed to use the already existing radiocommunication infrastructure for the system to be cheaper and more accessible.

Global Positioning System (GPS) which is the most used radiolocalization system nowadays is not accurate enough, especially indoors [1]–[4]. Using a satelite system makes it more vulnerable to multipath propagation, lack of line of sight (LOS) or signal scattering. It is especially noticeable in indoor environment making it nearly impossible in practical usage.

There are also localization methods that use measurements of a Received Signal Strength (RSS) from base stations (BS) of cellular networks or access points (AP) of Wireless Fidelity (WiFi) networks [2], [3], [5]. The biggest advantage of this approach is that it uses an already existing infrastructure, and what is important, in terms of WiFi an infrastructure created inside of buildings.

In this article we propose a localization system based on the RSS measurements and a fingerprinting method [6], [7]. What is more, designed position estimation system is assisted with deep learning (DL) algorithms. Taking into account the non-linearity of localization process the demand for using DL algorithms and the number of research papers using DL algorithms in localization are rising [8], [9]. Given the fingerprinting method it can be used in measurement phase [10], [11] or in localization estimation phase [12].

In existing research it is common to consider either indoor or outdoor localization with relatively small localization areas [12], [13]. Researchers also usually take measurements of only WiFi RSS indicator (RSSI) [8]–[10], [12], [14], [15]. Innovative radiolocation system described in this paper locates users in both indoor and outdoor environments and works on the basis of measurements from both Long Term Evolution (LTE) network and WiFi. Additionally it is supported with DL algorithms.

The main contributions of the paper are:

- We designed an innovative indoor and outdoor radiolocalization system that uses measurements from LTE and WiFi networks and is supported with DL algorithms.
- We created a measurement Android application that measures WiFi RSSI, LTE RSSI, LTE Reference Signal Receive Power (RSRP) and LTE Reference Signal Receive Quality (RSRQ). With this application we carried out the measurements, generated a fingerprinting database and formed data sets.
- We implemented a system of deep neural networks (DNN). System of DNN takes a vector of measurements datasetS from one user and calculates user's position. DNN were trained and tested with created earlier data sets. The results show that the hybrid approach of using signals from both LTE and WiFi network gives more accurate position estimation than using signal from only one network.

This work was supported by the grant No. xxxxxx "The research of new instruments in LaTeX" financed from internal science fund of University of Technology.

S. Urwan, D.R. Wysocka and A. Pietrzak (e-mail: s171751, s171562, s171921@student.pg.edu.pl).

K. K. Cwalina is with Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology, 80-233 Gdansk, Poland; (e-mail: kkcwalina@pg.edu.pl).

The rest of the paper is organized as follows — Section II presents a discussion of related work on the WiFi and LTE signal based positioning systems using DNN algorithms. In Section III we described a system prototype. Conducted measurement scenarios are presented in section IV. The result analysis of the proposed system is discussed in Section V and we conclude our paper in Section VI.

II. Related works

Deep learning based localization using signals from LTE or WiFi networks has been studied in the past. In this section we present different approaches from previous research. Authors of those articles usually take into consideration only an indoor localization.

When using a fingerprinting method the process of collecting measurements from a vast area can be time and memory consuming. In [10] authors created radio maps using deep gaussian process which describes a relation between RSS measurements and position in which the measurements were taken. Thanks to this approach they constructed a radio map using only 20% of measurements similar to the one constructed with 100% of measurements.

Using only measurements of WiFi RSS can result in small localization precision. To address this problem, a hybrid RSS and channel state information (CSI) system is proposed in [8], [12]. In [8] fingerprint database is made of RSS and CSI with a high correlation. Measurements from 28 reference points (RP) were collected in 16x8 m^2 room. About 90% of location errors is less than 1.5 m in this system . In [12] it has been found that DL using CSI signals achives better localization accuracy than DL using RSS signals. Convolution neural network (CNN) using CSI signals achieved maximal localization error of 0.92 m with probability of 99.97 %.

Giving the non-linearility and complexity of position estimation using RSS signals researchers tend to implement a systems of different algorithms [14]–[16]. In [14] authors proposed a DL system integrating CNN, Siamese architecture and regression network. Proposed method achieved mean positioning error of 1.3 m in the 80x20 m^2 area with the fast-moving user. In [15] system which includes DNN, CNN, Dempster-Schafter theory and AutoEncoder in the 14.4x7.2 m^2 room achieved Root Mean Square Error (RMSE) of 1.5 m. In [16] authors used pedestrian dead reckoning with WiFi weighted path loss algorithm and linear Kalman filter. This system uses accelerometer, gyroscope and magnetometer sensors. With the path over a rectangular of size 29x45 m the maximum positioning error was 1.5 m

In [17] authors decided to consider a heterogenous indoor localization system including measurements from both LTE and WiFi network. They also examined localization accuracy when using only LTE signals or only WiFi signals. In the localization area of two 3,5x4,5 m^2 rooms the smallest RMSE localization error was 0.9 m for the combination of both networks.

In the table I we presented a summary of already described in this chapter and other published articles in which authors proposed radiolocalization fingerprinting systems supported with DL algorithms.

TABLE I
SUMMARY OF EXISTING RESEARCH ABOUT RADIOLOCALIZATION FINGERPRINTING SYSTEMS SUPPORTED WITH DL ALGORITHM

Article	Area size	Data sources	Algorithm	Performance
Indoor localization				
[8]	16x10 m^2	WiFi	DNN	90% errors less than 1.5 m
[9]	390x270 m^2	WiFi	DNN	5 m - max error
[10]	2300 m^2	WiFi	DNN	1.3 m - mean error
[12]	13,82x8,56 m^2	WiFi	DNN, CNN	0.9 m - max error
[14]	80x20 m^2	WiFi	DNN, CNN	1.2 m - mean error
[15]	14,4x7,2 m^2	WiFi	DNN, CNN	1.5 m - mean error
[17]	2 3,5x4,5 m^2 rooms	WiFi, LTE	DNN	0.9 m - mean error
[18]	2 3000 m^2 floors	WiFi	Bayes filter, hidden Markov Model	1.9 m - mean error
[19]	10x10 m^2 floors	WiFi	Weighted Fuzzy Matching, Kalman Filter	0.4 m - mean error
Outdoor localization				
[13]	60x60 m^2	WiFi, LTE	DNN	0.4 m - mean error
[20]	100x100 m^2	simulation RSS	DNN	5.5 m - mean error

From the analysis of existing research we can conclude that there is a lack of systems combining both indoor and outdoor radiolocalization. We also can notice that there aren't many papers in which authors designed a heterogeneous systems with measurements from both LTE and WiFi networks.

III. System prototype

The system created as part of this project aims to locate users in indoor and outdoor environments. Locating users is based on the radio signature method and the deep learning algorithm. Due to the use of fingerprinting, it is necessary to collect measurements of signal strength from WiFi and LTE wireless networks selected in this project. Moreover, the system works on publicly available mobile phones. Therefore, the program implementation of the project assumes the creation of a measuring station in the form of an application for mobile phones with the Android system, used to download information about signals from WiFi and LTE networks. No need to purchase additional specialized devices, only the use of the existing infrastructure of selected networks makes the construction of the system universal at the expense of less accurate measurements. The collected data is stored in a dedicated database. After collecting the necessary information about the signals measured by the user's device, the system determines its location. A deep learning algorithm, specifically a feedforward neural network, was chosen as the locating algorithm. In the training phase of the neural network, a genetic algorithm was used, which is responsible for the determination of such hyperparameters defining the network architecture, for which the trained model obtains the lowest possible user location error. However, in the test phase, the collected vector of input data is processed by the selected trained network model.

A. Concept

For the purpose of this paper developed positioning system requires only generally available mobile phones with Android system. As there is no need to use additional specialized devices, the use of the existing infrastructure of LTE and WiFi networks makes the construction of the system universal at the expense of less accurate measurements. The collected data is stored in a dedicated database. After collecting the necessary information about the signals measured by the user's device, the system determines its location. A deep learning algorithm, specifically a feedforward neural network, was chosen as the locating algorithm. In the training phase of the neural network a genetic algorithm was used, which is responsible for the determination of such hyperparameters of the neural network, for which the trained model obtains the lowest error of user's estimated location. In the test phase, the collected vector of input data is processed by the selected trained network model.

B. Android measurement application

Measurement application prototype was developed for mobile phone with Android system to collect information for fingerprinting positioning system and send it to the external server. According to the structure and stack of Android system there is a possibility to obtain information about radio signal derived from LTE and WiFi networks via Java. For this project authors used three Xiaomi Redmi Note 8 Pro smartphones. See specification in a Table II.

TABLE II
USED SMARTPHONES.

User	Android ver.	API level	Build number	Operator
1	11	30	RP1A.200720.011	Plus
2	10	29	QP1A.190711.020	T-Mobile
3	10	29	QP1A.190711.020	Orange

The object-oriented programming language Java 1.8 and the Android Studio 4.2 programming environment under the open-source license were used to implement all the functions of the measuring application. The application implements a method that periodically reads information about the signals received by the mobile phone, emitted by nearby LTE eNB, nearby WiFi AP and GPS satellites. Measurements from the GPS system were performed in order to compare the location errors of the designed system with the existing radiolocation system. The frequency of updating the read parameters was assumed on the basis of the article [21]. Additionally, in accordance with the design assumptions, in order to identify and distinguish users and check their movement history, a unique identifier of user equipment and the current time

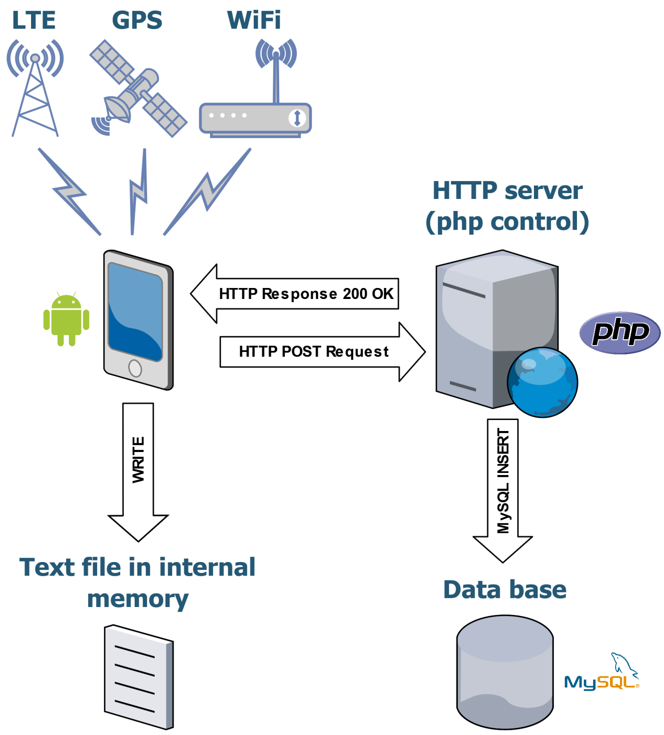

Fig. 1. Diagram of used components and their connections in the designed radiolocation system in the part of data acquisition.

stamp are read from the Android system. In the described measuring application, the functionality of saving the collected information locally in the devices' memory has also been implemented in order to protect against the loss of data sent via the Internet to the external server. Measurements can be performed in two modes: measurement performed for a specified period of time or measurements performed for an indefinite period of time until the user stops the application.

On the HTTP server side, incoming requests (POST) of the mobile application are directed to port 80 and handled by a script written in PHP language. The mobile application through the Volley library at the moment of creating a request to send each set of measurement data creates listeners responsible for error handling, process of connecting the application with the server and handling the 200 OK message from the server. In addition, the PHP code on the server verify users and allow database to create new data records. One cycle of the processing and transmission loop of the measurement data set is completed upon receipt of the server feedback on the status of the operation performed. Described part of the radiolocation system is presented in Fig. 1.

There are few limitations associated with Java programming language for Android and application layer of Android system. First of all, reffering to Android's documentation [23] there is a correlation between the version of Android installed on the mobile phone and available Java language methods. All used methods and fields are presented in Table III.

The analysis presented above shows the following limitations in the designed radiolocation system for mobile phones with the Android system:

TABLE III
USED SMARTPHONES.

Classes CellInfo, CellIdentityLte, CellSignalStrengthLte		
Method name	Min API level	Proper functioning on tested phone
getPci	17	Yes
getTimeStamp	17	Yes
getRsrp	26	Yes
getRsrq	26	Yes
getRssi	29	Yes/No
getCqi	26	No
getRssnr	26	No
Class ScanResult		
Field name	Min API level	Proper functioning on the phone
timestamp	17	Yes
frequency	1	Yes
SSID	1	Yes
BSSID	1	Yes
level	1	Yes
Class Location		
Method name	Min API level	Proper functioning on the phone
getLatitude	1	Yes
getLongitude	1	Yes
getAltitude	1	Yes

- API level versions consist of different lists of usable classes and methods, which excludes the use of e.g. getRssi method,
- the correct operation of the methods that can be used in Android Studio for each API level depends on the decisions of Android mobile phone manufacturers regarding the implementation of individual methods in their devices, e.g. the getCqi method, despite the availability for API level 29, does not function properly on mobile phone Xiaomi Redmi Note 8 Pro,
- the application for the planned radiolocation system is not possible to implement on every mobile phone with the Android system, which guarantees correct operation.

C. Fingerprinting positioning method

Fingerprinting method assisted with DL algorithms consists of offline and online phase.

1) offline phase: During an offline phase the signal's measurements are taken in earlier chosen reference points (RP). In this project we decided to collect measurements of WiFi RSSI, LTE RSSI, LTE RSRP and LTE RSRQ. As a result a radio map is generated and with data preprocessing deep neural networks' input vectors are created. Input vectors with corresponding reference coordinates are then used to train neural network models.

2) online phase: In an online phase we collect testing data. The same measurements as in an offline phase are taken and with data preprocessing input vectors are created which are then processed by trained neural network model and an estimated user's position is calculated.

D. Data preprocessing

During a data analysis we realized that WiFi RSSI has a value of 2147483647 sometimes, meaning that it was impossible to measure it. We decided to change those values to zero. What is more, BSSID was in a non-numeric format which is not possible to be processed with neural network so we mapped it to a numeric format.

One of the goals of this paper is to examine an input vector configuration influence on position estimation therefore we created vectors consisting of different number of measurements from WiFi and LTE networks. In case of not having enough measurements we filled the gaps with zeros [22].

For the outdoor scenario we also changed measured reference longitude and latitude to the ECEF coordinates which made it easier to calculate RMSE in meter unit.

In order to process data with neural network it is necessary to normalize it. It improves neural network calculations and decreases training error. We decided to normalize our data by scaling it to a range of (0, 1) with an equation (1):

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

where:

x_{norm} - normalized value
x - not normalized value
x_{min} - minimum range of data
x_{max} - maximum range of data

E. Deep learning

Given the non-linearity of position estimation we have chosen a deep learning algorithm as localization algorithm.

1) Deep neural networks: We used feedforward neural networks in order to minimize computational workload at the expense of position estimation accuracy, comparing to the usage of convolutional networks.

In order to obtain the smallest possible error in user localization, a hierarchical deep learning solution was developed. In the context of user localization, the hierarchical neural network system developed in this project, in which indoor and outdoor localization was performed, is first to determine whether the user is outside or inside the building. For the outdoor classification, the location of the user's position in the outside area is determined. On the other hand, if it has been assigned indoors, the classification of the floor on which the user is located takes place. Once the floor is designated, the location of the user on the designated floor is determined.

The number of hidden layers, the number of nodes in each hidden layer, the number of training epochs, the mini-batch size and the learning rate were selected using a genetic algorithm. The Rectified Linear Unit (ReLU) activation function and the Adaptive Moment Estimation (ADAM) algorithm were used in the hidden layers in either regression and classification networks. Additionally, in the output layers of regression networks as well as binary and multi classification networks, the linear, sigmoidal and softmax activation functions were used respectively. For classification networks the error function is the cross entropy function, while for regression networks the error function is the mean square error function.

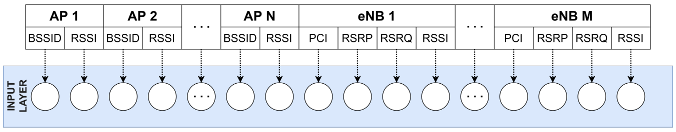

Fig. 2. Construction of the input vector of a deep neural network.

The position estimation error obtained from regression network models is described by RMSE which is given in meters. The accuracy of the classification network models is determined by the percentages [0% - 100%].

Deep neural networks models were implemented with the use of programming language Python 3.9 and the library Pytorch 1.9. All models were trained and tested on a Windows 10 21H1 workstation with Ryzen 5 3600 CPU (6-core 3,59 GHz), 32 GB RAM, GTX 1660 Super (GPU) and 512 GB SSD.

2) Input vector structures: As mentioned earlier, based on the fingerprinting method, the input data vector for the neural networks consists of measured RSRP, RSRQ and RSSI values from LTE networks and RSSI from WIFI networks. Moreover, it was decided to investigate the effect of the number of selected signal sources on the localization accuracy. Localization accuracy was also investigated using measurements only from WIFI or only from LTE. In fig. 2 is shown the structure of the input data vectors for configurations with N signal sources from WIFI and M signal sources from LTE.

Reference vectors, which are necessary in supervised learning, have to be defined for both analyzed neural network model type (classification and regression). For the classifier network, the reference vector consists of a single number specifying the class isIndoor for indoor and outdoor environment classification or floorNumber for floor classification. For the outdoor regression, the reference data are the longitude and latitude converted to ECEF system coordinates: RefxLat, RefyLon and Refz. For the indoor regression, the reference data are the x, y coordinates fixed in the local coordinate system: RefxLat and RefyLon.

3) Supervised learning: The processed input data is then divided into training, validation and test parts in a ratio of 60:20:20. The training and validation parts are divided into subsets with a given number of input vectors.

The training and validation process continues for a predetermined number of iterations or until an interruption condition is met. In the design, the network training is interrupted if during the next 30 iterations the validation RMSE increases or the difference between the training and validation error is greater than a predetermined threshold and during the next 30 iterations the validation RMSE does not decrease. This is necessary to avoid overtraining the network as well as to reduce the training time when smaller training error values are not achieved. Importantly, training is interrupted and the saved neural network model is the stored model from before the 30 iterations countdown began.

4) Hyperparameters tuning: The effectiveness of deep neural networks operation depends on many factors, including hyperparameters or noise of the data provided to the input layer. The selection of hyperparameters of deep neural networks can be performed by grid search algorithm, random search algorithm or genetic algorithm [24], [25]. In this project, authors decided to implement a genetic algorithm in order to search a limited set of values of the tested hyperparameters for the optimal model architectures for each of measurement scenarios. Table IV present ranges of values of all tested hyperparameters.

TABLE IV
RANGES OF VALUES OF TESTED HYPERPARAMETERS.

Hyperparameter	Values Range for Classification	Values Range for Regression
Number of Hidden Layers	{1, 2, 3, 4, 5, 6}	
Number of Nodes	{10, 20, 30, ..., 400}	
Number of Epochs	{10, 11, 12, ..., 100}	max 500
Batch size	{32, 64, 128, 256, 512}	
Learning Rate	{1e-5,5e-5,1e-4,5e-4,1e-3,5e-3,1e-2,5e-2,1e-1}	

The presented ranges were selected based on the research described in [25]. Depending on the problem contained in the data set, the ranges of values of hyperparameters vary and similar effectiveness of the network operation can be achieved for extremely divergent values of the hyperparameters. The selection of a wide range of hyperparameter values for the grid search algorithm and the random search algorithm will allow to increase the number of possible solutions giving similar network efficiency results. However, such a choice will increase algorithm's operating time. According to the literature and conducted research, the use of a genetic algorithm allows for an effective search of a multidimensional set of hyperparameters [25]. This results in a reduction of the time needed to find the optimal combination of hyperparameters.

In order to function properly, the genetic algorithm requires the determination of the conditions for interrupting its operation. There are many methods and approaches used in the literature, however, in the implementation of this project, two mechanisms of breaking the genetic algorithm were used:

- when a certain number of generations of solutions were exceeded,
- when the most effective solution of a given population reaches a certain threshold of the accuracy of the classification or the accuracy of the location estimate.

In this project it was established that the limit value of the iteration of the genetic algorithm will be 10, although other values are also used in scientific works, e.g. 5 [29]. It is worth emphasizing that this number may directly affect the results of the searched space. Consequently, a small number of iterations may not lead to finding the optimal solution. On the other hand, too many iterations can significantly increase the working time of the algorithm by searching the multidimensional space of hyperparameters more carefully. For each tested measurement scenario a model effectiveness threshold was defined upon reaching which the genetic algorithm would be interrupted. The threshold values have been determined on the basis of scientific research conducted in the field of radiolocation and the results achieved, summarized in Table

I. In order to interrupt the work of the genetic algorithm at the right moment, i.e. taking into account the computational complexity and the result in the form of the most effective architecture found, the method of comparing the effectiveness between successive models of deep neural networks can also be used. Small differences in values, e.g. the root of the mean square error between the best models after several iterations of the genetic algorithm may indicate the lack of new, better solutions on the examined plane.

Fig. 3 shows a block diagram of a genetic algorithm implemented for the purposes of this project, supporting the selection of hyperparameters of deep neural networks. The genetic algorithm developed for this project is limited by a maximum number of 10 iterations, and each population consists of 20 individuals, i.e. solutions. Each individual of the population is a set of values of the examined hyperparameters. The algorithm begins by randomizing the initial population of deep neural network architectures. On the basis of training of all models of a given population, the models are sorted according to the results of the achieved work efficiency. Then, the best 10 models are selected that will be involved in further processes of the genetic algorithm. The most effective models from the previous population from sequence number 2 to 10 are involved in the mutation process. It was decided that with a probability of 0.05 [25] each model would be able to be mutated with an equal probability of one of the 5 hyperparameters (in the case of classification) and one of the 4 hyperparameters (in the case of regression). The mutation process was carried out by drawing a new value of a given hyperparameter from the range constituting +/- 10% of the entire specified range of values in relation to the last value of a given hyperaparameter [25]. The value of 10% will allow a more detailed examination of the searched space, verify the effectiveness of the individual's operation, i.e. architecture with a modified gene value, and assess whether the introduced change was beneficial.

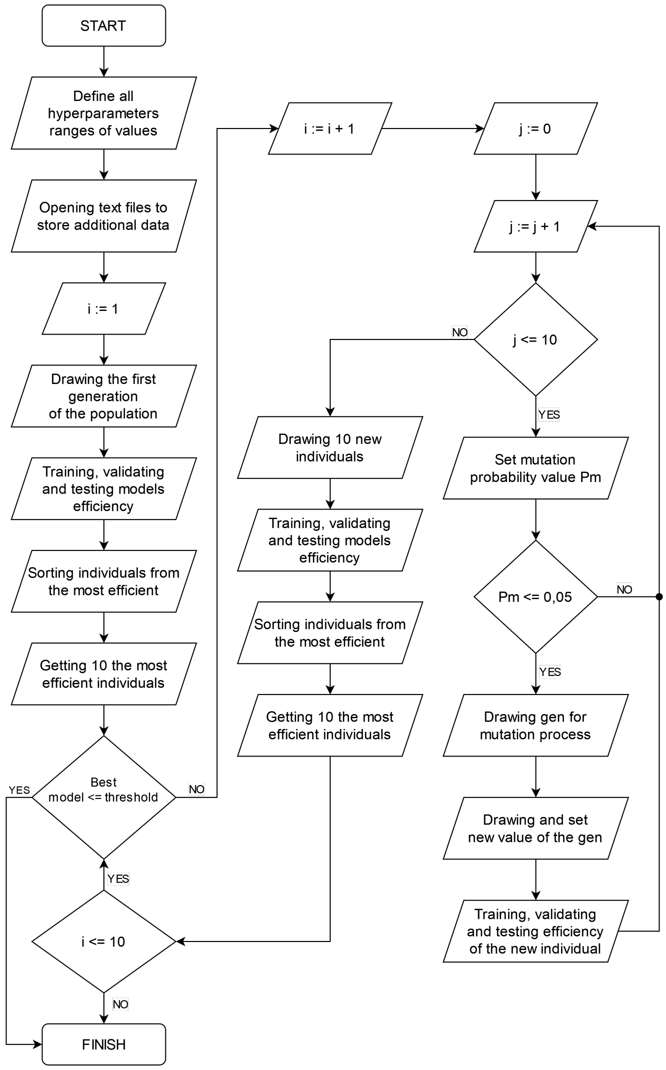

Fig. 3. Block diagram of used genetic algorithm

F. System architecture

The hierarchical approach requires the interconnection of individual neural networks, with the output of one network indicating which network will be used next. Fig. 4 shows a functional block diagram of the created neural network system.

Each of the deep neural networks that make up the hierarchical approach described above consists of an input layer, n hidden layers and an output layer with the number of output elements in the vector depending on the network model.

IV. SCENARIOS

The purpose of this project is to locate users inside and outside of the buildings. It is therefore necessary to carry out measurements in both of these environments. Each environment consists of several measurement scenarios.

Due to the lack of information on the locations of nearby WiFi access points and cellular networks base stations and information on propagation loss, which also are used for estimating UE locations [26], the method of creating a radio map with the use of radio signatures was selected. The localization process using Received Signal Strength method is divided into two phases: the offline phase and the online phase. In the offline phase, a radio map with reference points was created, while in the online phase the effectiveness of the tested localization algorithms was observed. For this reason, separate measurement scenarios for both of the mentioned phases were distinguished. Three mobile devices were held by users at a height of 100-120 cm above the ground in a gesture representing the use of the device.

In the offline phase measurements for the training set were taken for one minute at each reference point, with all of the measurement data sets being gathered every 0.5 seconds. The three users were in a static position - standing motionless with the UE in front of them. In order to test the designed radiolocation system in the online phase the testing route was covered by three users at a walking pace without stopping at individual reference points.

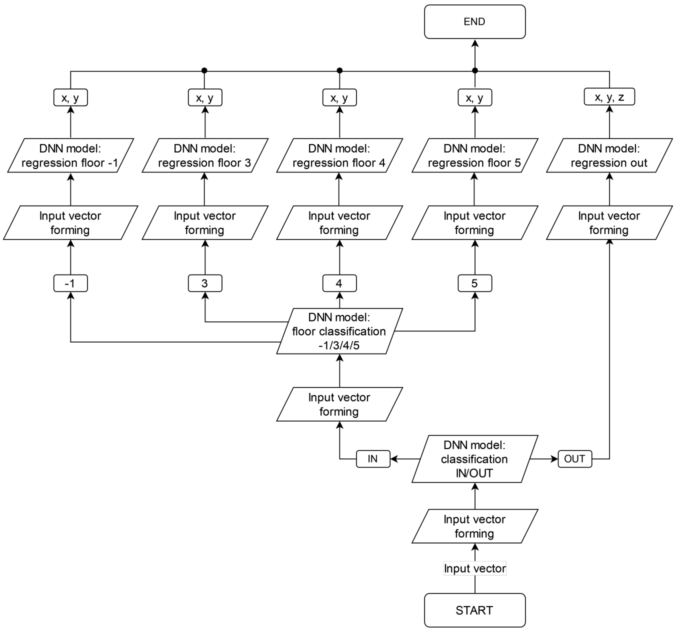

Fig. 4. Architecture of the designed radiolocation system based on a set/hierarchy(?) of DNN models.

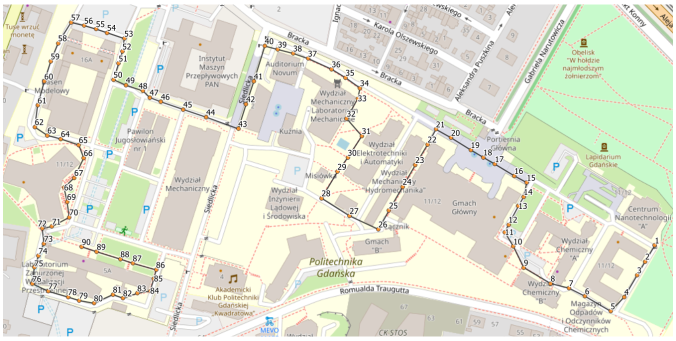

Fig. 5. Outdoor measurement scenarios.

A. Outdoor scenario

The studied environment in the described scenario is the area of the campus of the Gdańsk University of Technology, where a 1760 m long route was marked out. The starting point was located in the middle of the road between the Nanotechnology Center and building C of the Faculty of Chemistry in the easternmost area of the campus. The end point was located in front of the main entrance to the building A of the Faculty of Electronics, Telecommunications and Informatics. The measurement route runs through most of the campus area of the university due to the need to investigate the effectiveness of the localization algorithm in various propagation conditions proposed in the possible future works. It is also worth noting that the positions of LTE base stations and WiFi access points are not known.

In the offline phase a radio map was created. Fig. 5. presents the proposed online phase route (gray line) with 90 reference points (orange points) of the offline phase marked on it. The distance between nearest reference points vary from 10 to 30

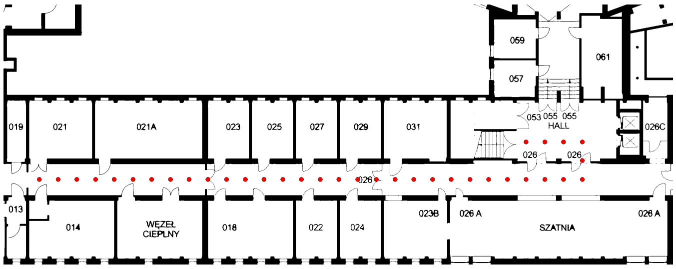

Fig. 6. Radio map for indoor scenario II A.

meters due to the need to conduct the measurement campaign as accurately as possible. RPs were situated in the so-called characteristic places. This resulted in decreasing the values of possible errors caused by the wrong mapping of the actual reference points positions on the map from which values such as latitude and longitude were gathered.

B. Indoor scenarios

All indoor scenarios measurements were carried out inside of the building A of the Faculty of Electronics, Telecommunications and Informatics (WETI A) of the Gdańsk University of Technology. A landmark (0, 0) has been established for each of the distinguished floors. Measurements from each of the floors were additionally marked with the floor number on which they were carried out. Similarly as in the outdoor scenario, there was information about the received signals from WiFi, GPS and cellular networks as well as additional user information in one set of measurement data.

The positions of the RPs in outdoor scenario have been set in characteristic places in order to reduce the error of mapping as much as possible. Setting a specific landmark (0,0) on each of the WETI A building's floors in indoor scenarios solves described problem. The research [27] on the impact of forming learning datasets on the effectiveness of deep learning in radiolocation applications shows that the positions of nearby reference points should be set with constant distance intervals. The RPs' grid distance was also examined in [30]. For this reason, a constant distance value between adjacent RPs of the created radio map was used. A systematic review of the literature shows that for the indoor environment the discussed value vary from about 1 m to 2 m [12], [14], [15], [28]. For the studied case the need to obtain a high UE position locations accuracy, it was decided to use the value of the discussed distance between nearby reference points equal to 1.5 m. This value will be the same for all of the indoor scenarios discussed below.

In the indoor environment, several measurement scenarios have been distinguished, depending on the measurement conditions and the location of measurement points. Scenario II A focuses on the environment with poor propagation properties - the basement of the WETI A building. This scenario has the highest number of reference points among the indoor scenarios. Next indoor scenarios are scenarios II B, II C and II D, where the offline phase RPs and the testing route of the online phase look the same. However, mentioned scenarios

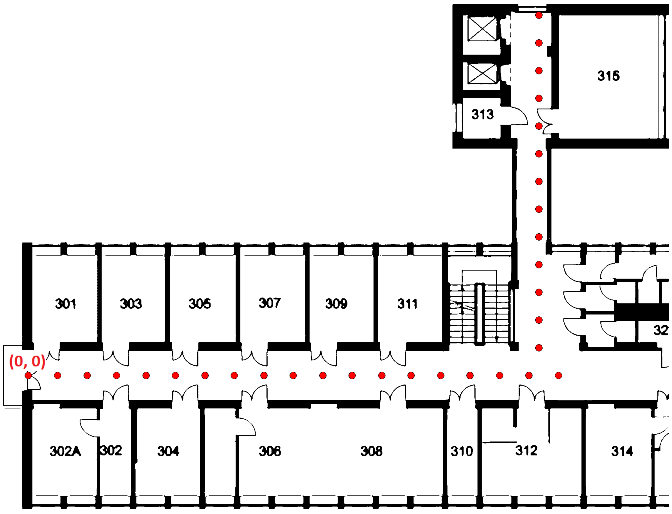

Fig. 7. Radio map for indoor scenarios II B, II C and II D.

differ by the floor number on which measurements were taken. In Fig. 6 and Fig. 7 radio maps of the offline phase are presented, respectively for the scenario carried out in the basement (II A) and the scenarios carried out on floors: 3, 4 and 5 (II B, II C, II D).

V. Results

This chapter of the paper is intended to present the results of the measurement application, genetic algorithm and deep neural networks in the designed radiolocation system. The wide scope of the works performed will allow to determine the capabilities of the prototype system in terms of time resolution and possible accuracy of users' location estimation in the described measurement scenarios. At the beginning, the ability of a mobile application to retrieve information about received signals via a mobile phone was presented. Then, the results of research on various structures of input vectors were presented, using the most effective architectures of neural network models obtained from the genetic algorithm.

A. Measurement application results

Before commencing the process of acquiring information about the radio signals received by the user's device for the designed radiolocation system, a series of test measurements was performed. Their purpose was to empirically confirm the choice of refreshing and recording time for radio signals. Table V summarizes the times between new information about radio signals that can be read by the Android application layer.

It was found that the average time of refreshing information about signals from the WiFi network is about 20 seconds (for the user using the default Android settings), and for the LTE network it is about 6 seconds. This means that the sufficient average time to create the input vector to the deep neural network will not be shorter than the average time to obtain new information about signals from the LTE or WiFi network.

Figure 8, 9 and 10 show the time dependencies between successive updates of information on received signals from

TABLE V
Time Results of Test Measurements for Outdoor Scenario.

WiFi scanning throttle limits turn on (default option)	
Min refreshing time	1.7s
Avg refreshing time	20s
Max refreshing time	89.4s
WiFi scanning throttle limits turn off	
Min refreshing time	1.1s
Avg refreshing time	2s
Max refreshing time	6.3s
LTE	
Min refreshing time	5s
Avg refreshing time	5.9s
Max refreshing time	27.7s
Selected time resolution of offline measurement phase	1s

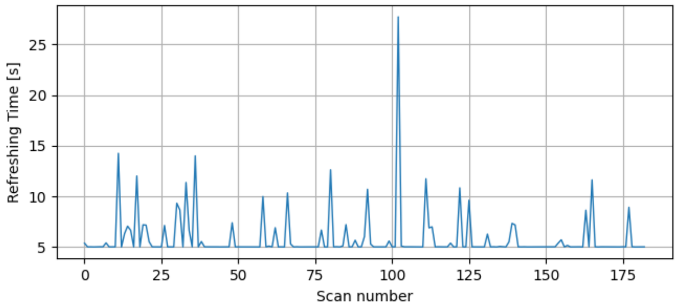

Fig. 8. The time between successive scans of information of received signals from the LTE cellular network.

the LTE network, WiFi network and the GPS system. It has been noticed that depending on the environment in which the measurements are performed, the time between reading the information about signals from the LTE network received by mobile phones available to the programmer is variable. This time for each scenario is greater than or equal to about 5 seconds. A dispersion of the refreshing time was noticed even on the adjacent floors of the WETI A building. Table V and Fig. 9 present the test results related to the limitations of the Android system. According to the documentation available for Android application developers, it follows that the limitations

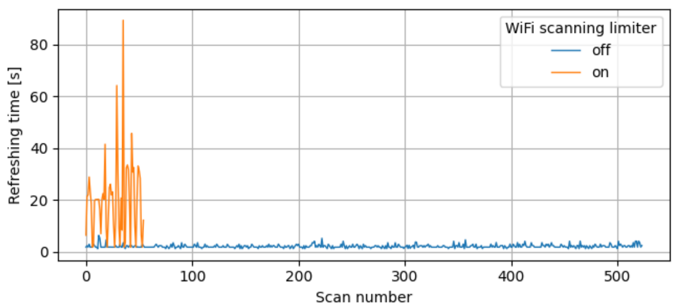

Fig. 9. The time between successive scans of information of received signals from the WiFI networks.

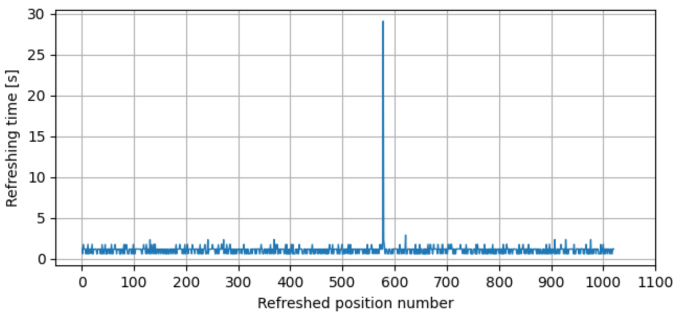

Fig. 10. The time between successive received user's location.

related to the increase in the interval between the available new information about signals from cellular networks and WiFi networks are dictated by the energy efficiency of mobile phones. Disabling the default setting in the developer options for limiting WiFi network scanning reduces the average waiting time (from about 20s to 2s) for a new WiFi network information scan made by mobile phones. The average time of updating the user's position from the GPS system that can be obtained by the measurement application is about 2s. Summing up the considerations - in the designed radiolocation system it is not possible to obtain a higher time resolution for receiving the position update than in the GPS location system.

In Fig. 11 is shown the discrepancy between the number of measurements received by user 1 and users 2 and 3. The number of measurements taken by user 1 is greater for the presented scenarios from about 27% to even about 50%. It should be mentioned here that the working time of the measurement application, and thus the acquisition time of measurement sets containing information about radio signals in each of the measurement scenarios, was the same for the three devices. Furthermore, the user device 1 has recorded a valid measurement of the RSSI parameter in each read measurement data set. It can be concluded that the implementation of a radiolocation system based on the radio signature method for devices with Android system depends not only on the brand and model of the device, which was presented in the theoretical introduction in this documentation, but also on the version of the Android system on a given device. It has been confirmed that the use of identical methods provided in different versions of the API may cause a different result, which may directly translate into the effectiveness of the prototype radiolocation system due to the missing actual values in the input vector of the neural network.

B. Statistical Data of the Received Radio Signals

1) Most common number of signal address appearences in input vector: Table VI shows most frequently occurrent numbers of BSSID and PCI addresses in the input vectors of the basement, 3rd floor, 4th floor, 5th floor and outdoor measurement scenarios, respectively. It was observed that in the case of WiFi networks, the most frequent number of BSSID addresses in the input vectors is 16. Only in the indoor scenario of the basement floor, the input vectors most often consisted of 14 BSSID addresses. In the case of cellular networks, the number of PCI addresses most frequently found in the input data vector is not so clear-cut. In the input vectors of the outdoor scenario, 1 PCI address was the most common, in the data obtained from the indoor scenarios on the basement floor and the 3rd floor most often there were 2 PCI addresses, while on the 4th and 5th floors, the most frequent number of PCI addresses was the number 3. Statistics of the frequency of occurrence of a given number PCI and BSSID addresses in the input vectors of deep neural networks are important from the perspective of the relationship between the selection of their appropriate combination and the effectiveness of deep neural networks.

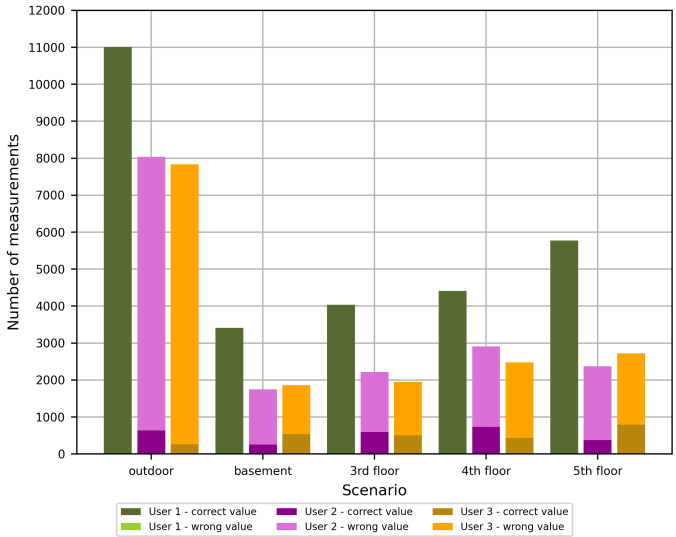

Fig. 11. Correct and incorrect measurements of the RSSI of the LTE network by mobile phones.

TABLE VI
MOST COMMON NUMBER OF LTE AND WIFI ADDRESSES' APPEARANCES IN INPUT VECTORS.

	PCI	BSSID
Indoor basement	2	14
Indoor 3rd floor	2	16
Indoor 4th floor	3	16
Indoor 5th floor	3	16
Outdoor	1	16

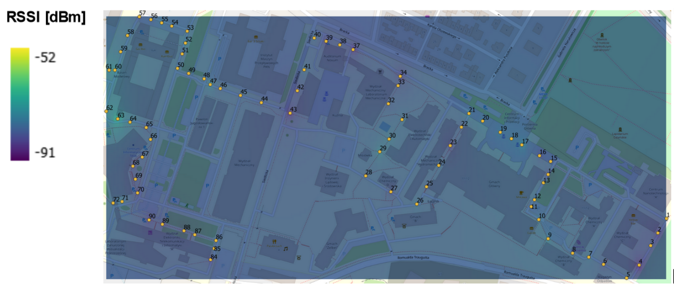

Fig. 12. Interpolated average RSSI power distribution for one user and one BSSID address

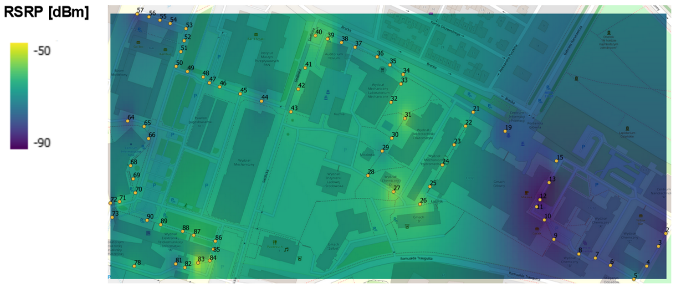

Fig. 13. Interpolated average RSRP power distribution for one user and one PCI address

2) Interpolated power distribution of the received signals: In order to present the maps of the interpolated power distribution of the received signals, it was decided to select the data obtained for one BSSID / PCI address by one user. The set of data needed to visualize the signal power distribution in the outdoor scenario was selected on the basis of the largest number of occurrences of a given address of downloaded data for one user and the greatest possible number of its occurrences at various measurement points. The figures 12 and 13 show an example of the distribution of average RSSI and RSRP signal strengths for data sets for one user and one BSSID and PCI address.

From the visualization of interpolated power distribution maps, it can be observed that the values of the measured powers differ significantly depending on the place of measurement. The strength of the received signals may be influenced by factors such as the topography, line of sight (or lack of it) and the user's distance from base stations and access points. To create the input vector of the deep neural network data, the number and the power of measured signals at a given reference point is also important. As a result, the received signals of less power may affect the effectiveness of deep neural networks.

C. Genetic algorithm results

The following tables VII and VIII show the results of the best architectures found after 10 iterations of the genetic algorithm for each measurement scenario. To test the effectiveness of the neural network models, the input vector structure was used, consisting of information from ten WiFi access points and information from three eNBs of the LTE network.

TABLE VII
RESULTS OF GENETIC ALGORITHM FOR CLASSIFICATION.

	Indoor-Outdoor Classification	Floor Classification
Iteration of Genetic Algorithm	2	6
Number of Hidden Layers	3	5
Number of Nodes	355	250
Batch size	256	128
Learning Rate	0.001	0.0005
Training Acc	99.9%	99.3%
Validation Acc	99.7%	97.1%
Testing Acc	99.6%	96.7%

The efficiency of indoor and outdoor environment classification is higher than the indoor and outdoor floor classification by approximately 3%. This may be due to the fact that in the first case one of the two possible classes is predicted, and in the second one is predicted from one of the four possible classes.

TABLE VIII
RESULTS OF GENETIC ALGORITHM FOR REGRESSION.

	Deep Indoor	3rd Floor	4th Floor	5th Floor	Outdoor
Iteration of Genetic Algorithm	1	7	10	9	2
Number of Hidden Layers	2	5	3	3	4
Number of Nodes	311	213	247	239	169
Batch size	64	32	32	128	32
Learning Rate	0.001	0.001	0.001	0.001	0.0005
Training RMSE	0.5m	0.3m	0.4m	0.4m	6.8m
Validation RMSE	2m	0.7m	1m	1m	15.9m
Testing RMSE	1.5m	0.7m	1.1m	1.1m	15m

The accuracy of the estimation of the location of users on the floors does not differ significantly (0.3 m between the 3rd and 4th and between the 3rd and 5th floors), and the greatest similarity of the obtained efficiency results occurs between the 4th and 5th floor. This may be the result of differences in the training data delivered to deep neural networks. The largest RMSE error in the user location estimate was obtained for the outdoor scenario and it was 15m for a network structure with 4 hidden layers and 169 nodes in each layer. In general, the structures of the best architectures for the regression problem are similar and in future work it would be necessary to verify the effectiveness of the selected architecture in all tested measurement scenarios.

D. Radiolocation system results

On the basis of the most effective architectures selected from the previous chapter, in this article the authors examined the influence of the content and structure of input vectors of deep neural networks. Figures 14, 15 and 16 show the results obtained by the best models trained on various WIFInLTEm input vector configurations, where n - the number of measurements from the WIFI network, m - the number of measurements from the LTE network in the input vector.

1) Classification error: In the Fig. 14 the results of classification accuracy are presented for the indoor and outdoor classification as well as the floor classification. The highest classification accuracy for both scenarios were obtained using models trained on the vectors consisting of a combination of WIFI and LTE measurements. The indoor and outdoor classification scenario has higher accuracy than the floor classification of the indoor scenario.

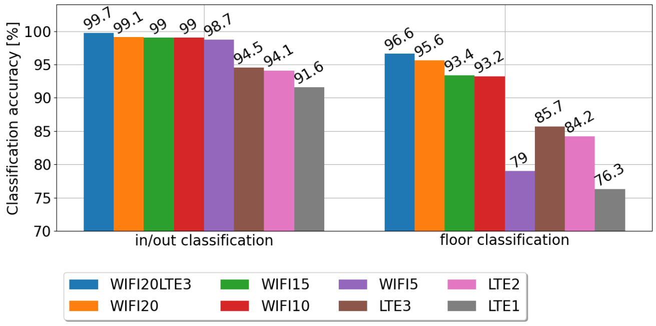

Fig. 14. Accuracy of classification of indoor and outdoor scenarios for the most effective architecture for various input vector structures.

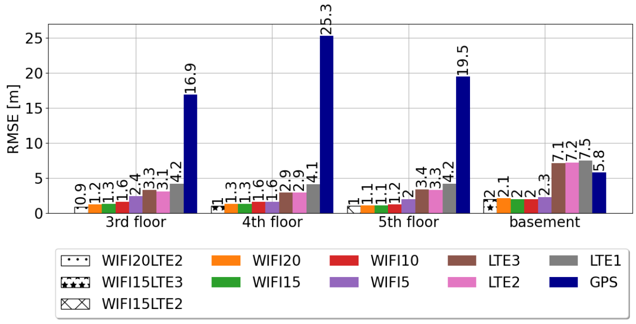

Fig. 15. RMSE error of user locations in the indoor scenarios for the most effective architectures for various input vector structures.

2) Positioning error: In the Fig. 15, 16 the results of RMSE error of user location in the indoor and oudoor scenarios are presented.

The results obtained by the models trained on the vectors consisting of WIFI measurements or a combination of WIFI and LTE measurements present the smallest RMSE location errors for regression. For the indoor regression on floors 3rd, 4th, 5th, basement and the outdoor regression the differences between the largest and smallest RMSE location error obtained by the given vectors are 1.5 m, 0.6 m, 1 m, 0.3 m and 19.1 m, respectively. The smallest location RMSE errors for each scenario were obtained using models trained on vectors consisting of a combination of WIFI and LTE measurements. The lowest RMSE error of 0.9 m of user's location was obtained for the indoor scenario on the third floor of the WETI A building.

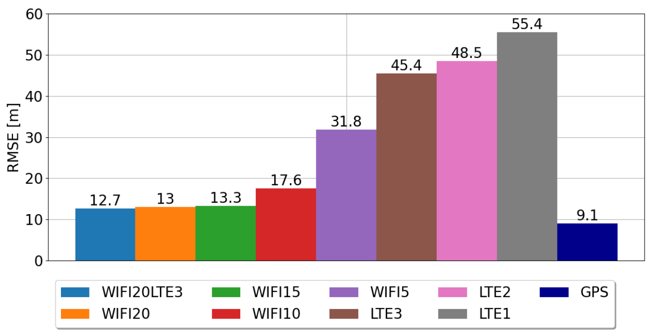

Fig. 16. RMSE error of user locations in the indoor scenarios for the most effective architectures for various input vector structures.

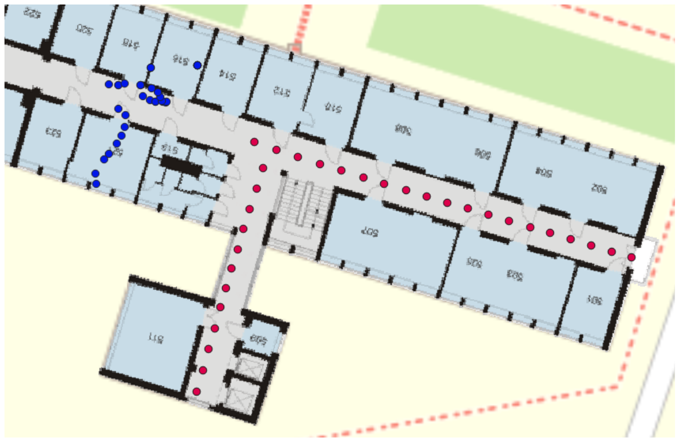

Fig. 17. Map of GPS route points

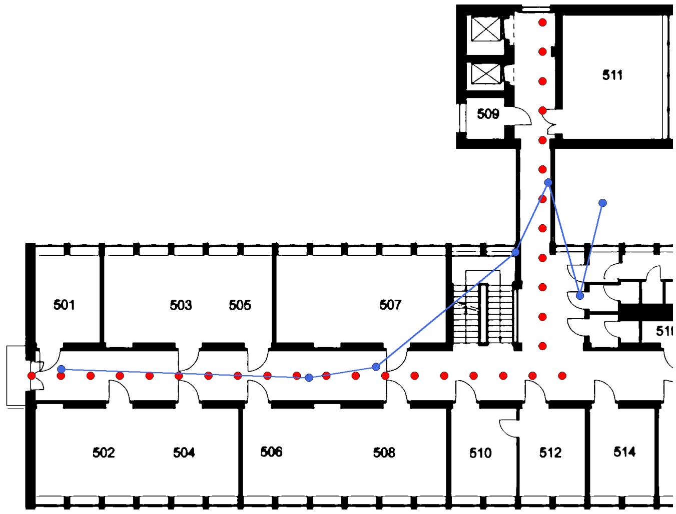

Fig. 18. Map of route points estimated by deep neural network models for the 'LTE2' input vector

From the visualization of the users' location estimates by the deep learning algorithm presented in Figure 17, 18, 19 and 20 it can be observed that the number of determined position estimates is smaller than the number of positions determined by the GPS system. It can be seen that the density of points representing the estimated location of users is different in relation to the input vectors containing separately information about signals from these networks. This results in the advantage of the classic GPS localization system over the method developed by the authors of this documentation, based on radio signals from WiFi and LTE networks. In the case of the GPS system, the assumption described in the theoretical introduction is confirmed that the GPS system provides greater time resolution. Based on the table, it can be concluded that the signals of the hybrid WiFi and LTE networks significantly

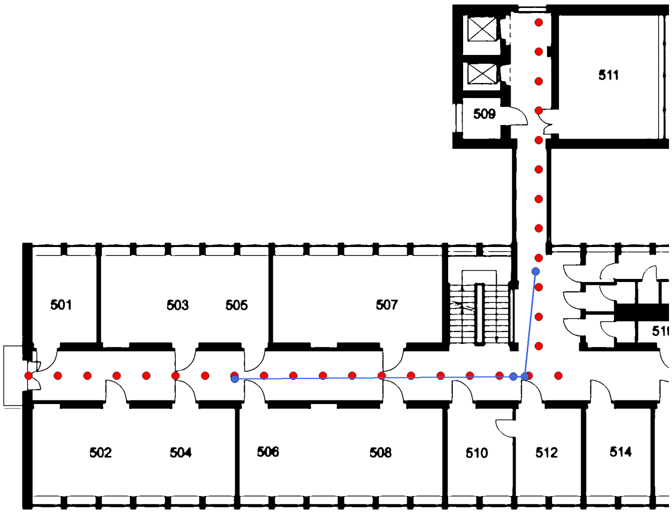

Fig. 19. Map of route points estimated by deep neural network models for the 'WiFI15' input vector

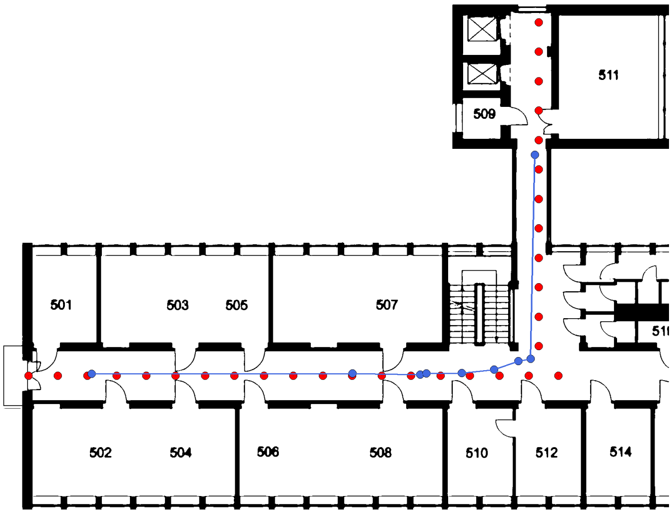

Fig. 20. Map of route points estimated by deep neural network models for the 'WIFI15LTE2' input vector

contribute to increasing the accuracy of the users' location estimate. The presented results of the location accuracy confirm the superiority of the proprietary radiolocation system over the GPS system in the indoor scenario. The difference in accuracy in the indoor scenario is from 3.8 meters to 16 meters, which is a significant increase in the accuracy of the location compared to the GPS system. Only in the outdoor scenario, the GPS system works more efficiently, achieving a 3.6 meters lower average location error.

VI. Conclusion

In this paper radiolocation system prototype based on signals of opportunity with the use of deep neural networks was presented. The superiority of the proposed system over GPS has been demonstrated. It has been shown that the use of input vectors consisting of the parameters of WiFi and LTE signals gives greater accuracy of users' location estimation than those consisting of only WiFi parameters or only LTE parameters. On the basis of the obtained results, it can be concluded that:

- the prototype of the proprietary radiolocation system based on the measurements of radio signals is characterized by a lower time resolution compared to the classic GPS location system
- the correct functioning of the prototype radiolocation system largely depends on the user's mobile phone and the version of the Android system,
- the most effective solutions - architectures of deep neural networks obtained by the genetic algorithm - do not differ from each other in terms of the obtained accuracy of system, and the very range of hyperparameters for which similar values of effectiveness are obtained is wide,
- the use of a combination of measurements from WiFi and LTE networks allows for a more precise location of users in a radiolocation system based on the radio signature method compared to using measurements only from the WiFi network or only from the LTE network,
- user's locations determined by the designed radiolocation system, using the deep learning algorithm, are burdened with a lower RMSE error than the locations of users designated by the GPS system in an indoor environment.

Future works will be focused on searching for the most effective combination of information of received signals by mobilne phones with the use of filtration. Additionally, due to the possibility of communication loss with the LTE network, developed radiolocation system should also retrieve and use information from 3G and 2G network to estimate user position.

References

[1] R. Khullar and Z. Dong, "Indoor localization framework with WiFi fingerprinting," 2017 26th Wireless and Optical Communication Conference (WOCC), 2017, pp. 1-6, doi: 10.1109/WOCC.2017.7928970.

[2] Kim Geok, Tan; Zar Aung, Khaing; Sandar Aung, Moe; Thu Soe, Min; Abdaziz, Azlan; Pao Liew, Chia; Hossain, Ferdous; Tso, Chih P.; Yong, Wong H. 2021. "Review of Indoor Positioning: Radio Wave Technology" Appl. Sci. 11, no. 1: 279. https://doi.org/10.3390/app11010279

[3] Y. Li, F. Shu, B. Shi, X. Cheng, Y. Song and J. Wang, "Enhanced RSS-based UAV Localization via Trajectory and Multi-base Stations," in IEEE Communications Letters, doi: 10.1109/LCOMM.2021.3061104.

[4] S. Han, T. Kang and J. Seo, "Smartphone Application to Estimate Distances from LTE Base Stations Based on Received Signal Strength Measurements," 2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), 2019, pp. 1-3, doi: 10.1109/ITC-CSCC.2019.8793365.

[5] A. Mathisen, S. Krogh Sørensen, A. Stisen, H. Blunck and K. Grønbæk, "A comparative analysis of Indoor WiFi Positioning at a large building complex," 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2016, pp. 1-8, doi: 10.1109/IPIN.2016.7743666.

[6] Xia, Shixiong; Liu, Yi; Yuan, Guan; Zhu, Mingjun; Wang, Zhaohui. 2017. "Indoor Fingerprint Positioning Based on Wi-Fi: An Overview" ISPRS Int. J. Geo-Inf. 6, no. 5: 135. https://doi.org/10.3390/ijgi6050135

[7] A. H. Salamah, M. Tamazin, M. A. Sharkas and M. Khedr, "An enhanced WiFi indoor localization system based on machine learning," 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2016, pp. 1-8, doi: 10.1109/IPIN.2016.7743586.

[8] C. Zhou, J. Liu, M. Sheng and J. Li, "Hybrid RSS/CSI Fingerprint Aided Indoor Localization: A Deep Learning based Approach," GLOBECOM 2020 - 2020 IEEE Global Communications Conference, 2020, pp. 1-6, doi: 10.1109/GLOBECOM42002.2020.9348190.

[9] B. S. Ciftler, A. Albaseer, N. Lasla and M. Abdallah, "Federated Learning for RSS Fingerprint-based Localization: A Privacy-Preserving Crowdsourcing Method", 2020 International Wireless Communications and Mobile Computing (IWCMC), 2020, pp. 2112-2117, doi: 10.1109/IWCMC48107.2020.9148111.

[10] X. Wang, X. Wang, S. Mao, J. Zhang, S. C. G. Periaswamy and J. Patton, "Indoor Radio Map Construction and Localization With Deep Gaussian Processes," in IEEE Internet of Things Journal, vol. 7, no. 11, pp. 11238-11249, Nov. 2020, doi: 10.1109/JIOT.2020.2996564.

[11] C. Zhou and A. Wieser, "Application of backpropagation neural networks to both stages of fingerprinting based WIPS," 2016 Fourth International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services (UPINLBS), 2016, pp. 207-217, doi: 10.1109/UPINLBS.2016.7809973.

[12] C. Hsieh, J. Chen and B. Nien, "Deep Learning-Based Indoor Localization Using Received Signal Strength and Channel State Information," in IEEE Access, vol. 7, pp. 33256-33267, 2019, doi: 10.1109/ACCESS.2019.2903487.

[13] G. B. Tarekegn, R. T. Juang, H. P. Lin, A. B. Adege and Y. Y. Munaye, "DFOPS: Deep-Learning-Based Fingerprinting Outdoor Positioning Scheme in Hybrid Networks," in IEEE Internet of Things Journal, vol. 8, no. 5, pp. 3717-3729, 1 March1, 2021, doi: 10.1109/JIOT.2020.3024845.

[14] W. Lin, C. Huang, N. Duc and H. Manh, "Wi-Fi Indoor Localization based on Multi-Task Deep Learning," 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), 2018, pp. 1-5, doi: 10.1109/ICDSP.2018.8631868.

[15] J. Zou, X. Guo, L. Li, S. Zhu and X. Feng, "Deep Regression Model for Received Signal Strength based WiFi Localization," 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), 2018, pp. 1-4, doi: 10.1109/ICDSP.2018.8631593.

[16] A. Poulose and D. S. Han, "Indoor Localization using PDR with Wi-Fi Weighted Path Loss Algorithm," 2019 International Conference on Information and Communication Technology Convergence (ICTC), 2019, pp. 689-693, doi: 10.1109/ICTC46691.2019.8939753.

[17] R. H. Jaafar and S. S. Saab, "A Neural Network Approach for Indoor Fingerprinting-Based Localization," 2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2018, pp. 537-542, doi: 10.1109/UEMCON.2018.8796646.

[18] N. Hernández, M. Ocaña, J. M. Alonso and E. Kim, "WiFi-based indoor localization and tracking of a moving device," 2014 Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), 2014, pp. 281-289, doi: 10.1109/UPINLBS.2014.7033738.

[19] H. Xujian and W. Hao, "WIFI Indoor Positioning Algorithm Based on Improved Kalman Filtering," 2016 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), 2016, pp. 349-352, doi: 10.1109/ICITBS.2016.83.

[20] K. Y. Seok and J. H. Lee, "Deep Learning Based Fingerprinting Scheme for Wireless Positioning," 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), 2020, pp. 312-314, doi: 10.1109/ICAIIC48513.2020.9065054.

[21] R. H. Jaafar and S. S. Saab, "A Neural Network Approach for Indoor Fingerprinting-Based Localization," 2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2018, pp. 537-542, doi: 10.1109/UEMCON.2018.8796646.

[22] H. Rizk, M. Torki and M. Youssef, "CellinDeep: Robust and Accurate Cellular-Based Indoor Localization via Deep Learning," in IEEE Sensors Journal, vol. 19, no. 6, pp. 2305-2312, 15 March15, 2019, doi: 10.1109/JSEN.2018.2885958.

[23] https://developer.android.com/reference

[24] I. Goodfellow, Y. Bengio and A. Courville, "Deep Learning", Massachusetts Institute of Technology, The MIT Press Cambridge, 2017

[25] S. Urwan, "Hyperparameters Selection of the Artificial Intelligence Used in the Radiolocalization System", Gdańsk 2020

[26] H. Bae, J. Oh, K. Lee and J. Oh, "Low-cost indoor positioning system using BLE (bluetooth low energy) based sensor fusion with constrained extended Kalman Filter," 2016 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2016, pp. 939-945, doi: 10.1109/ROBIO.2016.7866445.

[27] D. R. Wysocka, "Research and Analysis of Methods for Forming Learning Datasets and Their Impact on the Effectiveness of the Artificial Intelligence for Radiolocation Applications" Gdańsk 2020

[28] Z. Ma, S. Poslad, J. Bigham, X. Zhang and L. Men, "A BLE RSSI ranking based indoor positioning system for generic smartphones," 2017 Wireless Telecommunications Symposium (WTS), 2017, pp. 1-8, doi: 10.1109/WTS.2017.7943542.

[29] Wonjae Lee and Hak-Young Kim, "Genetic algorithm implementation in Python," Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05), 2005, pp. 8-11, doi: 10.1109/ICIS.2005.69.

[30] Gansemer, S. & Pueschel, S. & Frackowiak, R. & Hakobyan, S. & Grossmann, Uwe. (2010). Improved RSSI-based Euclidean distance positioning algorithm for large and dynamic WLAN environments. Computing. 9. 37-44.