Recognition of Human Identity by Detection of User Activity

Giuseppe Scardino, Ignazio Infantino, and Filippo Vella

ICAR - CNR, Viale delle Scienze, Ed 11 - 90128 Palermo, Italy
{scardino,infantino,filippo.vella}@pa.icar.cnr.it

Abstract. The paper describes a system able to recognize the users identity according how she/he looks at the monitor while using a given interface. The system does not need invasive measurements that could limit the naturalness of her/his actions. The proposed approach clusters the sequences of observed points on the screen and characterizes the user identity according the relevant detected patterns. Moreover, the system is able to identify patterns in order to have a more accurate recognition and to create prototypes of natural facial dynamics in user expressions. The possibility to characterize people through facial movements introduces a new perspective on human-machine interaction. For example, a user can obtain different contents according her/his mood or a software interface can modify itself to keep a higher attention from a bored user. The success rate of the classification using only 7 parameters is around 68%. The approach is based on k-means that is tuned to maximize an index involving the number of true-positive detections and conditional probabilities. A different evaluation of this parameter allows to focus on the identification of a single user or to spot a general movement for a wide range of people The experiments show that the performance can reach the 90% of correct recognition.

1 Introduction

Over the last few years the approach followed in the field of human-computer interfaces (HCI) has sensibly changed. The focus has been shifted on the so-called human-centered design, namely the creation of interaction systems made for humans and based on models of human behaviour [1] and cognitive capabilities [2]. This type of design requires a thorough analysis and proper processing of the information flowing in man-machine communication: linguistic messages, the non-linguistic vocalizations, emotions, attitudes, facial expressions, head movements and hand movements, body posture, and, finally the context in which they are transmitted [3]. In general, the modelling of human behaviour is a challenging task and is based on various behavioural signals: behavioural and affective states (e.g. fear, joy, inattention, stress), the manipulative behaviour (actions used to act on environment objects or self-manipulative actions such as lip biting), the specific signs of culture (conventional signs, such as a head nod or a thumbs-up). The behaviour detected by the actions should also be associated

L. Marinos and I. Askoxylakis (Eds.): HAS/HCII 2013, LNCS 8030, pp. 49–58, 2013.

to a model of human intentions [4] able to take into account the context and be consistent with a cognitive model of the user [5]. Such models could then be integrated into a cognitive architecture with the aim of representing not only the user's mental model [6] but also the main mechanisms of human reasoning such as perception, memory, decision, planning, emotional and affective states, motivation, sociability, and so on (see for example [7], [2]).

A full understanding of human behaviour [1] hinges on the perception of complex signals such as facial expressions, posture and body movements and on the modelling of the context through the identification of objects, and interactions with other components the real environment. The modern techniques of computer vision, and sophisticated machine learning methods allow us to collect and process such data in a more accurate and robust way [8]. If an automated system captures the temporal extension of these signals, it is possible make predictions and create expectations about their possible evolution. It is also possible to detect human intentions, in a simplified way, classifying the elementary actions of a human agent and identifying the usual task associated to the action[4]. The particular way to execute a given task is the basis of a biometric recognition.

For example in [9] the identity is guessed from footsteps, a multimodal system in [10] uses face and speech information, dynamic keystroke analysis are used in [11], and so on.

1.1 Aims and Motivation of the Presented Work

On the basis of the above considerations, the paper proposes a system to model user behaviour and identity recognition in a common real situation: when a user is in front of a computer screen her/his actions are bound to what she/he is viewing at and the way she/he can interact with the application. A way to characterise the user behaviour is to consider head/eyes movements, facial expressions and which region of the monitor scene is observed. The aim of this work is to capture the user behaviour when she/he is browsing an internet page or is using a software interface. This application allows to classify the user reactions in front of a computer and to distinguish different users by its personal movement when interacts with the computer.

2 User Activity Detection

The combination of computer vision [12] and models of human actions [13] make possible to design sophisticated user interfaces and user modelling systems. The proposed system uses the Microsoft Kinect camera to track the point of the screen where the user is looking at. The Kinect system allows to obtain information on skeleton and face movements, using an infrared sensor and a VGA camera (640 x 480 pixel). Using the Microsoft SDK 1.5, the Kinect camera is able to provide information about the user position and to segment the user head. The head position and orientation are characterised with angles along three axes for all the possible orientations, according the movements of yaw, pitch and roll.

The values of the angles provide information about where the user is looking. Given the orientation of the head, a line orthogonal to the eyes line and parallel to the desk is considered. The point where this line meets the screen plane is the point where the user is looking at. The user activity is described through: gaze tracking, facial expression, face coefficients.

2.1 Gaze Tracking

To exactly estimate the point where the user is looking, it is necessary to know where the monitor and the user are positioned in the space. This information is obtained though a calibration step, where it is possible to know the monitor position and the user position in 3D space. The procedure requires that the user, positioned in front of the monitor, looks for few seconds at each monitor corners remaining in a fixed position. In this phase, the angles of the head are saved and provide a reference for all the head movements.

The position where the user is looking on the screen is calculated comparing the angles of the head at a given moment with the values stored in the calibration phase. In order to make the algorithm more robust, the values of all four corners are stored although three points would be enough to perform the calibration. If a value is not detected or it is affected by a large error it is possible to estimate the correct parameters trusting on three of the four values.

After the calibration session, the user can use normally the computer. During the work session, the system stores the information about the point where the user is looking at. The calibration values are valid for different sessions until the monitor, or the kinect camera or the user position are moved. This feature allows to make an unique calibration per user and use the system parameters for multiple captures without making any others calibrations.

Tracking Precision. The chosen method is quite simple, but it provides promising results and it is not constrained to a fixed user position. The user can move in a circle with radius of 35/40 cm from the initial calibration position with a slight error in gaze estimation. This case covers the standard scenario where the user is sat at the desk and can move its chair in a limited space. Considering the case shown in figure 1, the user is in front of the monitor. The calibration estimates the values of position and angles of user head when she/he looks at the points A and B. The C point is calculated as follows:

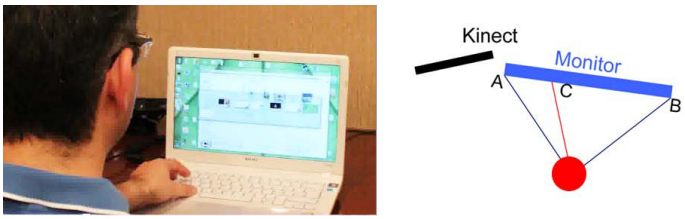

Fig. 1. User position in front of the monitor and Kinect sensor

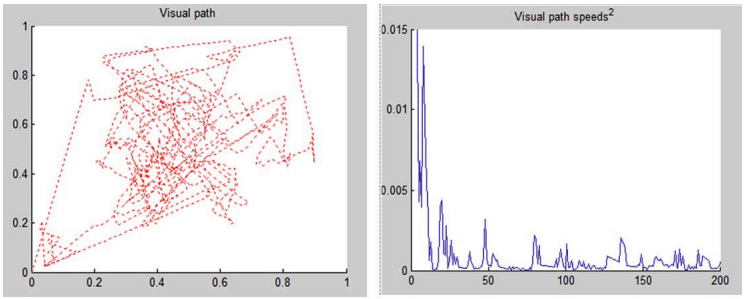

Fig. 2. Tracking activity of human working on his laptop. On the left (a) is represented the diagram of estimated points observed on the monitor and on the right (b) the velocity of movements.

1. roto-translate the values to obtain the center of the axis on the user position;
2. calculate the max value between the saved angle on the top right and bottom right monitor corners in ZX plan;
3. calculate the min value between the saved angle on the top left and bottom left monitor corners in ZX plan;
4. calculate the difference between max and min value calculated at point 2 and 3: $deltaAngleZX = maxValue - minValue$;
5. the mouse position on x screen cordinates are obtained by this formula:

$$xPosition = \frac{actualPosition.ZXangle - maxValue}{deltaAngleZX} * xScreenResolution$$

The same procedure is used to obtain the screen position on y axis using the angles on ZY plan. The obtained values have ripples and the evaluation is affected from noise given by Kinect sensor. To make the estimation more stable has been employed a Kalman filter [14] that allows to have a robust estimation also when noise is present. The Kalman filter is a recursive estimator which evaluates the state of a dynamic system from a series of measurements. This filter has the drawback of being not sufficiently responsive when there are small variations in the input data. An example of acquired data during a tracking session are shown in figure 2.

To measure the accuracy of the tracking system we developed a routine that shows a small moving rectangle on the screen, while the user tracks it with the gaze. The system calculates the difference between the rectangle position and the value on the screen calculated according with the user gaze position. Experiments show a mean errors of about 9.0 pixels (both along x axis and y axis), using a monitor resolution of 1366x778 pixels.

2.2 Extraction of Facial Expression

To detect the facial expression we used some information about the user and which screen region she/he is viewing. The gathered information are: x, y screen

coordinates, the six coefficients of Animation Units characterizing the human face. These six coefficients, called Animation Units (AU)[15], are a subset coefficients defined in the Candide3 model [16] that uses 87 2D points on the face to track the user head. From these six values it is possible to classify the facial expression considering seven basic expressions: neutral face, upper lip raised, jaw lowered, lip stretched, brow lowered, lip corner depressed and outer brow raised.

The range of these coefficients is between -1.0 and 1.0. The first coefficient indicate the lips movement, the value +1 indicates the lips completely opened and -1 completely closed. The second is referred to the lower jaw movement, so +1 indicates completely opened and -1 closed. The third coefficient, indicates how the lips are stretched, the value is defined as follow: 0 is neutral, +1 is fully stretched (joker smile), -0.5 rounded (pout) and -1 is fully rounded (kissing mouth). The fourth coefficient is an index referred to the brow, -1 is for brow raised and +1 for fully lowered. The fifth, the lip corner depressor, indicates -1 for a very happy smile and +1 for a very sad frown. The last coefficient is an index of the outer brow, -1 indicates fully lowered (a very sad face) and +1 raised (deep surprise).

The afore-mentioned six coefficients are related to the configuration of facial features and can be used to classify the emotional state of the user. Through a series of IF-THEN rules and a set of threshold values basic facial expressions are detected (neutral, smiling, angry, sad, surprised, fearful). The rules and thresholds, as used in Microsoft original source code, allow to obtain the facial expression from the six Animation Units as described here:

if (AU[3] > 0.1 and AU[5] > 0.05) the eyebrows are lowered, so set an angry configuration.
else if (AU[3] < -0.1 and AU[2] > and AU[4] > 0.1) eyebrow up and mouth stretched, fearful configuration.
else if (AU[1] > 0.1 and AU[3] < -0.1) eyebrow up and mouth open, surprised configuration.
else if ((AU[2] - AU[4]) > 0.1 and AU[4] < 0) lips are stretched, assume smiling configuration.
else if ((AU[2] - AU[4]) < 0 and AU[5] < -0.3) lips low and eyebrow slanted up, sad configuration.
else by default, set a neutral configuration.

2.3 Activity by Temporal Sequences

We take into account a dynamic evolution of the users activity by sequence of facial action units. Moreover an important parameter that can be extracted is the movement speed of her/his observed point on the screen. We consider that this speed is an own characteristic of the user and different users have different statistics in the fruition of a content on the computer screen. For each couple of frames we calculate the difference, in absolute value, between the position in the current frame and the position in the previous frame estimating the speed at a given moment. We form in this way a vector containing the values of the six animation units at a given time t and the speed at the same time.

$$\mathbf{V_t} = [p_1, p_2, p_3, p_4, p_5, p_6, s]$$

The value of $\mathbf{V_t}$ are saved every 0.3 second in order to store the face parameters and to follow the head movement. To consider a temporal evolution of these parameters, a temporal window of 10 samples is considered. The temporal window is then moved considering the newest values and discarding the oldest one. Our hypothesis is that every user has a different behaviour in front of a computer and this is tightly correlated with her/his personality. The behaviour can be extracted detecting characteristic dynamic facial configuration and classifying new detections according their classes. The next step is the clustering of captured data ($\mathbf{V_t}$) and aggregate them in homogeneous sets. Each cluster can be annotated counting how many samples of a given user are mapped on the cluster itself. The presence of a single or more labels (identifying multiple users) is considered as a measure indicating how trustable the labels are. If a clusters is composed by values of a single user, the values that are in that cluster are very discriminating and identify with good accuracy the user. On the other side, if a cluster is annotated with labels coming from multiple users the values aggregated in this cluster are bound to multiple users and its values are not very discriminating. These values are not peculiar of a single user and have a reduced identification capability. We adopted the well known k-means to cluster the $\mathbf{V_t}$ vectors with different numbers of target clusters.

2.4 Experiment Setup

In order to test our system with first experiments, we asked four volunteers to use an internet browser in front of a screen, free to take their usual position during a work session with a laptop. The session included the navigation on the internet from the same web page of a popular news site. The dataset is being extended with the capture of user behaviour in relation to different types of sites (e-commerce portals, social networks, web search portals, and so on).

The kinect camera was placed in front of them and close to the monitor. It was asked to the users to act and browse normally according their usual behaviour. The captured values have been elaborated in real time to extract the face parameters and to evaluate the speed of the gaze movement at a given instant.

2.5 K-Means Based Clustering

The users have been using the browser without constraints for approximately 2400 seconds, recording the location of the observed points, the six coefficients of facial expression, and a thumbnail image of the observed regions. The dataset is composed of 6659 acquisitions of the parameters of interest. For this experiment setup, only the six coefficients and the module of velocity of observed location are used to identify the user. Ten sequential instants of these parameters are grouped to create input vector of size of 70 to perform k-means clustering [17].

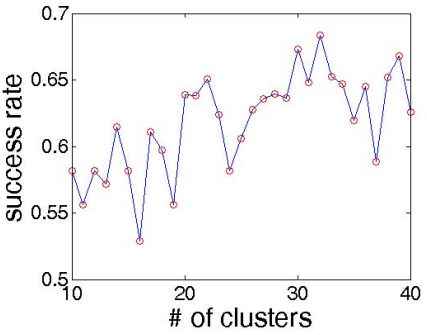

Fig. 3. This figure shows the accuracy rate to vary the cluster numbers

For the training phase sequences of 600 consecutive elements for each user have been extracted (i.e. 2400 acquisitions of 6659) and were created the 2400 sequences considering a temporal window of ten samples.

After fixing the number of clusters, we have performed the standard k-means with a repetition of 10 for the random assignments of initial seeds. The clusters obtained were then labeled with the name of the user with the greatest number of occurrences of the recordings. The optimal number of clusters was obtained by analyzing the trend of the rate of success in the classification see figure 3, and is equal to 32 (corresponding to a success rates of 0.68). It is observed that with a cluster number greater than 10, there are many clusters that collect the data of a single user, while the remaining portion is often related to only 2 users, but with a clear predominance of a user relative to each other.

Always with the time window of 10 successive instants, test data have been associated with closest clusters according to the Euclidean distance. The users label associated with the nearest cluster is the one that is attributed to the test input, and can be compared with the correct label.

2.6 Results Evaluation

The success rate of classification obtained using 7 parameters recorded is around 68%, which is an encouraging result that warrants further in-depth testing.

Having chosen an approach based on a supervised k-means, allows us to search among the resulting clusters those that best identify a particular user. In this way we have typical behavioural patterns that may be the target of research to identify a user, or a class of users in case of wide-ranging trials. The identified pattern presents correct recognition rates that can reach over 90% in our experiments.

To determine the ore representative pattern (center cluster), we have created an index R that takes into account not only the true-positive rate, but also the probabilities P1 and P2. R is the index of representative of the cluster in relation to the user

$$R_u = P1_i * P2_i * (TP_i/N2_i) * 10^4 \tag{1}$$

where

$$P2_i = \frac{N2_i}{\sum_{i=1}^{32}(N2_i)} \tag{2}$$

with $i = 1, ..., 32$ and $u = 1, ..., 4$

Table 1 shows the calculated values of the 32 clusters obtained in the trials. The first column shows the cluster id. The second one is relative to the label of the user who is present in greater numbers in that cluster. The third column shows L1, i.e. the percentage of that label with respect to all the labels in the cluster. The fourth column shows the percentage P1 of samples of the training set that belong to the cluster. N2 is the number of elements in the data set of tests that were considered to belong to the cluster. TP is the number of true positives detected. R is the index of representative of the cluster in relation to the user.

Table 1. Cluster Stats. The table is divided in two columns, on the left are shown clusters from 1 to 16 and on the right are shown clusters from 17 to 32. The 15th and 16th clusters are not calculable because no data from test set is present in the clusters.

Cluster	User	L1	P1	N2	TP	R	Cluster	User	L1	P1	N2	TP	R
1	1	0.87	0.124	103	0	0.00	17	3	0.64	0.061	67	10	1.43
2	1	1.00	0.016	4	0	0.00	18	3	1.0	0.011	145	130	3.44
3	1	0.67	0.052	13	0	0.00	19	3	1.0	0.063	51	15	2.22
4	1	0.94	0.028	131	131	**8.74**	20	3	0.95	0.049	410	231	**26.51**
5	1	1.00	0.033	53	52	4.03	21	3	0.54	0.040	193	183	17.05
6	2	1.00	0.015	159	42	1.52	22	3	0.72	0.018	76	0	0.00
7	2	0.96	0.021	120	29	1.42	23	3	0.84	0.025	124	0	0.00
8	2	1.00	0.011	197	126	3.34	24	4	0.88	0.014	39	35	1.17
9	2	1.00	0.009	232	208	4.28	25	4	0.92	0.016	655	622	**23.18**
10	2	0.71	0.026	81	27	1.64	26	4	0.74	0.022	234	161	8.37
11	2	0.73	0.020	84	50	2.40	27	4	1.00	0.035	32	18	1.50
12	2	1.00	0.026	63	41	2.49	28	4	1.00	0.024	283	253	14.14
13	2	1.00	0.028	105	79	5.19	29	4	0.85	0.038	134	88	7.94
14	2	1.00	0.044	229	194	**20.17**	30	4	1.00	0.060	150	137	19.35
15	2	1.00	0.043	0	0	-	31	4	0.62	0.005	7	7	0.09
16	2	1.00	0.004	0	0	-	32	4	0.91	0.014	74	34	1.10

2.7 User Distinctive Basic Dynamic Facial Expressions

In figure 1 are listed for each user clusters most representative according to the index R. The center cluster corresponding to it shows us the values of the related face configuration coefficients in the 10 consecutive instants. Starting from the element of the entire dataset that is more close to it, it is possible to recover the 3D configuration of the 87 points of the face and their temporal evolution. This set of values constitutes a kind of dynamic basic distinctive facial configuration of the user.

3 Conclusion and Future Works

The presented work demonstrates the potential of the detection of facial expressions with rgbd sensors available on the market today. In particular, in the context of user modelling we have demonstrated that it is possible to recognize the user's identity from his facial expressions and from what she/he observes on a monitor, without invasive measurements that limit the naturalness of her/his actions. The proposed approach uses k-means clustering, and a phase of the training on consecutive users movements, allow us to characterize the identity of user with near 90% of success rate. It is possible to identify patterns in order to have more accurate recognition, and to create prototypes of natural dynamics facial expressions of the user.

The possibility to characterize people through facial movement introduces new view on human-machine interaction, in fact, a user can obtain different contents according your mood, or a software interface can modify itself to keep more attention from a bored user.

The ongoing experimentation involves more extensive testing on a larger number of users, different types of software with which they interact, the possibility to integrate visual memory and the integration of the whole system in a defined cognitive architecture.

Acknowledge. Financial support of the research is partially given by "Ministero dello Sviluppo Economico e Innovazione (MISE), bando MADE IN ITALY", project MI01_00424.

References

1. Pantic, M., Patras, I.: Dynamics of Facial Expression: Recognition of Faci-al Actions and Their Temporal Segments Form Face Profile Image Sequences. IEEE Trans. Systems, Man, and Cybernetics Part B 36(2), 443–449 (2006)
2. Gaglio, S., Infantino, I., Pilato, G., Rizzo, R., Vella, F.: Vision and emotional flow in a cognitive architecture for human-machine interaction. Frontiers in Artificial Intelligence and Applications 233, 112–117 (2011); cited By (since 1996) 1
3. Infantino, I., Rizzo, R., Gaglio, S.: A framework for sign language sentence recognition by commonsense context. IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews 37(5), 1034–1039 (2007); cited By (since 1996) 5.
4. Kelley, R., Tavakkoli, A., King, C., Nicolescu, M., Nicolescu, M., Bebis, G.: Understanding human intentions via hidden markov models in autonomous mobile robots. In: Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction (HRI 2008) (2008)
5. Infantino, I., Lodato, C., Lopes, S., Vella, F.: Implementation of a Intentional Vision System to support Cognitive Architectures. In: Proc. of 3rd International Conference on Computer Vision Theory and Applications VISAPP 2008, International Workshop on Robotic Perception (VISAPP-RoboPerc 2008) (2008)

6. Carroll, J.M., Olson, J.: Mental Models In Human-Computer Interaction. In: Helander, M. (ed.) Handbook of Human-Computer Interaction, pp. 135–158. Elsevier Ltd., Amsterdam (1990)
7. Infantino, I., Pilato, G., Rizzo, R., Vella, F.: I feel blue: Robots and humans sharing color representation for emotional cognitive interaction. In: Chella, A., Pirrone, R., Sorbello, R., Jóhannsdóttir, K.R. (eds.) Biologically Inspired Cognitive Architectures 2012. AISC, vol. 196, pp. 161–166. Springer, Heidelberg (2013)
8. Kelley, R., Tavakkoli, A., King, C., Nicolescu, M., Nicolescu, M.: Understanding Activities and Intentions for Human-Robot Interaction. In: Human-Robot Interaction. InTech (2010)
9. Orr, R., Abowd, G.: The smart floor: a mechanism for natural user identification and tracking. In: CHI 2000 Extended Abstracts on Human Factors in Computing Systems, pp. 275–276. ACM (2000)
10. Ben-Yacoub, S., Abdeljaoued, Y., Mayoraz, E.: Fusion of face and speech data for person identity verification. IEEE Transactions on Neural Networks 10(5), 1065–1074 (1999)
11. Bergadano, F., Gunetti, D., Picardi, C.: Identity verification through dynamic keystroke analysis. Intelligent Data Analysis 7(5), 469–496 (2003)
12. Turk, M.: Computer vision in the interface. Communications of the ACM 47(1), 60–67 (2004)
13. Aggarwal, J., Park, S.: Human motion: Modeling and recognition of actions and interactions. In: Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 3DPVT 2004, pp. 640–647. IEEE (2004)
14. Kalman, R.E.: A New Approach to Linear Filtering and Prediction Problems. Transactions of the ASME–Journal of Basic Engineering 82(Series D), 35–45 (1960)
15. Microsoft: Face tracking, `http://msdn.microsoft.com/en-us/library/jj130970.aspx`
16. Ahlberg, J.: TCANDIDE-3 – an updated parameterized face. Technical report, Dept. of Electrical Engineering, Linkping University, Sweden (2001), `http://www.icg.isy.liu.se/candide/`
17. Basu, S., Bilenko, M., Mooney, R.: A probabilistic framework for semi-supervised clustering. In: Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 59–68. ACM (2004)