Intelligent Home Systems for Ubiquitous User Support by Using Neural Networks and Rule-Based Approach

Marcin Woźniak and Dawid Połap

***Abstract*—Artificial intelligence methods applied in smart home environments can give a ubiquitous support of people, provide automatic control of system settings to lower the costs of operation, improve energetic efficiency, and longer endurance of components. In this article, we discuss the use of neural networks and rule-based systems as the components of automatic control over house elements. In the proposed framework, we store the knowledge and teach the system about the home in parallel to operation. Initially, the system starts with global settings, however, data are collected during use and parallel to this regular training is run so when the new knowledge guarantees higher efficiency, the system switch to use it. We have developed a new neural-based mechanism with rules control method to lower the costs of operation while keeping the needs of users. The components were tested and discussed due to practical application in our everyday life.**

***Index Terms*—Neural networks, rule-based systems, smart environments.**

I. Introduction

The concept of "smart" is increasingly appearing in a variety of technological goods, which are intended to streamline and improve our lives. Special products of this type are technologies based on the KNX standard for electrical installation, which allows communication between all electrical components throughout the home. What is more, it is possible to connect external elements to this type of installation. Examples are a mobile phone or a portable computer that can be used as elements which bring together all the appliances in the home using an external controller in the KNX standard. That kind of smart environment would be very helpful, especially for people with disabilities [1]. Another big advantage is efficiency, since such solutions would be ecological due to the possibility of various energy saving options (e.g., by turning OFF unnecessary devices or lights). Now, we need artificial intelligence methods to support control within these frameworks.

Manuscript received June 19, 2019; revised August 10, 2019; accepted September 2, 2019. Date of publication November 4, 2019; date of current version January 17, 2020. This work was supported by the Rector of the Silesian University of Technology under Proquality Grant 2019. Paper no. TII-19-2571. *(Corresponding author: Marcin Wozniak.)*

The authors are with the Faculty of Applied Mathematics, Silesian University of Technology, 44-100 Gliwice, Poland (e-mail: marcin.wozniak@polsl.pl; dawid.polap@polsl.pl).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2019.2951089

The KNX is the main element of communication standard between objects within the Internet of Things (IoT). Through this type of connection, it is possible to retrieve data from various sensors, process it together in a detailed analysis, and return the information in a specific form. However as for everything, constant changes and adaptation to new technologies require further research. The smart concept in which we can describe our homes is primarily involving new methods of artificial intelligence (AI). In [2], the authors presented detection of unsupervised visits when the owners are not at homes. The whole idea is based on information sent to the mobile phones when unexpected activity is detected by analyzing readings from various sensors located throughout the building. While in [3], the idea of using detection sensors was described as a foundation for security monitoring. In [4], the communication protocol for IoT has been described. The use of motion detectors can be used not only in the area of security, but also for location of people and things in rooms [5], as well as the analysis of their movement path [6] or intention recognition [7]. "Smart" technologies also contribute to energy saving by reducing unnecessary excesses, what concludes in minimizing bills. Joo and Choi presented a framework for energy management in multiple smart homes by the analysis of the use of dispersed energy sources. Similarly, in [9], predictive control was introduced to manage a small building elements. Smart control of energy resources can be done via AI methods [10] or intelligent frameworks for the balanced control of electricity use. Another issue is sound analysis. Progress in this direction was shown in [11]. The authors used acoustic sensors to analyze sounds and control silency aspects by proposed implementations for the integrated management system. Another area is information flow between different equipments that exchange information about conditions in the interaction mode [12] or static coordinator control mode [13]. Devoted IoT data systems are also very efficient in industrial site [14]. In fact, still we have many unresolved and open questions for AI learning systems to be used for the improvement in the quality of our lives [15]. An interesting fact is that most AI solutions are based on machine learning techniques, as shown in [16], advantages and disadvantages of such implementations open important perspectives for further development.

In this article, we propose the idea of the AI framework that collects the data in a real time and uses new controllers based on the neural network decision composed with rule system as presented in Fig. 1. The construction of the proposed AI

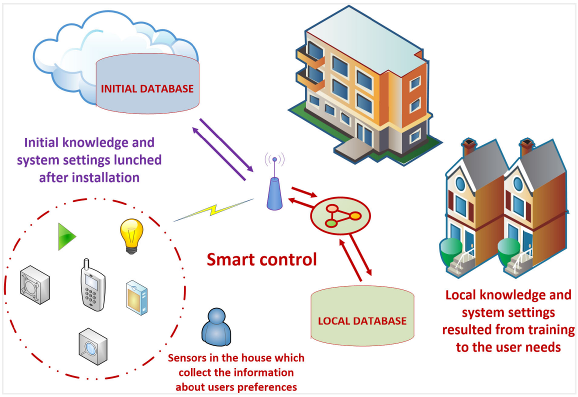


Fig. 1. Sample idea of the system, where initial data and settings are launched after installation to run the system, and during operation, the system learns about the needs of the users.

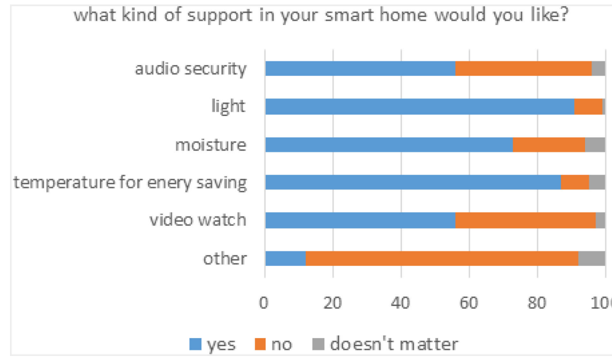


Fig. 2. Results of questions sent to potential users of smart environments. Before research, we have asked 100 students at our department to answer which options of home environment control would be best for them.

controllers has been described in details and tested. The novelty of our approach is in presented construction of ubiquitous smart control. Initially, when the system is newly installed and there is no information about any current actions, the global knowledge is used to control the house. During time, when people start to use facilities in the house and adjust the systems for their needs the data from sensors is collected. The system is trained to achieve the highest possible efficiency in these conditions and when precision is better, we change to use the revised knowledge. Newly proposed AI techniques support constant data retrieval and precise decision making in all conditions. In the following sections, we present and discuss the proposed AI decision model, advantages of our idea, and show the possibility of implementing our technology for everyday use. The initial research show high potential and many possibilities for further improvements.

II. Proposed IoT Operating Model Architecture

Architecture based on AI methods needs to be trained with a certain database called knowledge. In the idea, the client buys a house with sensors and software. However the newly installed system will not have enough knowledge about family preferences, so it will not be useful. We solve this problem in a way that makes it possible to use the system from the very beginning.

The solution is to modify the database (or use two in parallel), for which we can model a composition of the proposed AI methods. We designed the system in which one control model would be trained with all the data obtained from all users (and therefore, from all the houses participating in the program), and the other one would be trained using real-time data. This solution requires a devoted module for downloading and transferring the data, second, there will be certain technical requirements for these devices that cooperate in the system and the main controller to operate the entire system.

A. Framework Architecture

Before modeling architecture of a smart home, we have asked 100 students at our department to answer some questions about their ideas of ubiquitous support at home. We asked them what kind of support would they expect. Propositions from students were evaluated and presented in Fig. 2. We were asking about typical ideas like light, humidity, or moisture control together with optimal temperature setting for lover energy costs, but we have also asked about automatic video or audio security systems at home. In most cases, positive answers reached above 50%, which showed that our intentions for this type of support were correct.

For the system construction, we assume that we have different sensors installed in each room. Examples of sensors may be a motion sensor, a moisture meter, a thermometer, a security microphone, and a light intensity meter, etc., as presented in Fig. 3. Each sensor returns some raw data to the systems. Such information should be processed. Unfortunately, the data exist in a variety of formats. For numerical values such as temperature measurement, no additional processing is required. In case of sound, image, or video—processing is a basic requirement. More precisely, only the most important elements from multimedia files should be extracted in order to obtain the most important features. Take the example of monitoring using video cameras—the file is composed of frames that are images. The system must extract only what has changed from previous frames to decide on the situation in the home.

Of course, applying different feature extraction methods could be very helpful, but there is a problem with enormous hardware load that can be caused due to many file types processed by a variety of methods. Let us notice that data is obtained constantly. For this purpose, we suggest using one method for sound and image security described later. Obtained features should be placed in the local database. Having processed the data, it should be statistically modeled to describe the action in the home. On the other hand, to manage energy, we do not need the data from the sound. Therefore, in the main controller, we need to manage results by giving certain weights for each data type when loading and downloading the knowledge from the database. In the proposed solution, downloading the data with given weights is interpreted by the classifier, which returns specific decisions about next actions. In our case, we propose a neural network

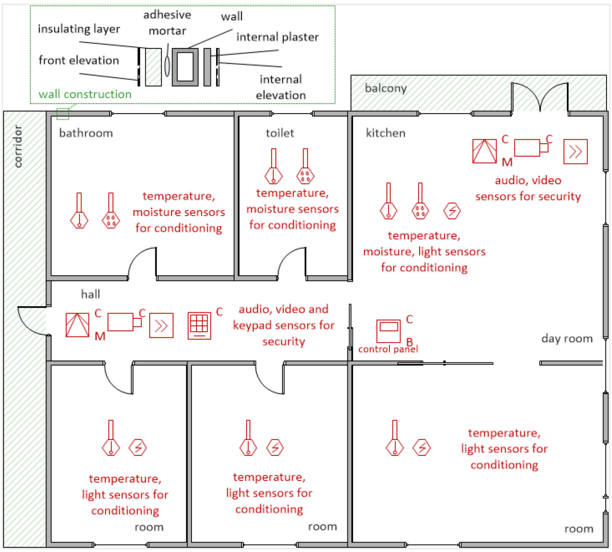


Fig. 3. Plan of the home with locations of sensors placed to control security aspects (motion detectors, sound, video control, and alarm coding system); sensors placed to maintain the best temperature and moisture conditions in the bathroom, toilet, and kitchen; and sensors placed to maintain the optimal temperature and lighting of the living rooms. The small schema shows construction of the wall insulation that prevents sudden temperature changes during winter and summer seasons.

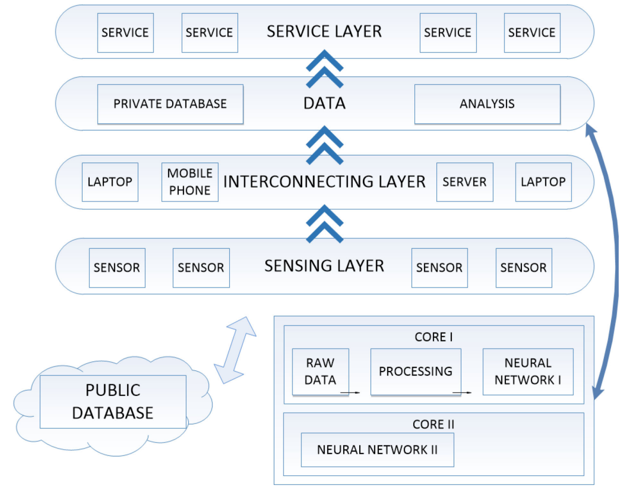


Fig. 4. Information flow in the proposed framework. The data run from sensors collecting the data about user's needs and forward this information over the network to train the local controller for the most efficient actions.

system as a classifier. Such classifier returns a value string, which needs to be analyzed for the incoming parameters and features. The analysis consists of calculating the differences between the obtained decision and the current state of all sensors. If there is a significant change, it means that the situation should be estimated by reducing or enlarging parameters.

From a technical point of view, the information flow at the home can be done using KNX. As mentioned before, when a user purchases a system for a new house, the classifier will not be ready for action due to lack of the data. To prevent this problem, we suggest using two databases. One—local—where collected data for a specific house will be kept. The second one—global—which will be placed in the external cloud, to which the obtained data will be added. This composition makes it possible to train the classifier even if we do not have a large amount of the data from the current system setting. This solution seems good, but it has one disadvantage—training is time consuming. Hence, one neural network should be put within the cloud. This global controller will learn and save the best configuration of the neural classifier at any time. In our solution to maintain ability of the smart control at any time, during the installation of the system, not only the initial data but also ready, exemplary neural network configuration will be downloaded. After a time, when the local database would be filled with new information, on the second core (the local server), a neural network training process will start. After reducing classification error, the newly trained devoted neural structure will replace the initial configuration. The sample visualization of these processes is presented in Fig. 4.

III. AI-Based Information Processing Model

The data from sensors need to be modified in such a way that the proposed classifier could use them. In practice, all the data must be defined in a numerical way. In case of different sensors (such as a thermometer), it is not a problem. However for sound and image preprocessing, we need to extract information before processing. To use one method for both files, one of them should be presented in the form of the other one. The easiest way is to define a sound in a graphical form. At first, let us define a mathematical form of the audio signal s at time interval t as

$$s(t) = \sum_{i=1}^{N} A_i(t) \sin[2\pi F_i(t)t + \omega_i(t)] \tag{1}$$

where amplitude is marked as $A(t)$, the frequency as $F(t)$, and a phase as $\omega(t)$. To present audio as image, we must convert this signal using the short-time Fourier transform [17] as

$$S\{s[n]\}(m, f) = \sum_{n=-\infty}^{\infty} s[n]w[n-m]\exp(-jfn). \tag{2}$$

The aforementioned formula can be used to calculate a graph of energy density estimation. To do that, we create the graph in two dimension (a spectrogram is a flatten version of the original sound wave). For each point (x, y) (x is time, y is frequency), we define intensity, which value is visible as a shade of color. The formula for the graph is defined as

$$\text{spectrogram}\{s(t)\}(t, f) \equiv |S(t, f)|^2. \tag{3}$$

Now, audio and image are in graphical form. Therefore, we can use one method to find important elements from these pictures. We find key-points using the *speeded up robust features (SURF)* algorithm [18]. For each received pixel position, we calculate differences in color between two frames. The results are stored in the vector used by the classifier.

To find interesting points, the SURF algorithm uses Hessian describing local changes around a given pixel according to

$$H(x,\omega) = \begin{bmatrix} L_{xx}(x,\omega) & L_{xy}(x,\omega) \\ L_{xy}(x,\omega) & L_{yy}(x,\omega) \end{bmatrix} \tag{4}$$

where $L_{xx}(x,\omega)$ is convulsion of the second derivative of the Gaussian function with an integral image I calculated as

$$L_{xx}(x,\omega) = I(x)\frac{\partial^2}{\partial x^2}g(\omega) \tag{5}$$

$$L_{yy}(x,\omega) = I(x)\frac{\partial^2}{\partial y^2}g(\omega) \tag{6}$$

$$L_{xy}(x,\omega) = I(x)\frac{\partial^2}{\partial xy}g(\omega) \tag{7}$$

where $g(\omega)$ is a kernel.

Let x be the point that has the sum of all values of the neighboring points presented as

$$I(x) = \sum_{i=0}^{i\leq x}\sum_{j=0}^{j\leq y} I(x,y). \tag{8}$$

The extrema (suspected of being important points) are found and SURF descriptor based on the Haar wavelet is calculated

$$\det(H_{\text{approximate}}) = D_{xx}D_{xy} - (wD_{xy})^2 \tag{9}$$

where w is the weight and $D_{xx} = L_{xx}(x,\omega)$.

In this way, we have a set of k points $\{(x_0,y_0)^n, (x_1,y_1)^n, \ldots, (x_k,y_k)^n)\}$ for n-th frame. We take basis frame (a pattern of user voice) and compare calculated pixel values in the same positions $\{(x_0,y_0)^{n+1}, (x_1,y_1)^{n+1}, \ldots, (x_k,y_k)^{n+1})\}$ by counting the difference on each color components (using the RGB model, which refers to *Red*, *Green*, and *Blue*) as

$$F(x_i^n, y_j^n) - F(x_i^{n+1}, y_i^{n+1})$$

where F is one of the RGB components. Of course, the acquisition of key points does not allow to define the content of the sound image, so it is worth converting each of these points into a feature map. We define the found key point as the center of the feature area. For each point, we take the neighborhood of the size 50×50, which should be modified in order to acquire features describing the area. Each of these areas is simplified relative to the average value of smaller neighbors. For each pixel at position (x,y), the neighborhood in relation to the length of $k \in \{3,4,5\}$ is calculated as

$$\frac{1}{3}\sum_{i=1}^{k}\sum_{j=1}^{k}(p((x+i,y+j)))) \mod 255$$
$$\text{where } p(\cdot) \in \{R(\cdot), G(\cdot), B(\cdot)\}, \tag{10}$$

which is the new value for each color component in this area. As a result, three images are created. Based on these images, a summed-area table for each of them is calculated, and the higher value for each position is transfer as a new color in feature map.

A. Neural Network Classifier

Spiking neural networks (SNNs) are models of impulse transmission between neurons in the brain [19]–[21]. Neurons are composed in three types of layers. The first one is input, which takes all data vectors, then there are hidden layers, which are used to process the information adaptively, and the last is output, which returns the response from the network. All neurons in each layer are connected with all neurons in neighboring layers by connections that are buried with weights. Neuron is passing data forward when the impulse (a spike of voltage) is created after reaching a threshold value ν. Using the SNN helps processing input information in a continuous adaptive way. Input data are forwarded to the system every 120 s, so if any change in conditions is recorded, the network will adaptively process this information for optimal control. In case of voice, we take ten most significant key points from recording of "Hello my home," if there are less points extracted input vector elements are zeroed. In case of sensor readings, we forward vector of these values to the second SNN.

In the mathematical model, we assume that X_{ij} will be a set of neurons in the j layer and $i = j-1$ means the previous layer. Impulse is passing in t_i time, so the neuron can be in one of two states—passive (when ν has not been reached) or active (when ν was reached). State of the neuron is formulated as

$$x_j(t) = \sum_{i\in\Xi_j} w_{ij}\epsilon(t-t_i) \tag{11}$$

where w_{ij} is the weight between ith and jth neurons, Ξ_j is a set of active neurons sending a signal, and $\epsilon(t)$ is called impulse function

$$\epsilon(t) = \frac{t}{\tau}\exp\left(1-\frac{t}{\tau}\right) \tag{12}$$

where τ is a constant value interpreted as membrane potential. Time needed to change the state in neuron is defined as

$$y_j^k(t) = \epsilon(t - t_j - d^k) \tag{13}$$

where d^k is the delay for kth connection, and t_j is time from the first reaching ν value. Using these formulas, the state can be improved as

$$x_j(t) = \sum_{i\in\Xi_j}\sum_{k=1}^{m} w_{ij}^k y_i^k(t). \tag{14}$$

1) Training Algorithm: Training is made by modifications of weights for connections between neurons. The most known algorithm to do that is back-propagation [22]. Let H defines the input layer, I hidden one, and J output. Input data will be represented as $\{[t_1,\ldots,t_h],\ldots\}$, and the needed time to impulse generating on $j \in J$ as $\{t_j^d\}$. Moreover, let define the error function that will be used to find out the actual level of training as the least-squares technique

$$E = \frac{1}{2}\sum_{j\in J}\left(t_j^a - t_j^d\right)^2 \tag{15}$$

where actual time to generate the impulse is marked as t_j^a and expected one as t_j^d. Weights modification is made from the last

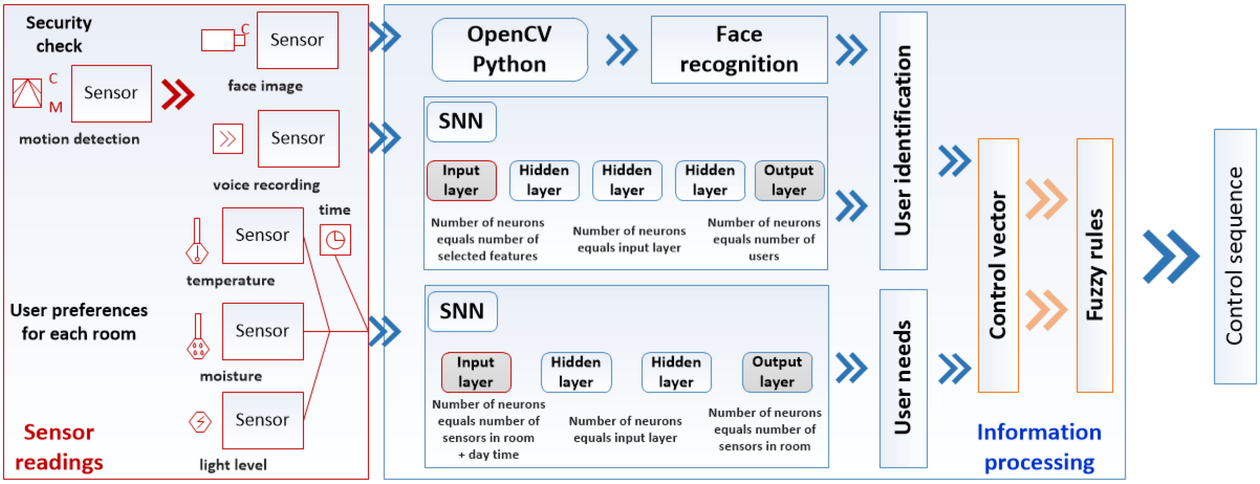


Fig. 5. Construction of the system, in which all sensor readings go through the following validation stages. After motion detection, the system verifies the user by face and voice recognition. If these are positively verified, the readings from conditioning sensors in different rooms are sent to processing, otherwise an alert is sent to operator. As a result of process, according to recognized user, conditions in his(her) room are adjusted in accordance to the day time. Each of heaters has three operating levels, drying units have only ON/OFF mode, while lights in rooms are all with two levels of operation adjusted by users during system setting.

layer to the first one. It forced the introduction formulas for each type of the layer except the input one that does not participate in training. For the last layer, the formula for modification will be

$$\Delta w_{ij}^k = -\eta y_i^k(t_j^a)\delta_j \tag{16}$$

where $\eta = \text{const}$ and δ is

$$\delta_j = \frac{\partial E}{\partial t_j^a}\frac{\partial t_j^a}{\partial x_j(t_j^a)}. \tag{17}$$

Formula for modification of weights between neurons in hidden layers is

$$\Delta w_{hi}^k = -\eta y_h^k(t_i^a)\delta_i \tag{18}$$

where

$$\delta_i = \frac{\sum_{j\in\Xi^i}\delta_j\left[\sum_k w_{ij}^k\left(\frac{\partial y_i^k(t_j^a)}{\partial t_i^a}\right)\right]}{\sum_{h\in\Xi^i}\sum_l w_{hi}^l\left(\frac{\partial y_h^l(t_i^a)}{\partial t_i^a}\right)}. \tag{19}$$

B. Rule-Based Controller

The construction of our classifier is composed of neural networks—each of them is trained with different data to fit user's needs. Face recognition from security cameras is done by using OpenCV library from Python language. Voice recognition is done by our proposed method. The results of all classifications are stored into a vector that is used to make a decision on pattern similarities (original pattern is created after the first network training, assuming that the home was in perfect condition) of users are correctly recognized. The model of the proposed control is presented in Fig. 5. Decision is made on the basis of pattern similarities evaluated using the Takagi–Sugeno–Kang fuzzy model. As input, it takes values from neural components, which are compared using rules in the database. The inputs are calculated as a difference between the results and the original pattern (created in ideal conditions in the building). The rule is constructed as

$$\text{If } (x_1 \text{ is } M_1 \text{ and } x_2 \text{ is } M_2) \text{ then } u_1 = f(x_1, x_2) \tag{20}$$

where x_1 and x_2 are input recalculated values, M_1 and M_2 are fuzzy sets, and f is a function. In order to adapt the controller, we create all the rules in one way

$$\begin{aligned}
&\text{If } (x_1 \text{ is big and } x_2 \text{ is below average})\\
&\qquad \text{then } u_1 = f_1(x_1, x_2)^{1/4}\\
&\text{If } (x_1 \text{ is big and } x_2 \text{ is above average})\\
&\qquad \text{then } u_2 = f_1(x_1, x_2)^{1/2}\\
&\text{If } (x_1 \text{ is small and } x_2 \text{ is below average})\\
&\qquad \text{then } u_3 = f_1(x_1, x_2)^{-1/2}\\
&\text{If } (x_1 \text{ is small and } x_2 \text{ is above average})\\
&\qquad \text{then } u_4 = f_1(x_1, x_2)^{-1/4}.
\end{aligned}$$

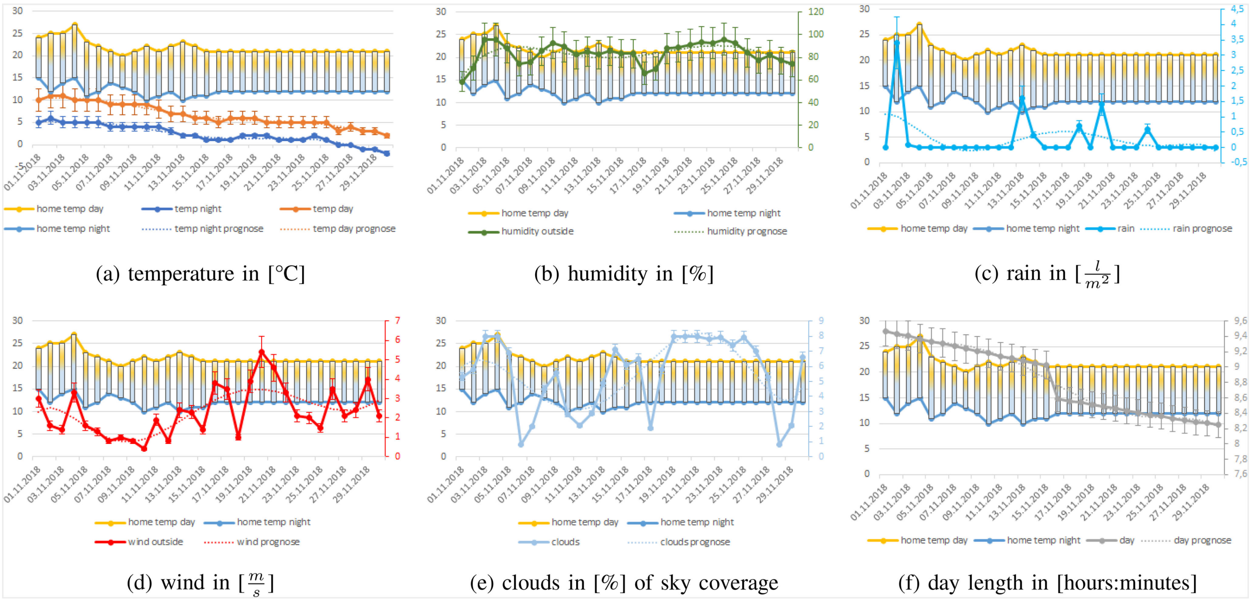
Rules allow to adjust different sensors. As a function f, we use the following quotient:

$$f(x_1, x_2) = \frac{\sum_{i=1}^4 w_i(x_i)u_i(x_1, x_2)}{\sum_{i=1}^4 w_i(x_i) + 1} \tag{21}$$

where w_i defines the minimum t-norm represented by the following equation:

$$w_i(\cdot) = \min\left\{\mu_{M_1}^i(x_1), \mu_{M_2}^i(x_2)\right\} \tag{22}$$

$$\mu_M = \max\left[\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right)\right] \tag{23}$$



(a) temperature in [°C] (b) humidity in [%] (c) rain in [$\frac{l}{m^2}$]

(d) wind in [$\frac{m}{s}$] (e) clouds in [%] of sky coverage (f) day length in [hours:minutes]

Fig. 6. Comparison of temperature control inside the home in contrast to: (a) temperature outside, (b) humidity outside, (c) rain outside, (d) wind outside, (e) clouds outside, and (f) length of day. The data presented in figures as contrast to measured home controlled temperatures are taken from open source dataset at meteomodel.pl for the Katowice region where experiment was done. As we see the system has learned expectations of users and kept without the influence from outside weather conditions. (a) Temperature in [°C]. (b) Humidity in [%]. (c) Rain in [$\frac{l}{m^2}$]. (d) Wind in [$\frac{m}{s}$]. (e) Clouds in [%] of sky coverage. (f) Day length in [hours:minutes].

where a, b, c, and d are positive coefficients and $a \leq b \leq c \leq d$. The returned result is as decision to set operation modes of heaters, drying, and lights to reach the optimal conditions.

IV. Experiments

For the research, we have used installation elements that support smart home environments like intelligent Wi-Fi sockets to control drying units and Wi-Fi light switches to set the level of light in rooms, similarly, we have used intelligent control elements for electric heating system in our flat. Security control modem was built upon two cameras and microphones in corridor and salon. All control elements are placed together with sensors as marked in Fig. 3. As a general knowledge, we have used initial settings for applied sensors. The data for the local data base were collected during 14 days of using the flat. In this period, the data were collected during nights and days. Our experimental setting was developed in our flat, where the family consists of father, mother, and children (one son and two daughters). The goal of this experiment was to train the controller to adjust devices for the lowest energy consumption and optimal user's needs. After training with the knowledge about needs from five users, the controller was switched on to use the local data base for managing the system.

In Fig. 6, we can see sample results of temperature control during November 2018 for one of users. Temperature was adjusted to the needs in first 14 days of November. On the chart, we can see fluctuations in the temperature inside the flat both at night and day. After that period, system started to recognize the needs and set optimal temperature at night for 12° and 21°during day. In the same time, temperature outside was lowering with each new day since this period is winter time in Europe. We can see that the trained system controls conditions keeping expected levels independently of outside weather changes. We have verified how the trained system (using the local data base with the knowledge about the home and user's needs) works. Readings from the sensors were evaluated by trained neural structures in categories: temperature, moisture, light, audio, and video security. To present confusion matrices in Fig. 7(a)–(e), we have used standard measures that describe performance of the proposed solution: TP (true positive), TN (true negative), FP (false positive), and FN (false negative). Using them, we have calculated F_1 score Λ, overlap Ψ, and sensitivity Υ as follows:

$$\Lambda = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \approx 0.88 \tag{24}$$

$$\Psi = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \approx 0.78 \tag{25}$$

$$\Upsilon = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \approx 0.88. \tag{26}$$

We can see that the proposed solution is reaching high precision and sensitivity, what means that our solution was well trained to classify the settings. The value of the overlap shows that decisions taken by the system for some measures can depend on outer conditions, which is true, e.g., for the temperature and humidity, which in period of winter can influence each. In Fig. 7(a), we see the results for temperature control are good since results show high rate of TP and relatively low FN and FP. For moisture control, the number of FP is higher. The results for audio and video security in Fig. 7(d) and (e) show high rates for TP, which are directly related to positive opinions

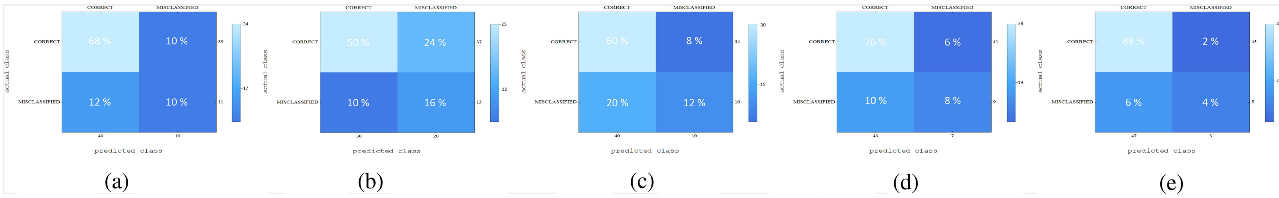


Fig. 7. Sample results of control by the implemented system for sensors readings. (a) Temperature control. (b) Moisture control. (c) Light control. (d) Audio security control. (e) Video security control.

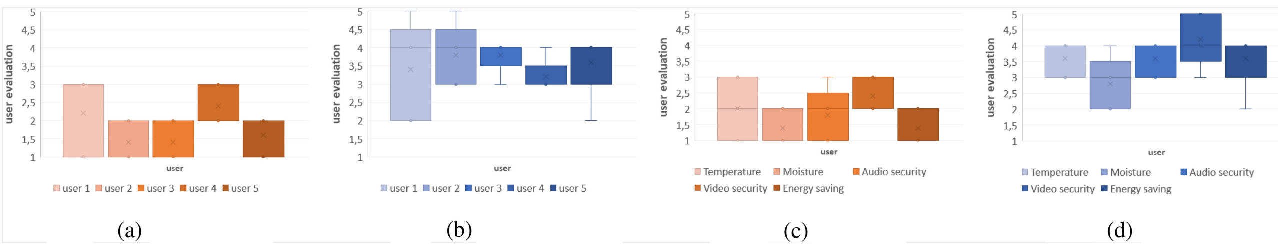


Fig. 8. Comparison of evaluations from the users and averaged results for each of features categories, where each bar represents opinion (5—best, 1—worst). All settings were clearly better evaluated after training. (a) Before training. (b) After training. (c) Before training. (d) After training.

TABLE I
RESULTS OF EVALUATION FROM FIVE USERS OF THE HOME (WITHOUT INTELLIGENT SUPPORT → WITH INTELLIGENT SUPPORT)

	User					
	I	II	III	IV	V	**Avg.**
Temperature	3→4	1→3	2→4	3→3	1→4	2→3.6
Moisture	1→2	2→4	1→3	2→3	1→2	1.4→2.8
Audio	3→4	1→3	1→4	2→3	2→4	1.8→3.6
Video	3→5	2→5	2→4	3→3	2→4	2.4→4.2
Energy saving	1→2	1→4	1→4	2→4	2→4	1.4→3.6
Avg.	2.2→3.4	1.4→3.8	1.4→3.8	2.4→3.2	1.6→3.6	

from all the users. We have also compared results of energy use in the period of experiment to analogical month previous year without control. In previous year, when temperature outside was equivalent energy counters recorded 408 [kW·h] use, while in the period of the proposed smart control counter recorded 360 [kW·h]. This means, we have reached 11.7% of energy save in month with home control. That is quite good result, which confirms positive impact of the proposed control system on the home budget. After one month of using the system, we asked users to give us opinions how they evaluate work of controllers in each of categories. The results compared to opinions without system are presented in Table I. Each of the users was giving a grade in each category using points from 1 (the worst) to 5 (the best). In Fig. 8, we can see the comparison of evaluations. Moisture received the lowest average evaluation of 2.8 points, while security settings received the highest average evaluation of 4.2 points. The other three features were evaluated at the same level. Analyzing the results, we see that avg. evaluation of home comfort was approx. higher of about 1–2 points with the proposed system working. All users expressed better satisfaction with intelligent control of home elements. On the other hand, we can see that moisture settings were less satisfactory. The explanation to this fact is winter period of the experiment, when people may have different feeling of cold, which significantly affects the thermal comfort, and thus, also the feeling of moisture in the air.

V. CONCLUSION

The proposed idea turned out to be efficient since results are promising. The drawback was the necessity to collect enough data to train the system to control elements. The initial knowledge based on situation in various flats must be adjusted to better support further actions of various sensors at home. One of solutions for solving this problem was transfer learning. The system may benefit from results of other similar flats and using this transferred knowledge training may be more beneficial. On the other hand, different quality of the sensors may make the system learn longer. The variety of sensors at market makes some imperfections, since different precision influence the time of training and also the efficiency. The system had knowledge about situations during days and nights in 2 weeks to start training, and finally, efficiency reached over 85%. The system was not tested for homes or blocks as we have designed it on flat. However, we think it may be easily extended. The reason is that modern architectures of blocks support all kinds of conditioning, air control, humidity, temperature, etc. Similarly, new homes are also equipped with various technological advances. Adding a control system may be important idea to create a complex solution of a user-friendly environment. However, some conditions must be assured. In case of temperature control, each flat may want different temperature settings, so when we extend our proposition to a whole block, the control becomes more complex since every heating or conditioning unit must work for special needs. Therefore, how to split/divide control between units for optimal energy use while different needs from different levels, flats, and rooms appear is a task to solve in future research. On the other hand, when implemented in a block, a transfer learning of the system will be much easier. We can also add new sensors and electronics, which can be simply connected to our system. This expansion is possible for window sliders and various home electronics, which together with new electric sockets and switches support remote methods of control also by computer assistants implemented in operating systems in our computers or mobiles. Design of support systems can give many possibilities for industrial applications in electric elements used for composition of smart homes.

VI. Final Remarks

The proposed system was developed to control five categories of home conditions, as expected by respondents of initial survey. The controllers were developed using neural networks and rule-based systems, which are main novelty of our solution. After gathering information from users about their needs, the system adopts in training to expectations and keeps conditions at home. Optimal setting not only improved the comfort of living but also enabled 11.7% energy save. Despite some drawbacks, the effectiveness of the proposed model showed that it is a good direction of research. Especially, the construction of the framework itself. The exchange of information between the cloud and the local system being installed at home can be very valuable in case of more complex structures. The proposed solution may be improved with additional features such as parallelization, acceleration of network's learning, or better adjustment of the controller itself due to more fitted fuzzy logic rules system. These possibilities will be investigated in the future research.

References

[1] T. Hargreaves, C. Wilson, and R. Hauxwell-Baldwin, "Learning to live in a smart home," *Building Res. Inf.*, vol. 46, no. 1, pp. 127–139, 2018.

[2] A. N. Aicha, G. Englebienne, and B. Kröse, "Unsupervised visit detection in smart homes," *Pervasive Mobile Comput.*, vol. 34, pp. 157–167, 2017.

[3] J. Dahmen, B. L. Thomas, D. J. Cook, and X. Wang, "Activity learning as a foundation for security monitoring in smart homes," *Sensors*, vol. 17, no. 4, pp. 737:1–737:17, 2017.

[4] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, "A privacy preserving communication protocol for IoT applications in smart homes," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1844–1852, Dec. 2017.

[5] F. Amirjavid, P. Spachos, and K. N. Plataniotis, "3-D object localization in smart homes: A distributed sensor and video mining approach," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1307–1316, Jun. 2018.

[6] M. Gochoo, T. Tan, V. Velusamy, S. Liu, D. Bayanduuren, and S. Huang, "Device-free non-privacy invasive classification of elderly travel patterns in a smart house using PIR sensors and DCNN," *IEEE Sensors J.*, vol. 18, no. 1, pp. 390–400, Jan. 2018.

[7] J. Rafferty, C. D. Nugent, J. Liu, and L. Chen, "From activity recognition to intention recognition for assisted living within smart homes," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 3, pp. 368–379, Jun. 2017.

[8] I.-Y. Joo and D.-H. Choi, "Distributed optimization framework for energy management of multiple smart homes with distributed energy resources," *IEEE Access*, vol. 5, pp. 15 551–15 560, 2017.

[9] A. Khakimova *et al.*, "Optimal energy management of a small-size building via hybrid model predictive control," *Energy Buildings*, vol. 140, pp. 1–8, 2017.

[10] G. Dong and Z. Chen, "Data-driven energy management in a home microgrid based on Bayesian optimal algorithm," *IEEE Trans. Ind. Inform.*, vol. 15, no. 2, pp. 869–877, Feb. 2019.

[11] H. Fernandes, M. Alam, J. Ferreira, and J. Fonseca, "Acoustic smart sensors based integrated system for smart homes," in *Proc. Int. Smart Cities Conf.*, 2017, pp. 1–5.

[12] M. Kaneko, K. Arima, T. Murakami, M. Isshiki, and H. Sugimura, "Design and implementation of interactive control system for smart houses," in *Proc. IEEE Int. Conf. Consum. Electron.*, 2017, pp. 283–284.

[13] A. Jurenoks and D. Jokić, "Sensor network information flow control method with static coordinator within Internet of Things in smart house environment," *Procedia Comput. Sci.*, vol. 104, pp. 385–392, 2017.

[14] Q. Zhang, L. T. Yang, Z. Chen, P. Li, and F. Bu, "An adaptive droupout deep computation model for industrial IoT Big Data learning with crowdsourcing to cloud computing," *IEEE Trans. Ind. Inform.*, vol. 15, no. 4, pp. 2330–2337, Apr. 2018.

[15] B. L. R. Stojkoska and K. V. Trivodaliev, "A review of Internet of Things for smart home: Challenges and solutions," *J. Cleaner Prod.*, vol. 140, pp. 1454–1464, 2017.

[16] I. A. Berg, O. E. Khorev, A. I. Matvevnina, and A. V. Prisjazhnyj, "Machine learning in smart home control systems-algorithms and new opportunities," *AIP Conf. Proc.*, vol. 1906, no. 1, 2017, Art. no. 070007.

[17] E. Sejdić, I. Djurović, and J. Jiang, "Time–frequency feature representation using energy concentration: An overview of recent advances," *Digit. Signal Process.*, vol. 19, no. 1, pp. 153–183, 2009.

[18] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vision*, 2006, pp. 404–417.

[19] W. Maass, "Networks of spiking neurons: The third generation of neural network models," *Neural Netw.*, vol. 10, no. 9, pp. 1659–1671, 1997.

[20] N. K. Kasabov, "NeuCube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data," *Neural Netw.*, vol. 52, pp. 62–76, 2014.

[21] P. A. Merolla *et al.*, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014.

[22] S. M. Bohte, J. N. Kok, and H. La Poutre, "Error-backpropagation in temporally encoded networks of spiking neurons," *Neurocomputing*, vol. 48, no. 1, pp. 17–37, 2002.

Marcin Woźniak received the M.Sc. degree in applied mathematics from the Silesian University of Technology, the Ph.D. degree in computational intelligence from the Czestochowa University of Technology, and the D.Sc. degree in computational intelligence from the Czestochowa University of Technology, in 2007, 2012, and 2019, respectively.

He is an Associate Professor with the Faculty of Applied Mathematics, Silesian University of Technology, Gliwice, Poland. In scientific career, he was a Visiting Researcher with universities in Italy, Sweden, and Germany. He has authored and coauthored around one hundred research papers in international conferences and journals. His main research interests include neural networks with their applications together with various aspects of applied computational intelligence.

Dr. Woźniak is a Scientific Supervisor in editions of "the Diamond Grant" and "the Best of the Best" programs for highly talented students from the Polish Ministry of Science and Higher Education. He participated in various scientific projects (as a Lead Investigator and as a Scientific Investigator or a Participant) with Polish and Italian universities. He has served as an Editor for various special issues of *Sensors*, IEEE ACCESS, Computational Intelligence and Neuroscience (CIN), etc., and as an Organizer or a Session Chair at various international conferences and symposiums like IEEE Symposium Series on Computational Intelligence, etc.

Dawid Połap received the M.Sc. degree in applied mathematics (with honors) from the Silesian University of Technology and the Ph.D. degree in computational intelligence from the Czestochowa University of Technology, in 2017 and 2019, respectively.

He is an Adjunct with the Faculty of Applied Mathematics, Silesian University of Technology in Gliwice, Gliwice, Poland. He has authored and coauthored more than 50 research papers in international conferences and journals. His main research interests include image processing and various aspects of machine learning.

Dr. Połap was awarded by the Polish Ministry of Science and Higher Education with a "Diamond Grant" for the most talented students. He has participated in various scientific projects (both as a Lead Investigator and as a Scientific Investigator) from Polish and Italian universities. He has served as an Editor for various special issues of *Algorithms*, etc.