Intelligent Living Room System Which Learns Human Activities

Madis VELLAMAE
Tallinn University of Technology
Tallinn, Estonia

Tomonori HASHIYAMA
Grad. School of Information Systems
The University of Electro-Communications
Tokyo, Japan

Abstract

Visions of smart houses and home automation technologies have been around for over three decades. Since that time, computers and technology have made a huge step forward and simple home automation is not so appealing anymore. In this paper we propose and prototype an intelligent adaptive living room system that enhances people’s daily life by figuring out our desires from our natural gestures, facial expressions and speech without direct interaction. Human behavior such as gestures, facial expressions and preferences vary depending on the person and environment. The meaning of the same gesture and facial expression made by different persons, or by the same person but in different situation, can have a different meaning. Therefore the system must be able to recognize the person and the situation and learn the preferences. In order to achieve this, artificial intelligence and machine learning algorithms, such as Hidden Markov Model (HMM) and Growing Hierarchical Self-Organizing Map (GH-SOM) are used. Microsoft Kinect for Windows sensor is used to monitor gestures, voice and locate people in the living room. The information gathered from multiple sensors and users’ desires recognized from gestures and facial expressions are combined in order to make correct decisions. As a result, the system seamlessly enhances people’s daily life by making it more comfortable by, for example, controlling the temperature, room lighting setting and ventilation. The system will learn each individual’s preferences in different situations. It will adapt to users and take actions based on user’s postures, gestures, speech and facial expressions and also location in the room.

Keywords: Adaptive House, Intelligent House, Intelligent System, Smart House.

1 Introduction

This research was carried out in order to propose and prototype an adaptive intelligent system for living room that makes every day life more comfortable.

Research and studies on the topic of home automation has existed for over three decades and many systems for home automation have been suggested and developed [1]. Even though, these systems are reliable and easy to use in most of the cases, most of these systems are not intelligent.

Studies about households that have automated home systems show, that most of the time people are satisfied with the systems, however, they would like those to be more intelligent and with the possibility to have completely personal settings according to each individuals preferences. There are also some other considerations, like high cost, inflexibility, poor manageable, and difficulty achieving security [1]. It is clear enough, that the inhabitants do not want to program their houses themselves and letting the professional technicians do it is rather expensive. Therefore there is a need for system which is able to learn the users’ lifestyle and needs and can train itself accordingly.

The paper is organized as follows. In Section 2, previously conducted research on the field is briefly introduced. In Section 3, the objective of current research is stated. Section 4 describes the proposed system in detail. The results of experiments conducted are stated and explained in Section 5. Finally, the paper is concluded in Section 6.

2 Related research and our objective

Mozer [2] introduced a system called ACHE, which stands for Adaptive Control of Home Environments, is introduced. The goal of ACHE is to train itself by observing the lifestyle and desires of the inhabitants, and learning to anticipate and accommodate their needs. The system controls basic residential comfort systems - air heating, lighting, ventilation and water heating. A prototype of the system was constructed in an actual residence which was completely renovated in 1992. As described in the paper [2], ACHE has two objectives: a) anticipation of inhabitants’ needs; and b) energy conservation.

Gopalakrishna et al. propose Exploiting Machine Learning for Intelligent Room Lighting Applications [3] that is an intelligent dynamic lighting platform in an office breakout area. They consider six features that may influence the desired lighting conditions that the system has to learn from. They also discuss and compare various rule-based classification algorithms and find Decision Table to be the most suitable for such kind of mission.

Based on aforementioned researches, our objective of the current research was to propose and proto-
type an adaptive intelligent living room automation system that learns each individual’s preferences and habits and then acts accordingly by figuring out the desires based on gestures, facial expressions and current environment settings. As a result, the system seamlessly enhances people everyday life by making it more comfortable, by for example, controlling the temperature, room lighting setting and ventilation.

3 Proposed System

3.1 Overview

The proposed system consists of different subsystems, each having it’s own purpose in order to support the goals of the full system. Output of each subsystem is an integer label defining the recognition in it’s own subspace. The main subsystems are (see Fig. 1): a) User recognition; b) Gesture recognition; c) Facial expression recognition; d) Decision module; e) Learning module. In order to achieve the functionality of afore mentioned subsystems, artificial intelligence and machine learning algorithms, such as Hidden Markov Model (HMM) and Growing Hierarchical Self-Organizing Map (GH-SOM) are used.

Figure 1: Overview of proposed system

In the proposed system, Microsoft Kinect for Windows sensor (Fig. 2) is used in order to obtain video stream, monitor users’ gestures, voice and location in the room. High Definition (HD) camera is used for facial expression recognition. Multiple sensors across the living room, such as temperature, lighting, airflow and state sensors of home appliances are used to gather information about current environmental settings. The appliances can then be controlled through dedicated controllers.

The prototype of the system is implemented in C# programming language using following frameworks: Microsoft .NET 4.0, EmguCV (OpenCV wrapper for .NET), Accord .NET, Microsoft Kinect for Windows SDK 1.7.

Figure 2: Microsoft Kinect for Windows sensor

3.2 Face recognition

3.2.1 Face area recognition

The face and facial expression recognition subsystem is implemented using EmguCV, which is a cross platform .NET wrapper to the OpenCV image processing library. OpenCV has a class called FaceRecognizer for face recognition, which offers Local Binary Patterns Histogram (LBPH) method. Since images of faces are high-dimensional dataset, a lower-dimensional representation must be found in order to attain reasonable computation times.

In order to get a picture of a face for recognition, first of all, the faces have to be found on the image. For that, EmguCV’s Haar Feature-based Cascade Classifier for Object Detection, that uses Haar-like features in order to recognize face area, is used. After face regions have been found, the areas are cropped from the picture. In order to improve accuracy of recognition, the cropped images of faces are converted to gray scale and resized to common constant size.

3.2.2 Local binary patterns

Local Binary Patterns (LBP) Histogram method uses LBP operator which labels the pixels of an image by thresholding the neighborhood of each pixel with the value of the center pixel and considering the result as a binary number. A formal description of LBP operator is shown by Eq. (1):

$$LBP(\mathbf{x}_c, \mathbf{y}_c) = \sum_{p=0}^{P-1} 2^p s(\mathbf{i}_p - \mathbf{i}_c)$$  \hspace{1cm} (1)

where \((\mathbf{x}_c, \mathbf{y}_c)\) is the central pixel with intensity \(\mathbf{i}_c\); \(P\) is the number of neighbor pixels; \(\mathbf{i}_p\) is the intensity of neighbor pixel; and \(s\) is the sign function defined as:

$$s(x) = \begin{cases} 
1, & \text{if } x \geq 0 \\
0, & \text{else} 
\end{cases}$$  \hspace{1cm} (2)

The histogram of the LBP labels can be used as a texture descriptor. The idea behind using LBP features is that the face images can be seen as a set of micro-patterns. This way a global description of
the face image is obtained [5]. See Fig. 3 for an illustration of the basic LBP operator.

![Figure 3: The basic LBP operator](image)

During the training process of OpenCV FaceRecognizer, the LBP image is calculated, then it is divided into local regions and a histogram from each is extracted. The spatially enhanced feature vector is then obtained by concatenating the local histograms. These histograms are called Local Binary Patterns Histograms (LBPH).

According to [5], LBP Histogram (LBPH) method outperforms other approaches under different facial expressions and lighting conditions. Due to the fact, our system will use LBP Histogram algorithm for face and facial expression recognition.

Despite the use of LBPH method, the quality of the face image is still very important in order to gain high recognition rates. One big aspect to consider is that in real world scenarios lighting conditions are varying a lot and so are the facial expression of users and also angle of the face in relation to the camera. Due to the reason, many pictures of the same user under different conditions must be provided as training data.

3.2.3 Recognition

After face regions have been found on the image, the areas are cropped from the picture and converted to gray scale which are then passed to LBPH face recognizer. Face recognition is performed by means of projecting image that is to be identified into the computed feature space and using a nearest neighbor classifier. Face recognition can only occur after the recognizer has been trained. The common scenario for face and facial expression recognition is the following: 1) Train the recognizer with images 2) Find faces on pictures 3) Crop the face area 4) Convert image to black and white 5) Pass the image to LBPH face recognizer.

3.3 User recognition

When a new user is introduced to the system, a sequence of images are taken for training the face recognizer. In order to improve initial recognition rates, he or she is asked to turn head to left and right. While system is running, the training database is automatically updated with new images under different lighting conditions and facial expressions and the system re-trains itself.

The person recognition subsystem will output an integer label according to the result of recognized person.

3.4 Facial expression recognition

Previous research [7] has shown high recognition rates of facial expressions using LBPH method.

The facial expression recognition system is trained with images from The Cohn-Kanade AU-Coded Facial Expression Database [8] which includes 486 sequences from 97 posers. Each sequence begins with a neutral expression and proceeds to a peak expression. An example of the images from the database can be seen on Fig. 4.

![Figure 4: Examples of images from The Cohn-Kanade AU-Coded Facial Expression Database (©Jeffrey Cohn)](image)

Since each sequence begins with a neutral expression and the peak expression is on the last image, only the last images of each sequence are chosen for training the facial expression recognizer.

3.5 Gesture Recognition

3.5.1 Overview

The movements of hands are called gestures, in this paper they exist in both space and time, therefore, a gesture is a spatio-temporal pattern which may be static or dynamic or both. The goal of gesture interpretation is to push the advanced human-machine communication to bring the performance of human-machine interaction close to human-human interaction [9].

Our proposed system recognizes 2D hand gestures. Gestures are recognized using the forward topology Hidden Markov Model (HMM) with 5 states (see Fig. 5) in conjunction with Viterbi path [10]. The HMM is trained by Baum-Welch algorithm [10]. The HMM will classify the input gesture and output a label.

![Figure 5: HMM forward topology with 5 states](image)

In the prototype, the subsystem is implemented...
using Accord .NET framework’s built in HMM capabilities.

### 3.5.2 Hidden Markov Models

Hidden Markov Models describe a doubly stochastic process, in which the present state of the system is not directly observable. At regularly spaced discrete times, the system undergoes a change of state according to a set of probabilities associated with that state [10].

HMMs have been applied to gesture recognition with great success [11]. The earliest approach to use HMMs to recognize gestures was by Yamamoto et al. [12] to recognize tennis swings.

Hidden Markov Model (HMM) can be described by Eq. (3):

\[
\lambda = (A, B, \pi)
\]

where [10]:

- The states \( S = \{s_1, s_2, \ldots, s_N\} \) where \( N \) is a number of states.
- An initial probability for starting from each state \( \pi_i, i = 1, 2, \ldots, N \) and \( \pi_i = P(s_i) \) at initial step. For forward HMM, the initial state probabilities have the property \( \pi_i = \begin{cases} 0, & \text{if } i \neq 1 \\ 1, & \text{if } i = 1 \end{cases} \).
- \( A = \{a_{ij}\} \) is the \( N \)-by-\( N \) transition matrix and \( a_{ij} \) is the probability of a transition from state \( S_i \) to \( S_j; 1 \leq i, j \leq N \).
- The set of possible observations (emissions) \( O = \{o_1, o_2, \ldots, o_T\} \) where \( T \) is the length of gesture path.
- The set of discrete symbols \( V = \{v_1, v_2, \ldots, v_M\} \) where \( M \) is the number of discrete symbols (in our case 12).
- An \( N \)-by-\( M \) observation matrix \( B = \{b_{im}\} \) where \( b_{im} \) is the probability of outputting symbol \( v_m \) while in the state \( s_i \).

#### 3.5.3 Feature vector

It is reasonable to assume that the length and speed of a gesture will vary amongst users. Therefore, selecting good features for recognizing gestures have significant role in the system performance in terms of recognition rates. There are three basic features from a gesture trajectory [11]: a) location; b) orientation; and c) velocity. The previous research [11] proved the orientation feature to be the best in terms of accuracy of 2D gesture recognition. Due to the fact, our proposed system uses orientation as feature vector.

The orientation in determined between two consecutive points of gesture path as shown by Eq. (4) where \( T \) is the length of gesture gesture path. We wish to use an HMM with a discrete observation symbol density so the orientation \( \theta_t \) is quantized by dividing it by \( \frac{\pi}{2} \) and rounded to the nearest integral value in order to generate discrete codewords from 0 to 11.

\[
\theta_t = \arctan\left( \frac{y_{t+1} - y_t}{x_{t+1} - x_t} \right); t = 1, 2, \ldots, T - 1
\]

![Figure 6: (a) the orientation between two consecutive points and (b) 12 directional codewords from 0 to 11](image)

#### 3.5.4 Gesture spotting

Gesture spotting is a way of locating meaningful gestures from a stream of input paths where it is critical to locate the start point and the end point of a gesture pattern [13]. Lee et al. have introduced the concept of a threshold model [13] that calculates the likelihood threshold of an input pattern and provides a confirmation mechanism for the provisionally matched gesture patterns. They propose that the threshold model is constructed by copying all the states from all gesture models and fully connect the states (ergodic model). A gesture is recognized only if the likelihood of the best gesture model is higher than that of the threshold model.

#### 3.6 Deciding the output action

All the output labels from different subsystems handling the input data is combined into a n-dimensional input vector for the decision making subsystem, where \( n \) is the number of outputs. The decision making subsystem composes of Growing Hierarchical Self-Organizing Map (GH-SOM) [14] and Decision Tables [16].

GH-SOM is a is a growing variant of the popular Self-Organizing Map (SOM) [15], a type of Artificial Neural Network (ANN) that is trained using unsupervised learning to produce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, called a map. We use GH-SOM in order to categorize the high dimensional input vector. The architecture of GH-SOM can be seen on Figure [7].
Based on the research of Gopalakrishna et al. [3], we believe that Decision Tables are most suitable for matching the output label of GH-SOM with action to be executed by the system. In order to train Decision Tables, the system continuously monitors the actions of the users and matches changes in sensor data with it. Based on the collected data the Decision Table is retrained if there are changes or new action-consequence pair has been registered.

4 Preliminary results

At the moment, our prototype system is capable of the following: 1) User recognition; 2) Facial gesture recognition; 3) Gesture recognition.

In order to conduct person recognition experiments, the system was trained with images of size 150x150 pixels of 4 different persons from The Cohn-Kanade AU-Coded Facial Expression Database plus the author, 3 images per person. Facial expressions on those training images were neutral. For facial expression recognition experiment, images of happy, sad and neutral facial expressions were used, each posed by 10 subjects from The Cohn-Kanade AU-Coded Facial Expression Database. As for gesture recognition, at first, the system was trained with gestures of: 1) being cold (rubbing body); 2) feeling hot (swiping forehead); and 3) picking up the phone.

In person recognition experiment, two sets of images were chosen to be recognized: in first set, the facial expression was the same as on training images; in second set, the images were chosen from peak facial expressions. Also, an image of person that was not in the training set was introduced. The system recognized all trained persons correctly for both sets of images. However, images of unknown person were recognized as somebody in the training set. Confusion matrix is shown in Table 1 where \( \text{Uk} \) means unknown person. This means, that we need to conduct more experiments in order to find out most optimal threshold for the recognition.

In facial expression recognition experiments, images from The Cohn-Kanade AU-Coded Facial Expression Database were chosen to be recognized. Those images were of the same facial expressions as were used for training. 100% accuracy was achieved. Also, recognition from video stream of one of the authors was experimented. With recognition from video stream, neutral facial expression was often recognized as sad. There were no false recognitions of happy expression. Results of the experiment are shown in Table 2.

In gesture recognition experiments, we noticed that the skeleton data from Microsoft Kinect Sensor was jittery and joints were not correctly tracked when hand crossed other joints. It was difficult to collect good samples for gestures expressing being cold and hot. This needs further research and some data smoothing algorithms to be applied. Therefore, a new training set of 1) circle; 2) swipe left; 3) swipe right; 4) swipe up; and 5) swipe down gestures was combined. High recognition rates were achieved with those gestures, although, we noticed that for the circle gesture, start and end point of the circle are important. When starting the gesture from left side and training data contained only circles started from right, the gesture was not recognized as a circle. Results of the experiment are shown in Table 3.

---

**Table 1: Confusion matrix of person recognition, rates are given in percent.**

<table>
<thead>
<tr>
<th>Subj.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Uk</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>Uk</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table 2: Confusion matrix of facial expression recognition, rates are given in percent**

<table>
<thead>
<tr>
<th>Expr.</th>
<th>Happy</th>
<th>Sad</th>
<th>Neutral</th>
</tr>
</thead>
<tbody>
<tr>
<td>Happy</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sad</td>
<td>0</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>Neutral</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

**Table 3: Confusion matrix of gesture recognition, rates are given in percent**

<table>
<thead>
<tr>
<th>Gesture</th>
<th>Circle</th>
<th>Left</th>
<th>Right</th>
<th>Up</th>
<th>Down</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circle</td>
<td>70</td>
<td>0</td>
<td>20</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>Left</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Right</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Up</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>Down</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>
5 Conclusions

This paper proposes an adaptive intelligent living room system that makes everyday life more comfortable. The system uses Microsoft Kinect for Windows sensor in order to monitor gestures and location of the users and HMM is used to recognize gestures. Furthermore, facial expressions are recognized on video stream from HD camera. The system learns it’s users’ behavior and re-trains itself over time. Results of experiments conducted so far show great potential of the proposed system. The system is capable of differentiating the same gesture and facial expression made by different persons, or by the same person but in different situation. This is fairly important, because human behavior such as gestures, facial expressions and preferences vary depending on the person and environment.

6 Acknowledgments

The author would like to thank:

- Japanese government and The University of Electro-Communications
- JUSST program and JASSO staff
- JASSO for scholarship

References


