An Affect-Responsive Interactive Photo Frame

Hamdi Dibeklioğlu, Ilkka Kosunen, Marcos Ortega Hortas, Albert Ali Salah, Petr Zuzánek

Abstract— **We develop an interactive photo-frame system in which a series of videos of a single person are automatically segmented and a response logic is derived to interact with the user in real-time. The system is composed of five modules. The first module analyzes the uploaded videos and prepares segments for interactive play, in an offline manner. The second module uses multi-modal input (activity levels, facial expression, etc.) to generate a user state. These states are used by the internal frame logic, the third module, to select segments from the offline-generated segment dictionary, and they determine the response of the system. A continuous video stream is synthesized from the prepared segments in accordance with the modeled state of the user. The system logic includes online/offline adaptation, which is based on stored input-output pairs during real-time operation, and offline learning to improve the system response. The fourth module is the application interface, which deals with handling the input and output streams. Finally, a dual-frame module is described to enhance the use of the system.**

I. Introduction

In this paper we describe a dynamic responsive photo frame. This system replaces a traditional static photograph with a video-based frame, where short segments of the recorded person are shown continuously, depending on the input received from the sensors attached to the interactive frame.

The prototypical scenario we consider is the photograph of a baby, set up in a different location, for instance in the living room of the grandparents. While there is no one around, the baby is asleep in the photo frame. Once a viewer arrives, the baby 'wakes up', and responds to the multimodal input received from its viewer. To realize such a system, we propose methods to automatically analyse and segment a number of video sequences to create a response dictionary, combined with a real-time affect- and activity-based analysis tool to select appropriate responses to the user. We then propose a number of system extensions and describe an evaluation methodology.

This system has a number of precursors. A responsive interactive system was proposed in [1], called an Audiovisual Sensitive Artificial Listener. It is a system in which virtual characters react to real users. Facial images and voice information in the videos are used to extract features, which are then submitted to analysers and interpreters that understand the user's state and determine the response of the virtual character. Hidden Markov Models (HMMs) are used in sequential recognition and synthesis problems. In [2], a dialogue model is proposed that is able recognize the user's emotional state, as well as decide on related acts. A Partially Observable Markov Decision Process approach is used with observed user's emotional states and actions.

A project which brought some interaction to photographs is the Spotlight project of Orit Zuckerman and Sajid Sadi, developed at MIT MediaLab[1]. In this project, 16 portraits are placed in a 4×4 layout. Each portrait has nine directional temporal gestures (i.e. one of nine images of the same person can be displayed in the portrait at any given time), which give the appearance of looking at one of the other portraits, or to the interacting user. The user of the system can select a portrait, at which point the remaining portraits will 'look' at it. This project demonstrates the concept of an interactive photograph with static content. While the combination of portraits create novel patterns all the time, the language of interaction is simple and crisply defined.

Another interactive photo frame project is the "Portrait of Cati" by Stefan Agamanolis, where the portrait in question can sense the proximity of the spectator, and act accordingly [3]. When no one is close to the portrait, Cati displays a neutral face. When someone approaches, it selects a random emotion, and displays it in proportion to the proximity of the spectator. If the selected expression is a smile, for instance, the closer the spectator comes, the wider Cati will smile. A similar project is the Morface installation, where an image of Mona Lisa was animated based on the proximity of the interacting person [4]. In this project camera-based tracking is used to determine proximity and head orientation of the user.

The system described in here is different in several aspects from the systems discussed in the literature. In our model the responses of the system are not fixed, but grow in time as the user uploads new videos. In this manner, the system maintains novelty. The two interactive systems we just described are suitable for art installations, but we target a home application, for which novelty plays an important role. Another aspect is that we use real videos in the systems output, with no manual annotation. This is much more challenging than producing appropriate responses through a carefully engineered synthesis framework, where the system has control over the output.

The bottleneck in our system is the real-time interaction, therefore we need to work with lightweight features. We first inspect simple and easy-to-recognize signals, and move to recognition of more complex stimuli. The second aspect that makes our work novel is that the response of the system is not manually (and precisely) defined. A fully automatic segmentation procedure is proposed to create self-contained response patterns, for which the precise semantics is not known at the onset. Our goal is to create a consistent system, in which certain user behaviour is used to produce a certain system response in a consistent manner, and the user is the primary driver of the interaction semantics.

The primary modality we use for real-time analysis is the facial expression of the user. At the core of our real-time module is the eMotion system, which recognizes six basic emotional expressions in realtime [5], [6]. This system uses a Bézier volume-based tracker for the face and a naïve Bayes classifier for expression classification. In a similar vein, Kaliouby et al. previously proposed a MindReader API which models head and facial movements over time by Dynamic Bayesian Networks, to infer a person's affective-cognitive states in real time [7]. In [8] a real-time emotion analysis system was proposed that used an efficient facial feature detection library in conjunction with a number of physiological signals. In the last few years, facial expression and action recognition have seen great improvements. For additional information on facial expression recognition, see [9], [10], [11].

This report is structured as follows. In Section II we describe the proposed system, its separate modules, and its use-cases. Section III describes the algorithmic aspects for each of the modules of the system. Section IV describes the experimental methodology and the assessment of the proposed algorithms within the application context. As the complete system implementation was not completed until

H. Dibeklioğlu and A.A. Salah are with the Informatics Institute, University of Amsterdam, the Netherlands. I. Kosunen is with Helsinki Institute of Technology, Finland. M. Ortega is with University of A Coruña, Spain. P. Zuzánek is with Czech Technical University, Czech Republic.

[1] http://ambient.media.mit.edu/people/sajid/past/spotlight.html

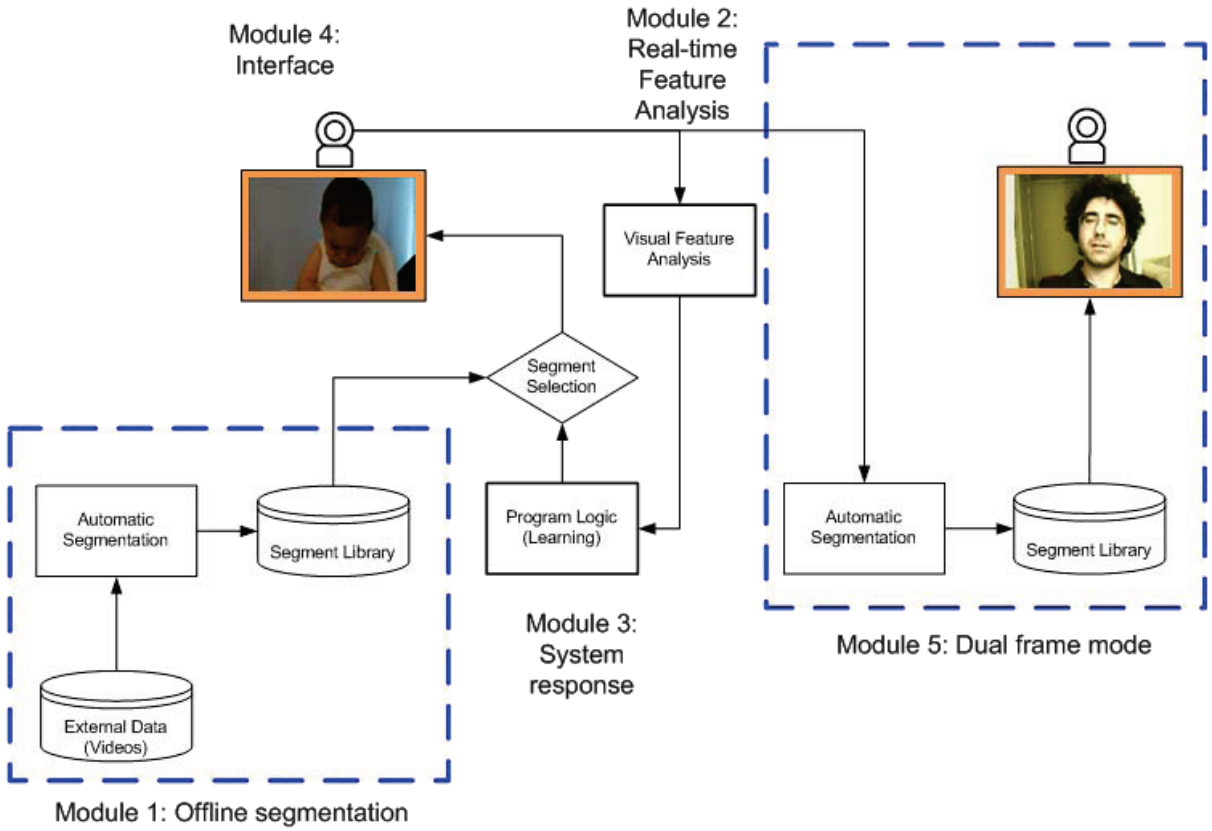


Fig. 1. The overview of the operation of the system. In the dual-frame mode, each frame is used to record new videos that are automatically segmented and added to the segment library of the other frame, establishing an asynchronous communication channel.

the end of the eNTERFACE Workshop, a usability study was not performed. However, such a study was planned during the Workshop. Finally, we conclude in Section V and summarize possible future directions.

II. Design of the Affect-responsive Photo Frame

In this section we describe the logic and the design choices for the affect-responsive photo frame, and introduce the separate modules. Our purpose is to create an emotional/personal digital artifact for continued use. This artifact is designed to adapt to each of its users, as well as prompt the user to adapt to its behavior by guiding the user. We will describe the whole system through the prototypical baby-grandmother scenario. This will help distinguishing the two analysis modules that are similar in principle, but work in offline and online modes, respectively.

A. Overview

The first part of the system is the offline segmentation module. The purpose of this module is to create a response segment library, composed of short video fragments. The input is any number of uploaded videos. In the prototypical use-case, these are the videos of the baby. These videos are analyzed in an offline fashion, and the segments are stored in a segment library. During interaction, the system will play these segments in a particular order.

The second module is the affect and activity analysis module. Here the visual (and in the future audio) input from the user is analyzed in real-time, and a feature vector is generated. This is the module that processes the behaviour of the grandmother in the use-case.

The feature vector is used by the third module, which is the system response logic. The features computed in the second module are used to select an appropriate video segment from the segment library. This module also incorporates learning, to fine-tune its response over time. The system uses its offline period to execute an unsupervised learning routine for this purpose.

The fourth module is the interface. The segments are displayed to the user in the photo frame, depending on the user input. For instance, a smile will trigger a response from the frame, but since we have no mechanism to interrupt the response of the system as soon as new input arrives, a faster feedback mechanism is integrated to the frame in form of coloured glyphs, displayed under the image. Each system response is associated with one glyph, and the brightness of the glyph indicates the proximity of users behaviour to the activating input for that particular response. Thus, if a response is triggered by a smile, a wide smile will activate its glyph immediately, and the response will be played once the current sequence ends playing.

Finally, the fifth module is implementation of the dual frame mode. Here there are two frames, in different locations. Each frame records new segments when it is interacting with a user, analyses those segments, and sends them over the Internet to the other frame system. These segments are added to the segment library of the other frame. They also come with some ground truth, we already know what kind of input elicited these responses in the first place, so we can associate their activation with similar input patterns. This design takes care of content management, and provides constant novelty to the system. Figure 1 shows the overall design of the system, complete with the dual-frame mode.

B. Offline Segmentation Module

The task of the offline segmentation module is to automatically generate meaningful and short segments from collected videos. These

are stored with indicators of affective content and activity levels. Segmentation errors here are not of great importance, as the synthesis module will eventually use all footage material.

The segmentation module uses optical flow calculations to find calm and active moments in the video. Each active moment of a specific length, surrounded by calm moments, is considered as one event and labeled as an active segment. Also, each calm period of sufficient length is labeled as a calm segment. Due to the generic nature of the optical flow calculation, the module is able to detect not only changes in facial expressions, but also events such as hand gestures (waving to the camera) and head movements.

Since we cannot assume a neutral initial pose, or an occlusion-free face area for the duration of the video, assessing facial expressions in these uncontrolled segments is really difficult. Furthermore, our use-case involves a baby, for which the expression analysis requires special training due to different facial morphology. Our experimental results have shown that the optical flow based segmentation creates segments similar to manual segmentation.

C. Real-time Feature Analysis Module

The real-time analysis module is motivated by the need of the system to analyze and characterize user behaviour in order to provide an appropriate response to any particular behaviour. Keeping this in mind, this module can be considered as the data source of the system, as it receives signals from the user and processes them to determine affect- and activity-based content. Since the data are gathered during real-time operation, the module must be able to analyze and process data in a real-time fashion, within reasonable computational assumptions.

The feature analysis module combines the input data into a single feature vector aimed to characterize the current action taken by the user of the system. Modelling the action as a feature vector has the critical advantage that it allows to generate an action space covering the possible feature vector values. This space can be further used to improve system responses to a specific user using machine learning techniques.

In our initial design of the system, we have focused on the following aspects of the user behaviour to be able to model a significant and complete set of different actions:

- Face: The location of the face is the first and most important feature of the system. It allows us to detect the presence of a user to initiate a session, and at the same time it offers information during the session such as movement with respect to camera's frame of reference, and proximity of the user.
- Eyes: The location of the eyes gives us information about the gaze direction of the user. In a system with synthesized responses, matching gaze direction with the user (shared focus of attention) or following the user's location with the gaze are both important for realistic interaction. Since we do not assume any control over the stored segments, there is no meaningful way we can match the gaze information with appropriate segments. However, we do know where the strongest action takes place in each segment, and the gaze information can potentially be matched to such a cue.
- Motion: The activity level of the user is a lightweight feature that can be usefully employed to characterize actions. We divide the face area into a grid and measure the amount of activity in each cell of the grid. This gives us a granular indication of facial activity levels.
- Expression: Facial expression analysis is computationally costly. In our prototype we detect the six prototypical facial expressions (joy, sadness, anger, fear, surprise, disgust). Our system gives soft membership values for each category (including neutral) at 15 fps.

In future, we plan to take more input channels into consideration, like color information (in order to detect presence of some predominant color in the scene, possibly indicating an object) or audio cues from the user.

1) Feature vector components: Fig. 2 shows the information that the real-time analysis module extracts from the input data in a given frame $\mathcal{F}$ in order to construct the feature vector. We also need to consider the action of the user in some interval of time to model the evolution of activity and movement. Therefore, we consider a past frame $\mathcal{F}'$, typically two or three frames prior to $\mathcal{F}$. The feature vector computed at $\mathcal{F}'$ is used in conjunction with the feature vector computed at $\mathcal{F}$ to determine the system response. In addition to these location and activity based features, we use facial expression analysis to provide us with the amount of expression present in each frame. This additional information comes as a normalized vector containing the amount for each one of the six basic facial expressions, plus the neutral expression (represented as $E_1 \dots E_7$). Table I summarizes the feature vector components related to the computations from the frame data.

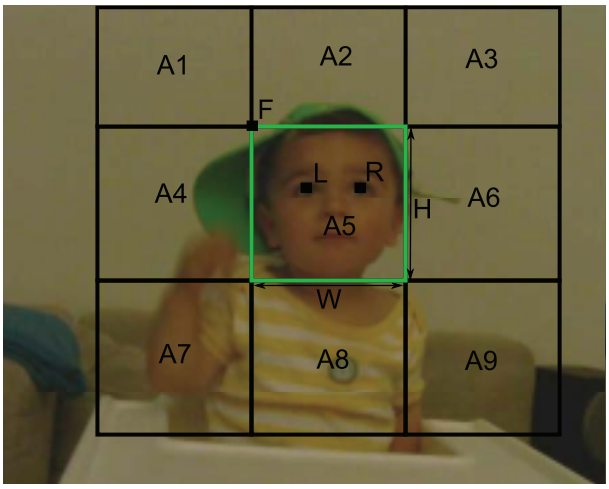


Fig. 2. Features gathered from the visual input of the user for a particular frame. (F_x, F_y) represents the coordinates of the left corner of the face region with respect to the image borders, W is its width and H is its height. $L = (L_x, L_y)$ and $R = (R_x, R_y)$ represent the left and right eye locations, respectively. $A_1 \dots A_9$ quantify the motion activity in each of the nine regions around the face. These nine parameters are measured as vectors, the magnitude ($|A|$) being the amount of average activity in the region and the direction (A_ρ being the mean direction for each region).

D. System Response Module

The system response determines the quality of interaction. If the automatic segmentation is successful, we have a number of short segments that can be played in any sequence. This forms a baseline for the operation of the system. The purpose of the system response module is to improve on this baseline by evaluating the user input in real-time, and by producing consistent and meaningful responses.

We have selected a finite state machine as the abstract representation of the system's operation in this module. This is the simplest possible model for interaction, where input and output relations are clearly (but probabilistically) indicated.

1) Simple prototype: As a simple first prototype, we have developed a simple, two-state finite state machine. The transition between the states were made to depend on the results of the Viola-Jones face detector (i.e. the input consisted of a Boolean variable representing

TABLE I

DESCRIPTION OF THE 41 FEATURES USED TO BUILD THE FEATURE VECTOR FOR FRAME $\mathcal{F}$. FEATURES ARE DEFINED IN TERMS OF THE COMPUTATIONS OF FRAME $\mathcal{F}$ AND REFERENCE FRAME $\mathcal{F}'$. THESE COMPUTATIONS CORRESPOND TO THE ONES EXPRESSED IN FIG. 2.

Index	Calculation	Definition
F_1, F_2	F_x, F_y	Face region left corner coordinates
F_3, F_4	W, H	Width and height of face region
F_5, F_6	$C_x - C'_x$, $C_y - C'_y$	Translation of the face region from $\mathcal{F}'$ to $\mathcal{F}$
F_7, F_8	$\frac{W}{W'}$, $\frac{H}{H'}$	Scale factor of the face region from $\mathcal{F}'$ to $\mathcal{F}$
F_9, F_{10}	L_x, L_y	Left eye center coordinates
F_{11}, F_{12}	$L_x - L'_x$, $L_y - L'_y$	Left eye center translation from $\mathcal{F}'$ to $\mathcal{F}$
F_{13}, F_{14}	R_x, R_y	Right eye center coordinates
F_{15}, F_{16}	$R_x - R'_x$, $R_y - R'_y$	Right eye center translation from $\mathcal{F}'$ to $\mathcal{F}$
$F_{17} \dots F_{25}$	$\lvert A_1 \rvert \dots \lvert A_9 \rvert$	Magnitude of motion vectors in regions A_1 to A_9
$F_{26} \dots F_{34}$	$A_{1\rho} \dots A_{9\rho}$	Motion vector directions for regions A_1 to A_9
$F_{35} \dots F_{41}$	$E_1 \dots E_7$	Amount of basic expressions present in the current frame

"face detected' and 'face not detected'). Fig. 3 depicts this two-state machine. We have used two expressive face action sequences ('Sad' and 'Smile', respectively) from the Cohn-Kanade database [12]. The advantages of using these sequences are that they are normalized with respect to face location and size, well-illuminated, and the expressions start from a neutral face and evolve into the full manifestation of the expression.

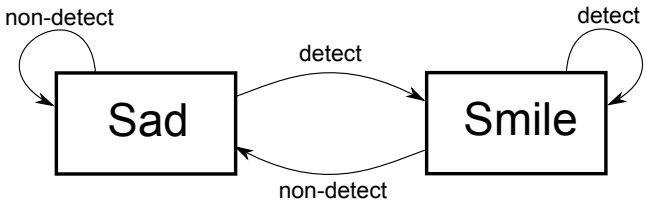


Fig. 3. Scheme of the two-state machine that changes the response of the system according to the results of the Viola-Jones face detector.

This prototype helped us to inspect the behaviour of the system under very simple operating principles, and led to the following observations:

- **Neutral state:** It is unnatural to repeat a video segment multiple times, as the jump from the last frame to the first frame induces an abrupt motion. In the prototype, we ensured the smooth transition between segments by playing them forwards and backwards in a single output cycle. Thus, any transition from the 'Sad' state to the 'Smile' state occurred when the face was displaying a neutral expression. We have decided to use such a 'neutral state' in all our state transitions. We define the neutral state as a frame with very low activity, so that the switch from a forward play to the backward play of the segment has minimum unnatural motion. We have also experimented with morphing between segments to have a smooth transition, but it is difficult to ensure a proper registration of anchor points between frames automatically to have a natural and smooth morphing sequence.
- **Uninterrupted play:** While the response logic requires the system to change behaviour as soon as a new user input is registered, it is unnatural to interrupt a sequence in progress and switch to another sequence. We decided to switch the segments (make a state transition) after the current segment is played completely. For the acknowledgement of the user input, a supplementary indicator is designed. This will be described shortly.
- **System responsivity:** With uninterrupted play, there is a related issue of the length of the video segments. Longer video segments means that the system response is delayed, while the segment is run to its end. A solution might be to eliminate longer sequences from the segment library, or to make them rare events in the operation of the system.
- **Video transitions:** When we have a transition between segments that naturally follow each other, the state transition is very smooth, as expected. However, switching to a distant segment of the same video session, and even more prominently, switching to a segment of another video session can be sharp and unnatural. These transition artifacts should be eliminated using a smoothing or blur function during the transitions. In [13] a subspace method is proposed to control real-time motion of an object or a person in a video sequence. The low-dimensional manifold where the images are projected can be used to define a trajectory, which is then back-projected to the original image space for a smooth transition. While this method is promising for controlling transitions between segments, the subspace projection will not be very successful with the dynamic and changing backgrounds we deal with. Subsequently, we use a much simpler scheme. If we have a transition from frame $\mathcal{F}_1$ to frame $\mathcal{F}_2$, we use an exponential forgetting function to synthesize transition frames, given by equation:

$$\mathcal{F}_3 = \alpha \cdot \mathcal{F}_1 + (1 - \alpha) \cdot \mathcal{F}_2 \tag{1}$$

where $\alpha \in (0; 1)$.

2) Design of the finite state machine: According to the observations we made following the prototype experiment, we have developed a more extensive finite state machine for the affect-responsive photo frame, depicted in Fig. 4.

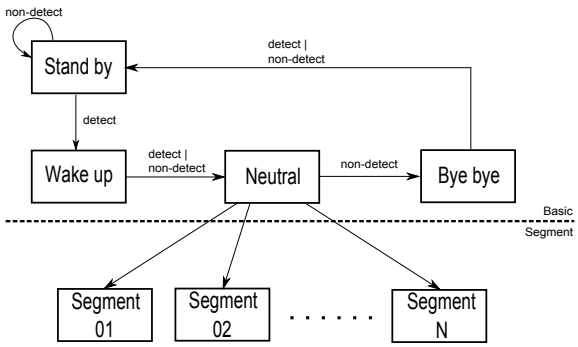


Fig. 4. Scheme of the finite state machine for system response. The two kinds of states are distinguished by the dotted line separator.

In this current implementation, we distinguish between *Basic states* and *Segment states*, respectively.

- **Basic states** are used to provide a general and consistent outlook to the system. When the system is not in use, a default state of low activity is played in loop. In the prototypical use-case, this state would depict the baby asleep in the frame. When a person is present, the baby wakes up, and normal operation is resumed. When the interacting user is absent for a long period, the system returns to the sleep mode. The basic states make sure that this skeleton response is properly displayed. They are assigned manually, although their segmentation need not be manual. The transition from one basic state to another basic state depends solely on the input from the face detector.
- **Segment states** constitute the dynamic part of the finite state machine. Each segment state S_i is associated with one video segment V_i from the segment library, as well as an expected feature vector F_i that will guide the activation of the segment. In the first working prototype of the system, implemented during the eNTERFACE, we have assigned the expected feature vectors randomly, by setting the activity and location based values to zero and setting one or two of the facial expression dimensions to larger values. Thus, basic expressions were used to elicit responses from the system. The segment V_i is activated when the feature vector describing the user's activity is close to the expected feature vector F_i. The 'closeness' here is described statistically, by specifying a Gaussian distribution around each expected feature vector, and admitting activation if the feature vector computed from the user's activity is close to the mean by one standard deviation.

To better understand and remember the user's response for each segment, a game-like strategy is used, where the responses of the system are 'unlocked' one by one. This means that the user has to discover the correct response expected by the system for each new video segment that is shown on the frame. At the beginning, all segments are locked. Once the correct response for the segment in line is found, the particular segment becomes active, and it can always be re-activated by producing the same response.

E. Interface

The interface of the affect-responsive photo frame contains a feedback mechanism to allow the user to see the immediate effect of its actions. This was necessary, as we treat each video segment as an integral entity, and play them in their entirety. Longer segments reduce the responsiveness of the system. In this section we describe our solution based on coloured glyphs, displayed under the photo frame. We also describe the external software packages we made use of to run the system on a stand-alone computer system.

1) Glyphs responses: The developed system aims to provide two-sided adaptation between the user and the machine. Two-sided adaptation means that it is not only the machine that learns the patterns of the user, but also that the user learns the reactions of the machine for different patterns. For this purpose, our system shows glyph responses in real time for patterns derived from the actions of the user at a specific moment. Each segment is pre-assigned to a glyph, which shows the relation between the user's behaviour and the system response, which is encoded by the intensity of the glyph. Higher intensity means that the user activity comes close to the activity that is associated with the particular segment. When the intensity reaches its maximum, the segment is activated. It is shown to the user once, and the user response that elicited the activation of the segment can be repeated for re-activation of the segment at later times.

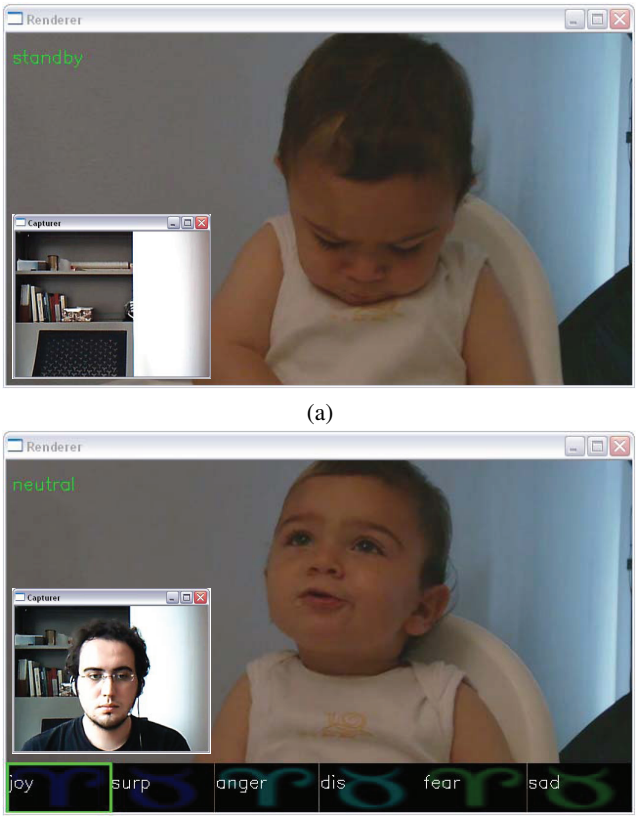


Fig. 5. (a) The stand-by mode and (b) the interaction mode of the system. The lower left corner shows the current camera input to the photo frame as a diagnostic tool.

Fig. 6 shows the system with the glyph responses for each segment. The order of the glyphs (from left to right) reflects the order of segments in the unlock queue. The third glyph glows bright in the example, which means that the current activity of the user is very close to the activity pattern that activated the third segment. The green bounding box around the fourth glyph shows that this is the next segment to be activated, and if the user wants to unlock this segment, he or she should watch this glyph for intensity changes, and adjust its behaviour to increase this intensity. The glyphs on the left side of the green bounding box are already unlocked, and at any given moment, the user can elicit these responses from the system by the same behaviour that was used to unlock the segment initially. Responses for segments to the right of the green bounding box are not known to the user yet.

2) External software: To enable facial expression analysis in our system, we have used the approach proposed in [6]. There is an existing software implementation of this method, packaged into the commercial eMotion application[2]. This program analyses a face image, and classifies the facial expression into basic emotional categories. We have modified some output channels of this expression analysis system and prepared a separate executable to avoid running face detection twice. In the prototype we have prepared, the modified eMotion software runs in addition to the main program, and feeds the facial analysis results to our system over a Telnet connection. Since both interaction and eMotion systems need camera input, we have used a third party camera splitter driver (SplitCam software) which clones the camera input for both applications.

[2]http://www.visual-recognition.nl/

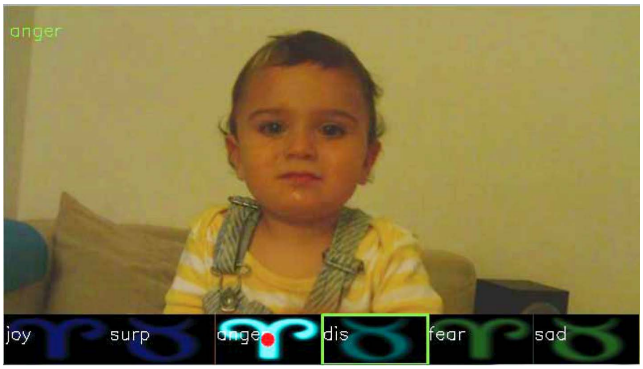


Fig. 6. The output window displays the segment and the glyphs below it. The red circle in the center of the third glyph shows the currently playing segment. In the top-left corner, the name of the currently playing segment is displayed. This is a prototype where each segment is named with basic expression categories. This information is normally not available to the system, as the segmentation is automatic.

F. Dual Frame Mode

The principle behind the dual-frame mode resembles that of the PhotoMirror appliance [14]. In PhotoMirror, a camera is hidden behind a mirror in a home setting, which can record segments of the inhabitants life, and play them back on the surface of the mirror (or another mirror). Similarly, the dual-frame mode of our system implies an asynchronous communication between two persons.

Consider our example scenario with the baby and the grandmother, and add to it a time-differential, where the baby lives in another continent. While the grandmother uses the interactive photo frame in her house, the system will record short segments of her activity (where the face detector is active) and create a segmented behavior library for the grandmother. These segments will be played on a second frame, placed in the baby's room. Through this symmetrical setup, we will also have a kind of action-response ground truth; the segments recorded from the grandmother's frame will be associated to particular segments of the baby. Then, these associations can be used to weakly guide the response patterns. Furthermore, each usage of the frame will send a sequence of new segments to the other frame, taking care of automatic content update for improved novelty.

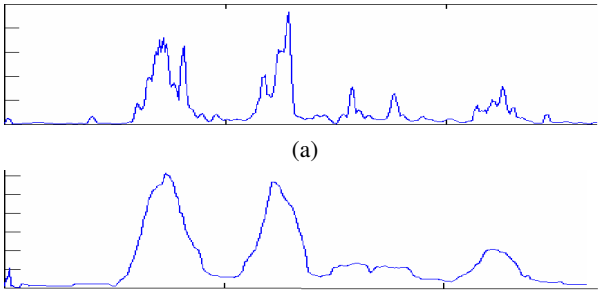
III. Algorithmic Aspects

A. Offline Segmentation Module

The optical flow algorithm can be controlled in various ways depending on the type of segmentation that is desired. First there is the question of whether the optical flow should be calculated between two consequent frames, or a longer period, which might be necessary if the video footage is very static. Secondly, the number of tracked features can be adjusted: in videos with lots of small, uninteresting motion, the algorithm could be set to track only the most important features. Furthermore, the distance between two unique features can be scaled, and the maximum effect of a given feature can thereby be made greater or smaller. This provides robustness against outliers, so that a single large deviation in a given feature, which may be the result of an outlier or noise, does not overly affect the result. With all these options, the module can be used to segment a wide variety of video content. We now discuss several aspects of this module.

1) Optical flow and motion energy: The optical flow calculations were performed with standard routines of the OpenCV library[3]. Optical flow is calculated by selecting the number of points or features in one frame image, and tracking the distance these points have moved in another frame. The tracked features can be selected in a variety of ways [15], but we used the Shi-Tomasi corner detection algorithm [16]. Once the features are selected, they are used using the Lucas-Kanade method for optical flow estimation [17]. The resulting optical flow for each feature in the frame (up to a pre-specified number of features) is then summed to produce a total amount of optical flow for each frame. Because we are interested in events that last for several seconds, the optical flow data are then smoothed using a moving average window to get rid of noise, as well as large fluctuations. This procedure is illustrated in Fig. 7.

After the smoothed optical flow data are generated, the algorithm goes through the data and finds extended periods of activity and calm, and generates both calm and active segments based on this information. The main problem is automatically selecting reasonable thresholds for what is considered an activity and what is not. This is done by defining the average activity as the amount of total optical flow in a frame, and then by taking a certain percentage of this amount to be the threshold for activity segmentation. This allows the algorithm to work with both very active videos, as well as comparatively static ones.



(a)

(b)

Fig. 7. (a) The original optical flow curve and (b) its smoothed version.

2) Frontal face detection: Apart from activity analysis, we rely on face information for both offline and online processing. The first step for this purpose is face detection. While it is possible to do a pass over the video segments that are processed offline to find the best (frontal) face, and combine this with tracking to provide robust face localization, this approach was not implemented. The ideal combination of frame-by-frame face detection and tracking is a possible extension left for the future work.

Because of its proven reliability, we have selected the well-known Viola & Jones algorithm for face detection [18]. For better accuracy we have used the improved version of Viola & Jones algorithm as proposed by Lienthart and Maydt [19]. In this improved version, 45° rotated Haar-like features (see Fig. 8) are used in addition to the original set of Haar features, and a post optimization of boosted classifiers is performed. While rotated Haar-like features increase the discrimination power of the framework, post optimization of the boosted classifiers provides for reduced false alarms.

The Viola & Jones method can be used to detect rotated faces with a cascade trained for this purpose. In general, frontal face images are easier to analyse, and the expression analysis module used in this study needs a frontal face at the initialization step. Therefore, we have used only frontal face cascades to recognize nearly frontal faces.

B. Real-time Feature Analysis Module

In this section we describe the techniques employed in the real-time feature analysis module in order to compute the feature vectors

[3]See David Staven's excellent tutorial "The OpenCV Library: Computing Optical Flow" for more information.

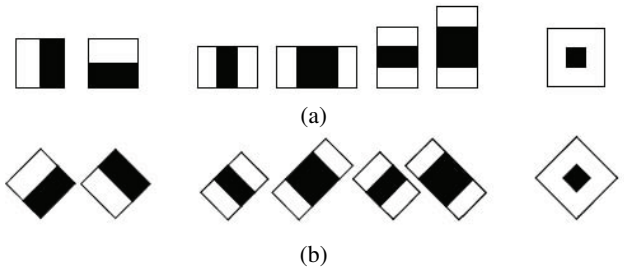


Fig. 8. (a) Haar-like edge, line, and center-surround features, respectively, and (b) their rotations [19].

representing the user actions to the system. For fast online analysis of the camera input we process the location and extent of the face, the locations of the eyes, the content of facial expressions, and the distribution of motion activity. Face detection was discussed in the previous section, the computation of the rest of the features is discussed next.

1) Face analysis: As discussed previously, face analysis starts with face detection. The presence of a face in the field of view of the camera is the main cue we use to arouse the system from its sleep mode. Future work can extend this easily by incorporating sound, such that a loud noise, or the utterance of a particular word can be used as triggers for activating the system.

The detection of eye locations and facial expression analysis both depend the detected face area. For the eye center localization, we used a technique based on isophote curvature, proposed by Valenti and Gevers [20]. The proposed method makes use of isophote properties to gain invariance to linear lighting changes (contrast and brightness) and rotational invariance. For every pixel, the center of the osculating circle of the isophote is computed from smoothed derivatives of the image brightness, so that each pixel can provide a vote for its own center. The eye center is surrounded by pixels whose curvature point in the eye-center direction, so it becomes very salient when these votes are pooled. The use of isophotes yields low computational cost (which allows for real-time processing) and robustness to rotation and linear illumination changes. Fig. 9 illustrates an example of the face and eye location on the feature analysis module.

The features extracted from the face allows for quantification of changes in different aspects. For instance the change in the scale of the facial area is indicative of movement towards the frame or away from it. The eye centers denote shifting foci of attention, although the system we employ does not have sufficient resolution to precisely determine the true focus of attention.

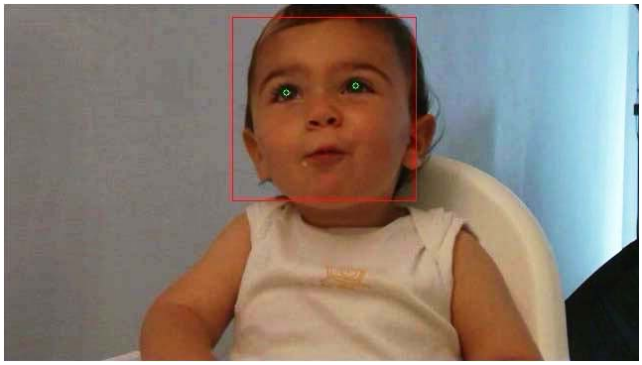
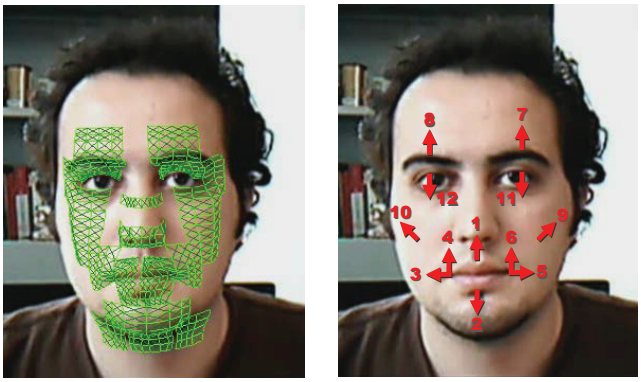


Fig. 9. An example of face and eye center localization.

For facial expression analysis we have used the system which is proposed in [6]. In this approach, the face is tracked by a piecewise Bézier volume deformation (PBVD) tracker, based on the



(a) (b)

Fig. 10. (a) The Bézier volume model. (b) The motion units.

system developed by Tao and Huang [21]. A three dimensional facial wireframe model is used for tracking. The generic face model consists of 16 surface patches, and it is warped to fit the estimated facial feature points, which are simply estimated by their expected locations with respect to the detected face region boundary. These expected locations are learned on a separate training set of faces.

The surface patches are embedded in Bézier volumes to generate a smooth and continuous model. A Bézier curve for $n+1$ control points can be written as:

$$x(u) = \sum_{i=0}^{n} b_i B_i^n(u),$$
$$x(u) = \sum_{i=0}^{n} b_i \binom{n}{i} u^i (1-u)^{n-i}, \tag{2}$$

where the control points b_i and $u \in [0,1]$ control model shape according to Bernstein polynomials, denoted with $B_i^n(u)$. The Bézier volume is an extension of the Bézier curve, and the displacement of the mesh nodes can be computed as $V = BD$, where B is again the mapping in terms of Bernstein polynomials, and D is a matrix whose columns are the control point displacement vectors of the Bézier volume.

After initialization of the facial model, head motion and facial surface deformations can be tracked. 2D image motions are estimated using template matching between frames at different resolutions. Previous frames are also used for better tracking. Estimated image motions are modelled as projections of true 3D motions. Therefore, 3D motion can be estimated using the 2D motions of several points on the mesh.

Expression classification is performed on a set of motion units, which indicate the movement of several mesh nodes on the Bézier volume with respect to the initial, neutral/frontal frame. 12 different motion units are defined as shown in Fig. 10. Unlike Ekman's Action Units [22], motion units represent not only the activation of a facial region, but also the direction and intensity of the motion. A naïve Bayes classifier is used to compute the posterior probabilities of seven basic expression categories (neutral, happiness, sadness, anger, fear, disgust, surprise).

2) Motion Energy and Activity Levels: The motion energy in a particular frame is computed by means of the optical flow. For its computation we use the technique proposed by Lucas and Kanade [17] for registration of images. This method assumes that the flow is essentially constant in a local neighbourhood of pixels under consideration, and solves the basic optical flow equations for all the pixels in that neighbourhood under a least squares criterion. By combining information from several nearby pixels, the Lucas-Kanade

method can often resolve the inherent ambiguity of the optical flow equation. It is also less sensitive to image noise compared to point-wise methods. In our particular case, we have used a pyramidal implementation of the Lucas-Kanade algorithm, developed by Jean-Yves Bouguet [23]. Fig. 11 shows a graphical example of the optical flow algorithm output for a particular frame.

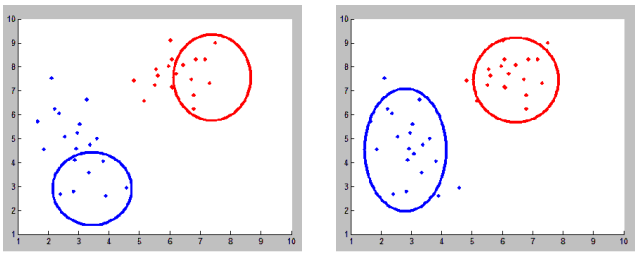
Fig. 11. Example of the optical flow vectors obtained in a frame using the pyramidal implementation of the Lucas and Kanade algorithm [23]. Optical flow vectors are represented as red arrows in the picture.

C. Learning and Adaptation

There are several ways to define interaction between a computer and its human user. The dominant paradigm is to specify the response of the computer precisely, given a certain input from the user. In the interactive photo frame, the manifestation of this paradigm is a static design of the system response logic, and a pre-specified input dictionary. There are however two immediate problems here. Our affect-sensing technology is not robust enough to assign crisp categories to different actions of different users. In other words, if the system is not trained for a specific person, there is a possibility that only a few input words will be activated during the lifetime of the system, and other response possibilities are left unexplored. The second problem is that the response dictionary of the system is not static, and grows each time a new video is added to the system.

The solution to both problems is to model the operation of the system as a dialogue, and let a consistent semiotics emerge through the interaction [24]. In this approach, the initial response of the system is random, or relates weakly to the actions of the user. However, during interaction, action-response pairs are stored. The system then periodically updates its response function by analysing the existing action-response pairs. This serves a two-fold purpose. 1) The response of the system becomes consistent over a period of usage, in that the user becomes able to trigger a certain response by a certain action, and these triggering actions are suitably idiosyncratic. 2) The system, by giving glyph-based feedback to the user, induces certain actions, yet if the user is not able to produce the expected valence, the learning process will shift the required activity to an appropriate level suitable for the user's activity range. In other words, the user and the system simultaneously adapt to each other, and for each user, the final response pattern of the system will be different.

Let F^t denote the feature responses collected during a session of interaction with a user. At a specific moment T of the session, if there are k active segments, and one additional segment that the user seeks to activate at the moment of analysis, there will be $k+1$ feature distributions, represented as $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, with $i = 1 \ldots k+1$. Here, each segment is activated by a feature response that is close to its distribution, as measured by the Mahalanobis distance between $\boldsymbol{\mu}_i$ and F^t.



(a) (b)

Fig. 12. The user responses (each point is one frame) projected to two dimensions. The response thresholds of the system are shown as ellipses for two segments (red and blue in the coloured version), (a) before adaptation (b) after adaptation.

We can take into account the idiosyncratic variations that are conditioned to users by letting the system adapt its response to the user. The terms that determine the system response are F^t, $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$. Since F^t is computed from the camera input recording user's behavior, the adaptation of the system is not concerned with it, but rather involves changing $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$. The idea is to update these variables for an improved modeling of user behavior. Fig. 12 illustrates this idea on a toy example.

The procedure we use for improving the adaptation of the system is simple. At periodical intervals, the parameters of the system are updated as follows:

$$h_i(F^t) = \frac{p(F^t|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\sum_{j=1}^{k+1} p(F^t|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}. \quad (3)$$

$$\boldsymbol{\mu}_i' = \alpha\boldsymbol{\mu}_i + (1-\alpha)\frac{\sum_{t=1}^{T} h_i(F^t)F^t}{\sum_{t=1}^{T} h_i(F^t)}. \quad (4)$$

$$\boldsymbol{\Sigma}_i' = \alpha\boldsymbol{\Sigma}_i + (1-\alpha)\frac{\sum_{t=1}^{T} h_i(F^t)(F^t-\boldsymbol{\mu}_i)(F^t-\boldsymbol{\mu}_i)^T}{\sum_{t=1}^{T} h_i(F^t)}. \quad (5)$$

Here, $h_i(F^t)$ denotes the normalized membership probabilities of a particular set of features F^t for behaviour segment i, $p(F^t|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ is computed from the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, and α is a control parameter. Small values of α will result in small adjustments in the systems behaviour, making it more responsive to the type of activities displayed by the user, as opposed to activities expected by the system. Large values of α may cause inconsistent behaviour in the system, and abrupt changes in response.

IV. System Assessment

We have constructed a working prototype of the system that has basic functionality. We summarize the achievements and assessment in this section.

A. Offline Segmentation

The offline segmentation module is completely implemented. To gain insight into its operation, we have manually segmented a number of video sequences. The system segmentation is then contrasted with manual segmentation, provided by five different persons for each video sequence. During manual segmentation, segments were also assigned labels. We have not constrained these labels in any way; the only constraint was conciseness. The freely available ANVIL multimedia annotation tool[4] was used.

Fig.13 shows a video sequence being processed in the ANVIL tool. Five different segmentations are displayed as rows at the bottom of

[4]http://www.anvil-software.de/

Fig. 13. The manual segmentation of videos and the corresponding automatically determined segmentation.

the video image. The temporal dimension is represented in each row in a left-to-right fashion. Labeled segments are represented as boxes, with the custom label written inside. The smoothed optical flow graph that is appended to the figure (aligned in the temporal axis) is not part of the annotation tool. It displays the result of automatic offline segmentation (as vertical bars) and the optical flow illustrates the 'reasoning' of the system in choosing these segments. The bars are elongated to intersect all five manual segmentations, so as to allow visual comparison. As it is evident from the figure, the most important segment boundaries (as evidenced by consensus among the taggers) is found by the automatic algorithm.

B. Real-time Feature Analysis

The real-time feature analysis module has been partly implemented. As we have discussed, some external software modules were employed to make the system work. The processing is no streamlined, and subsequently the computation burden of real-time feature computation is high. This is a common problem we have noted in similar systems. The SEMAINE API [25], which is developed for building emotion-oriented systems, and which provides a rich set of tools for this purpose, was considered for usage in an early stage of development. Our initial experiments have shown that enabling the facial feature analysis module in this system required a lot of computational resources. The information provided by the API in this modality is quite detailed, which led us to pursue a computationally cheaper system that would nonetheless be useful in guiding the interaction. The full assessment of this module is closely tied to usability studies with real subjects, which was not performed during the Workshop.

C. Real-time Facial Expression Analysis

We have assessed the accuracy of the eMotion software on the Cohn-Kanade AU-Coded Facial Expression Database [12]. In this database, there are approximately 500 image sequences from 100 subjects. These short videos each start with a neutral and frontal face display, and with little overall movement of the face display an emotional expression. Cohn-Kanade dataset has single action unit displays, action unit combinations, as well as six universal expressions, all annotated by experts. Without any manual facial landmark correction, the eMotion software provides 70.68 per cent average classification accuracy for six emotional expressions on this database. We have used 249 of the emotional expression sequences (46 joy, 49 surprise, 33 anger, 37 disgust, 41 fear, 43 sadness sequences) with three-fold cross validation to obtain the accuracy. Warping the generic face model of the eMotion software into a more accurate face representation anchored by seven manually annotated facial feature points (outer eye corners, inner eye corners, nose tip, and mouth corners) by a Thin-Plate Spline algorithm [26] has increased the average classification accuracy to 80.72 per cent. Fig. 14 shows the classification accuracy of the eMotion software for different emotional expressions, with and without manual landmark correction.

V. Conclusions and Future Work

We have developed a working prototype for an affect-responsive photo frame application. Our report sketches the main parts of the application, focusing on only visual features. The voice and speech modalities can be added to the system following the same principles, at the cost of higher computational complexity. We have completed the offline segmentation, feature analysis and the interface modules. The adaptation and dual-frame modules were not implemented during the Workshop. The dual-frame mode of the system is particulary interesting, as it solves the content acquisition and maintenance

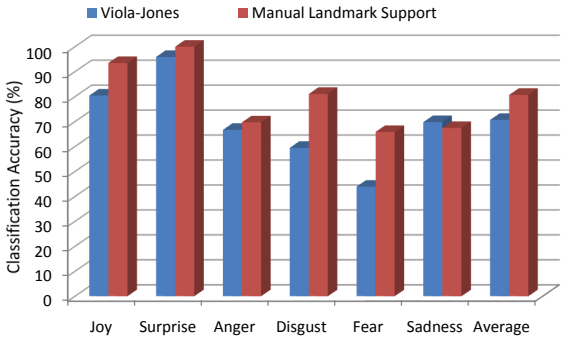


Fig. 14. Classification accuracies of eMotion software for different emotional expressions with and without manual landmark correction.

problems. This is the most important aspect that separates this work from similar digital constructions in the literature. We do not assume carefully recorded and annotated response patterns, but process the input and the output of the system automatically.

Our preliminary experiments have shown us that the proposed system is interesting and engaging. We have not conducted formal usability studies, but earlier prototypes were inspected and practical aspects of design were discussed. A thorough user assessment requires usability studies on a reasonable set of subjects, which can then reveal limitations of the system in longer term usage. It is conceivable that our automatic content management results in less meaningful segments than a hand-crafted set of responses. It remains to be seen whether the constant novelty created by the dual usage of the system is sufficient to offset this handicap, or even to turn it into an advantage.

REFERENCES

[1] M. Schröder, E. Bevacqua, F. Eyben, H. Gunes, D. Heylen, M. ter Maat, S. Pammi, M. Pantic, C. Pelachaud, B. Schuller, et al., "A Demonstration of Audiovisual Sensitive Artificial Listeners", in *Proc. Int. Conf. on Affective Computing & Intelligent Interaction, Amsterdam, Netherlands, IEEE*, 2009.

[2] T.H. Bui, J. Zwiers, M. Poel, and A. Nijholt, "Toward affective dialogue modeling using partially observable Markov decision processes", in *Proc. Workshop Emotion and Computing, 29th Annual German Conf. on Artificial Intelligence*, 2006, pp. 47–50.

[3] S. Agamanolis, "Beyond Communication: Human Connectedness as a Research Agenda", *Networked Neighbourhoods*, pp. 307–344, 2006.

[4] M. Mancas, R. Chessini, S. Hidot, C. Machy, R. Ben Madhkour, and T. Ravet, "Morface: Face morphing", *Quarterly Progress Scientific Report of the Numediart Research Program*, vol. 2, no. 2, pp. 33–39, 2009.

[5] N. Sebe, M.S. Lew, Y. Sun, I. Cohen, T. Gevers, and T.S. Huang, "Authentic facial expression analysis", *Image and Vision Computing*, vol. 25, no. 12, pp. 1856–1863, 2007.

[6] R. Valenti, N. Sebe, and T. Gevers, "Facial expression recognition: A fully integrated approach", in *Proc. 14th Int. Conf. of Image Analysis and Processing-Workshops*. IEEE Computer Society, 2007, pp. 125–130.

[7] R. Kaliouby and P. Robinson, "Real-time inference of complex mental states from facial expressions and head gestures", *Real-time vision for human-computer interaction*, pp. 181–200, 2005.

[8] J.N. Bailenson, E.D. Pontikakis, I.B. Mauss, J.J. Gross, M.E. Jabon, C.A.C. Hutcherson, C. Nass, and O. John, "Real-time classification of evoked emotions using facial feature tracking and physiological responses", *International journal of human-computer studies*, vol. 66, no. 5, pp. 303–317, 2008.

[9] B. Fasel and J. Luettin, "Recognition of asymmetric facial action unit activities and intensities", in *Int. Conf. on Pattern Recognition*, 2000, vol. 15, pp. 1100–1103.

[10] M. Pantic and L.J.M. Rothkrantz, "Automatic analysis of facial expressions: The state of the art", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424–1445, 2000.

[11] Y.L. Tian, T. Kanade, and J.F. Cohn, "Facial expression analysis", *Handbook of face recognition*, pp. 247–275, 2005.

[12] T. Kanade, J.F. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis", in *Proc. AFGR*, 2000.

[13] D. Okwechime, E.J. Ong, and R. Bowden, "Real-Time Motion Control Using Pose Space Probability Density Estimation", in *Proc. ICCV*, 2009.

[14] P. Markopoulos, B. Bongers, E. Alphen, J. Dekker, W. Dijk, S. Messemaker, J. Poppel, B. Vlist, D. Volman, and G. Wanrooij, "The PhotoMirror appliance: affective awareness in the hallway", *Personal and Ubiquitous Computing*, vol. 10, no. 2, pp. 128–135, 2006.

[15] M. Zuliani, C. Kenney, and B. S. Manjunath, "A mathematical comparison of point detectors", *Computer Vision and Pattern Recognition Workshop*, vol. 11, pp. 172, 2004.

[16] J. Shi and C. Tomasi, "Good features to track", in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*. IEEE, 1994, pp. 593–600.

[17] Bruce D. Lucas and Takeo Kanade, "An iterative image registration technique with an application to stereo vision", in *IJCAI*, 1981, pp. 674–679.

[18] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", in *IEEE Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, pp. 511–518.

[19] R. Lienhart and J. Maydt, "An extended set of haarlike features for rapid object detection", in *IEEE International Conference on Image Processing*, 2002, vol. 1, pp. 900–903.

[20] Roberto Valenti and Theo Gevers, "Accurate eye center location and tracking using isophote curvature", in *CVPR*, 2008.

[21] H. Tao and TS Huang, "Connected vibrations: a modal analysis approach for non-rigid motion tracking", in *Proc. CVPR*, 1998, pp. 735–740.

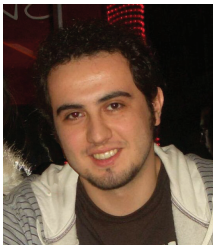
[22] P. Ekman, W.V. Friesen, and J.C. Hager, *Facial action coding system*, Consulting Psychologists Press Palo Alto, CA, 1978.

[23] Jean-Yves Bouguet, "Pyramidal implementation of the lucas kanade feature tracker description of the algorithm", 2000.

[24] AA Salah and BAM Schouten, "Semiosis and the relevance of context for the ami environment", *Proc. European Conf. on Computing and Philosophy (ECAP)*, 2009.

[25] M. Schroeder, "The SEMAINE API: Towards a Standards-Based Framework for Building Emotion-Oriented Systems", *Advances in Human- Computer Interaction*, 2010.

[26] F.L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations", *IEEE Transactions on pattern analysis and machine intelligence*, vol. 11, no. 6, pp. 567–585, 1989.



Hamdi Dibeklioğlu was born in Denizli, Turkey in 1983. He received his B.Sc. degree from Computer Engineering Department of Yeditepe University, in 2006, and his M.Sc. degree from Computer Engineering Department of Boğaziçi University, in 2008. He is currently a research&teaching assistant and a Ph.D. student at Intelligent Systems Lab Amsterdam, University of Amsterdam. His research interests include computer vision, biometrics, pattern recognition and intelligent human-computer interfaces. He works on Human Behavior Analysis under supervision of Professor Theo Gevers.
E-mail: h.dibeklioglu@uva.nl

Ilkka Kosunen is studying computer science at the University of Helsinki and also working as a research assistant at Helsinki Institute for Information technology, where he is developing various biosignal adaptive applications. His research interests include machine learning and biofeedback.
E-mail: ilkka.kosunen@hiit.fi

Marcos Ortega Hortas received his MSc degree in Computer Science from University of A Coruña, Spain, in 2004 and his PhD degree in 2009 from the Department of Computer Science of the same University, with a work focused on the use of retinal vessel tree as a biometric pattern for authentication purposes. He also worked on face biometrics studying the face evolution due to ageing effects as a visiting researcher on the University of Sassari in the Computer Vision Laboratory. He currently serves as a postdoctoral fellow on the University of A Coruña. His research areas of interest are medical image analysis, computer vision, biometrics and human behaviour analysis.
E-mail: mortega@udc.es

Albert Ali Salah received his PhD in 2007 from the Dept. of Computer Engineering of Boğaziçi University, with a dissertation on biologically inspired 3D face recognition. This work was supported by two FP6 networks of excellence: BIOSECURE on multimodal biometrics, and SIMILAR on human-computer interaction, which gave rise to the eNTERFACE Workshops. His research areas are human behaviour analysis, pattern recognition, biometrics, and multimodal information processing. He received the inaugural EBF Biometrics Research Award in 2006, and joined with the Signals and Images group at CWI, Amsterdam as a BRICKS scholar. He is presently a researcher at the Informatics Institute of the University of Amsterdam. He is the co-chair of the eNTERFACE'10 Workshop.
E-mail: a.a.salah@uva.nl

Petr Zuzánek was born in Trutnov, Czech Republic in 1988. He received his B.Sc. degree at Czech Technical University in Prague departure of Cybernetics in June 2010. He is M.Sc. student at Czech Technical University in Prague. He is currently working on the real-time tracking of nearly linear objects from video sequences captured by flying observer at Center for Machine Perception (http://cmp.felk.cvut.cz). This work is a continuation of his bachelor project. His supervisor is Dr. Karel Zimmermann.
E-mail: zuzanpet@fel.cvut.cz