Accepted at the IEEE/ACM 4th International Workshop on Emotion Awareness in Software Engineering (SEmotion 2019), Copyright IEEE

Toward Usability Problem Identification Based on User Emotions Derived from Facial Expressions

Jan Ole Johanssen, Jan Philip Bernius, and Bernd Bruegge
Department of Informatics
Technical University of Munich
Garching bei München, Germany
jan.johanssen@tum.de, janphilip.bernius@tum.de, bruegge@in.tum.de

***Abstract*—Tracking down usability problems poses a challenge for developers since users rarely report explicit feedback without being asked for it. Hence, implicit feedback represents a valuable information source, in particular for rapid development processes with frequent software releases. Users' emotions expressed by their facial expressions during interactions with the application can act as the source of such information. Recent development in consumer hardware offers mechanisms to efficiently detect facial expressions. We developed a framework for interactive mobile applications to harness consumer hardware camera technology for facial feature extraction to enable emotion detection following the facial action coding system. In a study with 12 participants, we evaluated its performance within a sample application that was seeded with usability problems. A qualitative analysis of the study results indicates that the framework is applicable for detecting user emotions from facial expressions. A quantitative analysis shows that emotional responses can be detected in three out of four cases and that they relate to usability problems. We conclude that, in combination with interaction events, the framework can support developers in the exploration of usability problems in interactive applications.**

***Index Terms*—user, user feedback, usability problem, usability evaluation, emotion, FACS, action unit, consumer hardware, continuous software engineering, framework, tool support.**

I. Introduction

During software evolution, developers rely on user feedback collected from various sources, such as from software distribution systems [1]–[3] or social media platforms [4], [5], to improve applications according to user needs. Developers collect and evaluate written user feedback: user comments or bug reports represent *explicit* knowledge as they formalize the knowledge themselves [6], [7]. Explicit feedback is not always accurate and can be incomplete: Users might not remember problems [8]—usability problems in particular. Further, dissatisfied users provide more feedback than satisfied users [9].

User feedback is important, especially in software development processes in which increments are frequently improved, such as in continuous software engineering (CSE) [10], [11]. Hence, other forms of user feedback have gained relevance: *implicit* user feedback, i.e., data monitored during users' interactions, promises to be a valuable information source.

User behavior data support elicitation of requirements [12]. Biometric measurements can be applied to determine emotional awareness [13], while the results can be useful for user interface assessment [8]. Facial expressions represent another measurement to derive user emotions during application usage.

Current infrared three-dimensional (3D) cameras have made face recognition technology available to a growing audience in their daily life[1]. In contrast to previous approaches, which usually relied on external, stationary equipment [14] that hinders its application in the everyday life of users, this advancement allows deriving user emotions from their facial expressions *in-situ* on mobile devices in their target environment. The collection of person-related characteristics using consumer hardware has been shown to be a viable concept [15]. Based on this finding, we attempt to identify a relationship between observed emotional responses by users and usability problems in mobile applications using consumer hardware.

We developed *EmotionKit*, a framework for deriving user emotions and relating them to user interface events. It harnesses consumer hardware camera technology for facial feature extraction. By using a list of common facial expressions [16], [17], EmotionKit does not require a machine learning approach to continuously calculate emotional measurements.

In a user study with 12 participants, we evaluated the applicability of EmotionKit and investigated relationships between observed emotions and usability problems in interactive mobile applications. Based on the results, we suggest that the user emotions should be collected and processed in combination with other knowledge sources, such as user interaction events. Then, user emotions can help identify usability problems. The overall vision is to integrate and visualize the results with other knowledge sources [18], [19]: The collection and processing of automatically collected user feedback can support developers in frequently and rapidly improving software increments during CSE, with the emotional response acting as one input source that is continuously provided from users.

This paper is structured as follows. Section II addresses the foundations of emotions. In Section III, we detail the technical aspects of the EmotionKit framework. We describe the study approach in Section IV, followed by a qualitative and quantitative analysis of the study's observations in Sections V and VI. In Section VII, we present an interpretation of results, plans for future work, a discussion of privacy aspects, and how EmotionKit can be used by developers. Related work is outlined Section VIII. Section IX concludes the paper.

[1] Apple Support Documentation: *About Face ID advanced technology*. November 2018. Available online: https://support.apple.com/en-us/HT208108.

Definitive version available at https://doi.org/10.1109/SEmotion.2019.00008

Accepted at the IEEE/ACM 4th International Workshop on Emotion Awareness in Software Engineering (SEmotion 2019), Copyright IEEE

II. Foundations

Emotions are important for understanding human intelligence, decision-making, and social interaction [20]. Literature is classified into two theories describing the term *emotion*: First, basic emotions refer to a list of the terms anger, disgust, fear, joy, sadness, and surprise [21]–[23]. Second, the dimensional theory describes emotions using multiple dimensions [24], [25]. *Activation*, *valence*, and *ambition* are the most commonly used dimensions [26]. In this work, we follow the basic emotions theory. A physiological response (or physical reaction to a stimulus) often has a direct impact on emotions [21], [27], [28]. Emotions can be described as a "mental experience" [23], and also as "neuromuscular activity of the face" [17]. Humans show their emotions in implicit and explicit ways, including facial expressions, speech, or body language [29]–[31]. Expression of emotions is, to some extent, specific to the spoken language and culture [29].

Changes in facial expressions—and therefore in facial features—can be described using the facial action coding system (FACS): it encompasses a set of 44 action units (AUs) that define a state of facial features or contraction of facial muscles [16], [31], [32]. It allows the description of thousands of possible facial expressions as a set of a few AUs [33].

The accurate judgment of human emotions is possible from the examination of facial behavior by observers [21], [22]. Since we aim to employ automated techniques, we are interested in the accuracy of measurements of facial components to derive emotions [22]. As emotions can be identified via facial expressions, it is possible to map a set of AUs to emotions. A system defining such a mapping is the emotional facial action system (EMFACS) [31], [34]. Most people cannot control all of their facial expressions arbitrarily [21]. Edge cases further complicate the detection of emotions and can only be addressed by analyzing muscle movement over time rather than through still photographs [30].

Almaliki *et al.* arrived at the conclusion that users do not like to provide feedback on the software [35]. The feedback collection process needs to adapt to fit users' individual needs to be successful, and the process should match with users' behavior at the time of providing feedback [35].

III. The EmotionKit Framework

EmotionKit is a framework for converting facial expressions into emotions. Figure 1 outlines the prototypical implementation that was developed as an Open Source[2] Apple iOS framework that leverages Apple's *ARKit*. The implementation follows a funnel logic to reduce ARKit measurements to emotions and is separated into three major stages.

a) Extracting facial expressions from ARKit: The first stage collects facial data using a *TrueDepth* camera. ARKit handles face detection and extraction of facial features. We can use the *ARAnchors* to retrieve facial data for each detected face. These data are stored and passed on to the next stage.

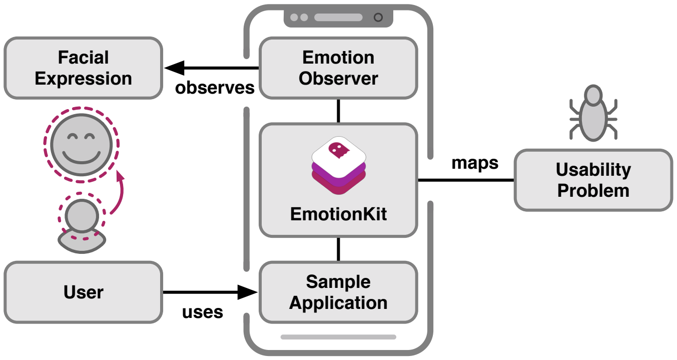

Fig. 1. A high-level overview of the EmotionKit framework that can be included in an interactive mobile application. It utilizes an emotion observer built into the smartphone to observe the users' facial expressions while they are facing the application for interaction. Using the knowledge extracted from the recognized emotions, it enables the mapping of usability problems to the software version that is currently used by the user.

b) Conversion to FACS Action Units: During the second stage, the ARKit data, the so-called *BlendShapeLocation*s, is translated into AUs. This reduces the number of data points from 52 down to 22[3]. While Apple does not refer to AUs in its documentation, our mapping suggests that their *BlendShapeLocation*s are inspired by the FACS system. Meanwhile, Apple does differentiate between left and right face movements, so most action unit conversions return the stronger sides shaping in our implementation. Three AUs are described best by a combination of two shapes, which is why we use their average.

c) Conversion to Emotions following EMFACS: We use the EMFACS system to map AUs to emotions. Given this relation and the AUs computed in the previous stage, we can calculate the probability for each emotion. As we follow this universal definition of emotions, we do require neither a training step in advance, nor computation intensive classification tasks. EmotionKit calculates the averages for all AUs, thereby making it possible to create probabilities for the seven emotions defined within the EMFACS system.

IV. User Study

This section describes the study approach to validate the applicability and reliability of EmotionKit.

A. Sample Application

As shown in Fig. 2, we created an application that displays six static views and each of them includes a usability problem derived from usability heuristics [9].

For the static **content** of the application, we chose a topic that—to our understanding—does not trigger any particular emotion in a participant: the history of the Munich subway system. Each line (U1, U2, U3, U4, U5, and U6) is represented with one static view that displays textual informational about its past, such as the date of opening or specifics about its stops.[4] The length of the text was chosen to achieve a reading time of approximately 30 seconds to 1 minute per screen.

[2] EmotionKit is part of the *Continuous User Understanding* platform [36] that offers multiple tools to developers. The core classes are located at https://github.com/cures-hub/cures-cuu-sdk/tree/master/CUU/Classes/EmotionKit.

[3] FACS defines 28 AUs. Due to ARKit limitations, we only detect 22 AUs.

[4] Text extracted from https://en.wikipedia.org/wiki/Munich_U-Bahn

Definitive version available at https://doi.org/10.1109/SEmotion.2019.00008

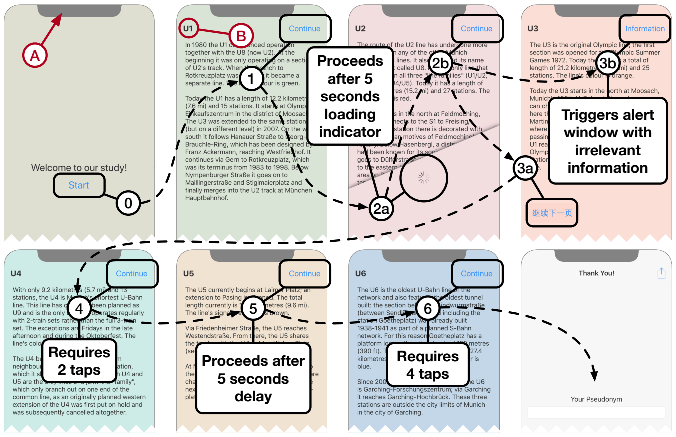

Fig. 2. In (A), the approximate location of the consumer device's *TrueDepth* camera is indicated. The sample application consists of eight views, while six views present static information about the Munich subway. The application is seeded with several usability problems, as defined by Nielsen [9] and highlighted in the following in *italic* font type. After the participant starts by tapping on (0), the U1 screen starts without any usability problems, while the subway line number is always presented in the top left corner (B). A tap on (1) brings the participant to the U2 screen which features *bad performance* with a loading indicator for 5 seconds (2a) before the actual text is displayed; (2b) triggers the next screen. The U3 screen features *inconsistency* (3a, 3b) by moving the continue button and using a Chinese text and also by adding a new button with different functionality, placed in the same spot used as on other views for the *continue* button. Button (3a) will allow the participant to navigate to the U4 screen which features *bad performance* and *non-functionality*, by making two taps necessary for the *continue* button (4). Hereafter, the U5 screen features *bad performance* and *missing feedback* by imposing a 5 seconds delay between the continue button tap (5) and the transition to the next view. Eventually, the U6 screen features the same problem as U4, this time requiring the *continue* button (6) to be tapped four times to work, following which the participant to will be taken to the finalization screen.

This time frame allowed us to make notes while the participants were busy reading the text. In addition, we attempted to obtain a neutral impression of the participants

For the application's dynamic **interactive** part, a *continue* button in the top right corner of the sample application serves the purpose of navigating to the next view. At the same time, this button reflected one of the major starting points for recording the usability problems.

B. Study Setting

The study was performed in a university seminar room. We prepared a *protocol* with the study procedure to ensure comparability between every session. We also prepared an *observation sheet* in which we collected observation notes.

In the protocol, we defined an introduction phase in which we welcomed the participants and explained their task. Hereafter, the participants started using the sample application equipped with EmotionKit. Two authors of this paper sat across the participants and acted as observers to note down any observation that they considered relevant, such as distinctive facial expressions. Some typical example notes were (a) *no reaction*, (b) *smiling*, (c) *hand in front of nose*, (d) *wondering face*, or (e) *twitching cheek muscle*. Mutual consensus after a short discussion was applied each time differences were recorded during merging of the notes.

C. Descriptive Data

We performed our study with 12 participants. All of them were either computer science students or academic staff. Each participant performed the same set of tasks and faced the same usability problems. The selection of the participants did not follow any rules. Participation was on a voluntary basis. Based on our impression from the contact with the participants throughout each study session, the existence of major confounding variables, such as the users' pre-experimental emotion or a stress situation, was ruled out. No further information about the participants was collected.

D. Threats to Validity

We derived the following non-exhaustive list of threats from the four aspects of validity by Runeson *et al.* [37].

1) Study Setting: The study was conducted in a laboratory environment. Participants were aware of the ongoing experiment. We tried to mitigate this threat by creating an atmosphere in which the participants did not feel observed.

2) Sample Size: The sample of participants represents a weakness of the study that potentially has an effect on the overall results. However, we strived for an exploratory approach in assessing and understanding the data to collect a first impression. Notably, a majority of usability problems can be derived from a comparatively small number of users [38].

3) Manual Observations: We did not measure pre- and post-experimental mood of the participants; for both aspects, we relied solely on manual observations. This approach allowed us to ensure and maintain comparability between participants. However, if we would have relied on individual reports of perceived emotions, we might have received incorrect, distorted, or biased results [8], [14].

4) Sample Application Design: The study application was designed to include a set of usability problems that do not guarantee the triggering of emotions in the participants. Interactions were limited to the simple navigation between static content views. We tried to mitigate this aspect by designing a sample application that comes close to the typical, text-based applications, using standard user interface elements.

V. Qualitative Analysis

Through a qualitative study analysis, we aimed to verify the possibility of detecting and recording user emotions from facial expressions using consumer hardware.

A. Data Processing

The sample application creates a log file that contains all recorded data. Each entry includes a timestamp, the current view within the application, and probabilities for the seven emotions calculated by EmotionKit. Actions performed by the user, e.g., a tap on a button, are recorded with their timestamp. Figure 3 shows a complete plot of the recorded time series for one participant. Its y-axis shows the change in emotion probability as described below in Equation 1.

$$y = \left| emotion - \overline{emotion} \right| \tag{1}$$

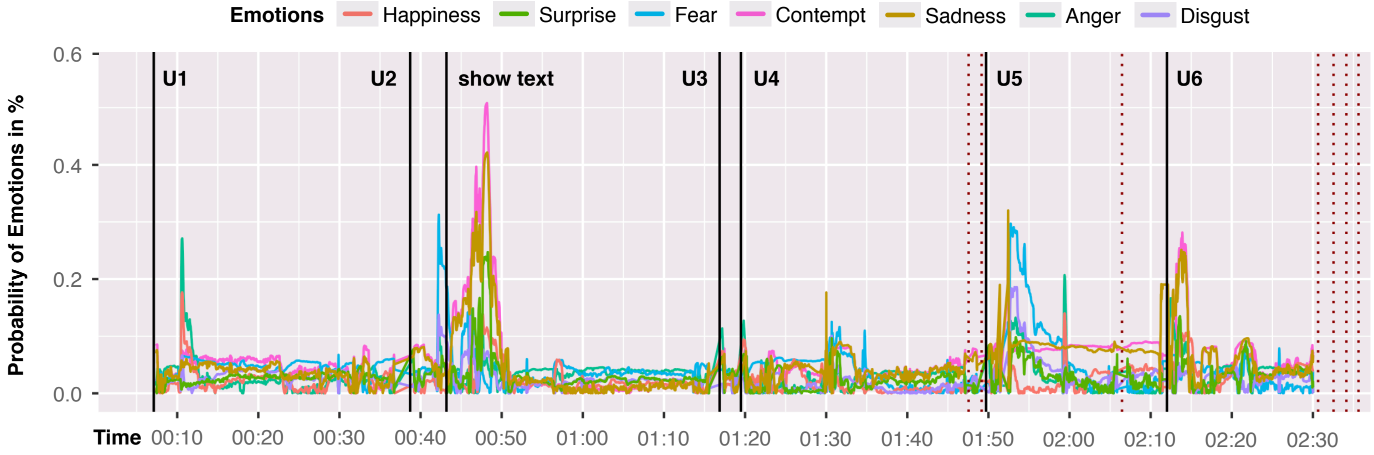

Fig. 3. Complete plot of observed emotions of one participant visualized over time on the x-axis. The appearance of a view is marked as a vertical line with the view name (`U1` to `U6`). The delayed appearance of text in `U2` view is highlighted the same way; user button taps are represented by red dotted lines.

B. Emotion Analysis

We compared the observers' notes with the recordings to analyze the emotion. After the transition to view `U2`, the participant was confused by the 5 seconds delay and appearance of the loading indicator. The participant expressed their confusion by saying "Ahh .. Ok". We observed *fear* near the end of the indicator and a strong expression of *contempt* and *sadness* after the text appeared. In a conversation after the experiment, the participant stated that they were confused since the application took a long time to load only text.

On view `U3`, the participant tapped the Chinese button right away and therefore skipped the view. EmotionKit recorded a combination of *anger*, *happiness*, and *sadness* on the transition to both views, `U3` and `U4`. In the observation notes, we noted that the participant appeared to be surprised. A change in *sadness* at `01:30` cannot be related to the manual observations.

The transition to view `U5` is followed by a change in *sadness* and *fear*. Our observation notes reveal outwards pull from the angles of the participants' mouth. A questioning hand gesture was noted. The last transition was not recorded, since the participant's hand covered the camera lens of the smartphone.

> **Observation 1.** Using EmotionKit, we are able to derive users' emotions that matched our manual observations.

Figure 3 exemplifies that the emotions were recorded with a time delay before and after the usability problem became visible to the participant, e.g., at `show text` or `U5`.

> **Observation 2.** In combination with user interface events, such as view changes or button taps, the observed emotions were identified and utilized to exploit usability problems.

VI. Quantitative Analysis

We performed a second, more detailed analysis of the recorded emotions to investigate quantitatively all participants' facial expressions toward the usability problems. By combining the individual emotions in one amplitude, we created a signature that reduced the noise sensitivity for the recordings.

A. Data Processing

As described in Equation 2, we sum up all seven emotions (happiness, sadness, surprise, anger, fear, disgust, and contempt) and derived a new value which—in contrast to Equation 1—can exceed 1.0. We refer to this value as the **emotional response**: The summation allows for a simplified identification of changes. In the emotional response, clear peaks can be observed, while the response during reading phases, i.e., phases with a neutral face, settles down to a steady and low level, generally reducing the noise as seen in Fig. 3.

$$y = \sum_{k=1}^{7} (|emotion - \overline{emotion}|)_k \tag{2}$$

B. Binary Classification

Based on the same data processing described in Section VI-A, we can describe the output of EmotionKit as the result of a binary classification to evaluate its performance:

TP is a true positive classification, in which the framework detects an emotional response that has also been recorded by the observers.

TN is a true negative classification, in which neither the framework nor the observers were able to detect an emotional response.

FP is a false positive classification, in which the framework detects an emotional response; however, the observers do not record an emotional response.

FN is a false negative classification in which the framework does not detect an emotional response; however, the observers do record an emotional response.

We relied on the observer notes to identify classes for the actual emotional responses. Subsequently, we manually created emotional response plots for every participant and read the emotional response to derive the predicted class. The outcomes of the two-class prediction are summarized in Fig. I, in which we follow the structure of the sample application introduced in Section IV-A.

Column titles ending with a "C" represent a static content view; e.g., **U1-C** describes results for screen of U1. Column titles ending with an "I" represent the application's interactive part containing the usability problem; e.g., **U1-I** describes the results for *bad performance* as denoted with (2a) in Fig. 2.

TABLE I
BINARY CLASSIFIER OUTCOMES OF THE EMOTIONKIT PERFORMANCE.

#	U1-C	U1-I	U2-C	U2-I	U3-C	U3-I	U4-C	U4-I	U5-C	U5-I	U6-C	U6-I
1	TP	TP	TN	FP	TP	TP	FP	TN	ERR	TP	FP	ERR
2	TP	TN	TN	TN	TP	TP	TN	TP	FP	TP	FP	TP
3	FP	TN	TN	TN	ERR	FP	TP	TP	FP	TP	TN	FP
4	TN	TP	TN	FP	TN	TP	TN	FP	TN	TP	TN	ERR
5	TN	FP	FP	FP	TN	TP	TP	TP	FP	FP	TN	TP
6	TN	FP	TN	FP	TN	TP	TN	FP	TN	FP	TN	TN
7	TN	TP	FP	TN	TN	TN	TP	TP	TN	TP	TN	TN
8	TN	FP	TN	FP	TP	FP	FP	TP	FP	TP	FP	TP
9	TP	TP	TN	TP	TP	TP	TN	TP	TN	TP	TP	TP
10	TP	TN	TN	ERR	TP	FN	TP	TN	FP	ERR	FP	ERR
11	ERR	FP	TP	TN	ERR	TP	TN	TN	TN	TP	TP	TP
12	TN	FP	TN	FP	TN	FP	TN	FP	TN	TP	TN	TP

Following this classification, we can describe the performance of EmotionKit following both the *Sensitivity* and *Specificity* values, as well as the *Accuracy* [39]: The sensitivity describes how many actual emotional responses have been correctly detected as an emotional response; this is known as the *recall* or the *true positive rate*. The specificity describes the number of occurrences in which a participant did not show any emotional response and that were detected as non-emotional response by EmotionKit; this is known as the *true negative rate*. The accuracy summarizes how many instances overall have been detected correctly. Table II lists the results split by a combined value as well as content and interaction parts only.

TABLE II
SENSITIVITY, SPECIFICITY, AND ACCURACY VALUES OF THE STUDY.

	Combined	**Content**	**Interaction**
Sensitivity	0.980	1.0	0.9706
Specificity	0.600	0.7308	0.3939
Accuracy	0.7407	0.7941	0.6866

EmotionKit detects the emotional responses of the participants (0.98 for the combined analysis; 1.0 for content; and 0.97 for interaction). Detecting non-emotional responses becomes more challenging as indicated by the results for specificity: Non-emotional responses are detected moderately for content (0.73), but detection during interaction is low (0.39). The accuracy states that three out of four instances of emotional responses are detected correctly in a combined scenario.

We observed 16 emotional responses and 38 non-emotional responses during the presentation of the static content views, while the participants responded with 34 emotional and 13 non-emotional responses to the interactive parts of the sample application. These data suggest that the study design follows its intention, i.e., participants are less influenced by the content and we are likely to observe the emotional responses related to the usability problems. The observation that 10 out of 12 participants in `U2-I` did not show any emotional response to the interaction itself eliminates the assumption that emotional responses are triggered by the transition—`U2-I` was the only interaction without any seeded usability problems.

More than one-fifth (23.61%, resp. 34 occurrences) of the observations were classified as `FP`. Two observers individually collected and mutually agreed to the observation notes, though they are not trained experts in reading emotions. Since the expressions are known to be sensitive to changes, probably many of the classified `FPs` may actually be `TPs`. However, since we detect more than two-third of correctly classified occurrences (69.44%, resp. 100 occurrences), we consider the false classification of `FPs` as a subject for further research.

> **Observation 3.** Participants show a higher emotional response toward the interactive- than to the static content; our data suggest that this behavior is related to the seeded usability problems. The occurrence of emotional responses can be detected better than the absence of emotional responses.

We classified some occurrences as an error and excluded them from the analysis. We defined an error as a situation in which EmotionKit could not record any data or the data was noisy because of a reason that was observed by the observers. While the total number of nine errors (6.25%) indicates the applicability of the approach, we assume that this number will be considerable higher in a real-world scenario, given more natural distractions and mobile-holding and -using positions. or generally other *external stimuli* [40].

VII. DISCUSSION

Our observations suggest that tracking down emotions from users' facial expressions can support the detection of usability problems. We discuss the impact for developers.

1) Understanding Emotions: Users react differently to the same usability problems. In the study, not all participants showed visible reactions to some or all problems, and in real-world scenarios, other factors, such as the cultural background, might influence users' facial expressions. Hence, developers are confronted with uncertainty when trying to understand the emotional reactions of multiple users to the same interactive element. We propose to analyze only individual emotional response graphs and look out for sudden changes for a short period of time, since such changes typically indicate unexpected software behavior. Furthermore, reading emotion requires psychology domain knowledge that a developer may lack; for example, a grin can be recorded as a combination of *contempt* and *sadness* as it becomes obvious from the plot in Fig. 3. We were able to understand the EmotionKit's recordings given the help of our manual observation sheets. As a long-term prospect and given an improved maturity level, EmotionKit might also support experts from other domains in understanding their users in respective tasks. This requires studies to confirm the sensor data reliability, explore parameters in the calibration process, and comparisons with other input sources, such as a front-facing mobile camera.

2) Creating Relations to Software Increment: Additional information about the application under observation is indispensable for interpreting the data. The recorded time stamps for the *view change* events, *show text* event, or the *user tap* events contributed considerably to the understanding of the observed emotion. Therefore, integrating EmotionKit with other knowledge sources can support developers to better understand the reason for a user emotion: Developers should be able to manually add events [41] at source code locations in which they expect usability problems. This can help analyze the emotional response. Furthermore, future research should address whether a particular usability engineering requirements can be map to a specific emotional response.

3) Automating Peak Detection: We assume that metrics about the observed emotions can be created to support the automatic identification of situations in which usability problems might have occurred. For example, similar to the examples given by Begel [42], a peak of *fear*, followed by either a peak of *contempt* and *sadness* may be treated as a sign for a *bad performance* issue. Similarly, tracking a different spectrum of emotions as suggested by Rozin and Cohen [43] may help notice the relevant changes in user feelings toward the software. The combination of metrics with an integration of EmotionKit into a CSE monitoring system will allow developers to benefit from automatic notifications as soon as emotional changes to their latest increments are detected.

4) User Data Collection: The collection of facial expressions and extraction of emotions is a highly sensitive process with consequences for user privacy. This becomes even more relevant as consumer hardware promises to improve sensor accuracy. EmotionKit should only be used with users who are aware that they are being recorded. Further, the developers themselves could act as an initial proxy for the user.

VIII. Related Work

Compared to other work addressing the process of extracting facial expressions from depth information [44], [45], EmotionKit relies on such data provided by existing components.

Feijó Filho *et al.* describe a "system to implement emotions logging in automated usability tests for mobile phones" [40] and augment its capability with information such as the users' location [46], [47]. As with EmotionKit, they utilize facial expressions to derive the emotions, however, they rely on images that are send to a remove server which decodes and interprets them using an emotion recognition software. In contrast, EmotionKit harnesses facial expressions recorded from a 3D depth camera and performs all calculations on the device. In addition, Feijó Filho *et al.* report on an initial evaluation with two subjects, focusing on positive and negative emotions [40], [46], [47], in which they report a successful application of the system; we present a study with 12 participants including a fine-grained analysis of results.

OpenFace 2.0 is a toolkit for facial behavior analysis [48]. It is capable of facial landmark detection, head pose estimation, eye gaze estimation, and facial expression recognition. The authors chose AUs as an objective way of describing facial models, an approach which is shared by EmotionKit. *FaceReader* applies vision based *fun-of-use* measurement [14]. The approach follows the FACS and EMFACS schema to detect emotions from facial models. FaceReader was evaluated in a controlled test environment. Subjects were video-taped and measurements compared with researchers' observations. The system can detect minor changes not noticed by the manual observation. McDuff *et al.* introduce a system for automatically recognizing facial expressions of online advertisement viewers [49]. They analyzed the response to internet advertisement videos. Viewers' faces were automatically feature coded following the FACS system, from which emotions were derived. Although we have applied the FACS system as well, there are some major differences. First, our focus lies on detecting usability problems in mobile applications. Second, rather than relying on the footage from a webcam, we base our emotion recognition on the output of a 3D camera.

Emotion recognition on consumer hardware becomes more available with commercial frameworks and services for both on-device[5] and online[6] emotion recognition. In contrast to our approach, these frameworks rely on the processing and evaluation of two-dimensional images. Furthermore, this work specifically focuses on finding usability problems.

The availability of 3D cameras in consumer devices attracted the attention of developers. *Loki*[7] is a proof-of-concept for emotion-targeted content delivery. The developers use the same input parameters as in this work, however, rely on a machine learning approach to derive the emotions from the facial features. In contrast, EmotionKit uses an established psychological concept to derive the users' emotions in mobile applications without the need for an initial training process.

IX. Conclusion

During rapid and frequent development processes such as CSE, implicit user feedback is required to improve the usability of software under development. We presented EmotionKit, a framework to detect emotions for usability problem identification that relies on EMFACS to understand facial expressions without the need for machine learning classifiers.

We explored the applicability and reliability of EmotionKit in a user study with 12 participants. In a qualitative analysis, we show that users react visibly to usability problems and that these reactions can be recorded using EmotionKit. The results suggest that a context in form of user interactions is required to fully understand the flow of events in relation to recorded emotions. In a quantitative analysis, we found that there is a higher emotional response toward interactive parts than static content, while the former related to the usability problems.

Acknowledgment. This work was supported by the DFG (German Research Foundation) under the Priority Programme SPP1593: Design For Future – Managed Software Evolution (CURES project). We thank the participants for their time and willingness in participating in the laboratory study.

[5] https://www.affectiva.com/product/emotion-sdk/

[6] https://azure.microsoft.com/en-us/services/cognitive-services/emotion/

[7] https://github.com/nwhacks-loki/loki

Definitive version available at https://doi.org/10.1109/SEmotion.2019.00008

References

[1] L. V. G. Carreño and K. Winbladh, "Analysis of user comments: An approach for software requirements evolution," in *2013 35th International Conference on Software Engineering (ICSE)*, May 2013, pp. 582–591.

[2] D. Pagano and B. Bruegge, "User involvement in software evolution practice: A case study," in *2013 35th International Conference on Software Engineering (ICSE)*, May 2013, pp. 953–962.

[3] S. Panichella, A. D. Sorbo, E. Guzman, C. A. Visaggio, G. Canfora, and H. C. Gall, "How can i improve my app? classifying user reviews for software maintenance and evolution," in *2015 IEEE Int. Conference on Software Maintenance and Evolution (ICSME)*, Sept 2015, pp. 281–290.

[4] E. Guzman, M. Ibrahim, and M. Glinz, "A little bird told me: Mining tweets for requirements and software evolution," in *2017 IEEE 25th Int. Requirements Eng. Conference (RE)*, Sept 2017, pp. 11–20.

[5] ——, "Mining twitter messages for software evolution," in *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*, May 2017, pp. 283–284.

[6] H. Krcmar, *Informationsmanagement*. Springer, 2015.

[7] I. Nonaka and H. Takeuchi, *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press, 1995.

[8] A. Bruun, E. L.-C. Law, M. Heintz, and P. S. Eriksen, "Asserting real-time emotions through cued-recall: Is it valid?" in *Proc. of the 9th Nordic Conf. on Human-Computer Interaction*. ACM, 2016, pp. 37:1–37:10.

[9] J. Nielsen, *Usability Engineering*. Elsevier LTD, Oxford, 1994.

[10] J. Bosch, *Continuous Software Engineering: An Introduction*. Springer, 2014.

[11] B. Fitzgerald and K.-J. Stol, "Continuous software engineering: A roadmap and agenda," *Journal of Systems and Software*, vol. 123, pp. 176–189, 2017.

[12] L. Liu, Q. Zhou, J. Liu, and Z. Cao, "Requirements cybernetics: Elicitation based on user behavioral data," *Journal of Systems and Software*, vol. 124, pp. 187 – 194, 2017.

[13] A. Fountaine and B. Sharif, "Emotional awareness in software development: Theory and measurement," in *2017 IEEE/ACM 2nd Int. Workshop on Emotion Awareness in Software Engineering*, May 2017, pp. 28–31.

[14] B. Zaman and T. Shrimpton-Smith, "The facereader: Measuring instant fun of use," in *Proceedings of the 4th Nordic Conference on Human-computer Interaction: Changing Roles*, ser. NordiCHI '06. New York, NY, USA: ACM, 2006, pp. 457–460.

[15] F. Schaule, J. O. Johanssen, B. Bruegge, and V. Loftness, "Employing consumer wearables to detect office workers' cognitive load for interruption management," *The PACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 1, pp. 32:1–32:20, Mar. 2018.

[16] P. Ekman and W. V. Friesen, *Facial Action Coding System*. Consulting Psychologists Press, 1978, no. Bd. 1.

[17] J. A. Russell and J. M. Fernández-Dols, *The Psychology of Facial Expression*. CAMBRIDGE UNIV PR, 2002.

[18] J. O. Johanssen, A. Kleebaum, B. Bruegge, and B. Paech, "Towards a systematic approach to integrate usage and decision knowledge in continuous software engineering," in *Proceedings of the 2nd Workshop on Continuous Software Engineering co-located with Software Engineering (SE 2017)*, Hannover, Germany, Feb. 2017, pp. 7–11.

[19] ——, "Towards the visualization of usage and decision knowledge in continuous software engineering," in *2017 IEEE Working Conference on Software Visualization (VISSOFT)*, Sept 2017, pp. 104–108.

[20] R. W. Picard, "Affective computing," *M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 321*, 1995.

[21] C. Darwin, *The Expression of Emotion in Man and Animals*. Project Gutenberg, 1872.

[22] P. Ekman, W. V. Friesen, and P. Ellsworth, *Emotion in the Human Face: Guidelines for Research and an Integration of Findings (General Psychology)*. Pergamon Press, 1972.

[23] M. Cabanac, "What is emotion?" *Behavioural Processes*, vol. 60, no. 2, pp. 69–83, nov 2002.

[24] W. M. Wundt, *System der Philosophie*, 2nd ed. W. Engelmann, 1897.

[25] H. Schlosberg, "The description of facial expressions in terms of two dimensions." *Journal of Experimental Psychology*, vol. 44, no. 4, pp. 229 – 237, 1952.

[26] ——, "Three dimensions of emotion," *Psychological Review*, vol. 61, no. 2, pp. 81 – 88, 1954.

[27] W. James, "II.—what is an emotion?" *Mind*, vol. os-IX, no. 34, pp. 188–205, 1884.

[28] R. Plutchik, "A psychoevolutionary theory of emotions," *Social Science Information*, vol. 21, no. 4-5, pp. 529–553, jul 1982.

[29] E. M. Albornoz and D. H. Milone, "Emotion recognition in never-seen languages using a novel ensemble method with emotion profiles," *IEEE Transactions on Affective Computing*, vol. 8, no. 1, pp. 43–53, Jan 2017.

[30] P. Ekman, W. V. Friesen, P. Ellsworth, A. P. Goldstein, and L. Krasner, *Emotion in the Human Face: Guidelines for Research and an Integration of Findings*, ser. Pergamon general psychology series. Elsevier Science, 2013.

[31] S. Z. Li and A. K. Jain, Eds., *Handbook of Face Recognition*. Springer London, 2011.

[32] M. Pantic, *Facial Expression Recognition*, 1st ed. Boston, MA: Springer US, 2009, ch. F, pp. 400–400.

[33] Z. Zeng, M. Pantic, and T. S. Huang, *Emotion Recognition based on Multimodal Information*. Springer, 2009, pp. 241–266.

[34] M. Lewis, J. M. Haviland-Jones, and L. F. Barrett, Eds., *Handbook of Emotions*, third edition ed. New York, NY: The Guilford Press, 2010.

[35] M. Almaliki, C. Ncube, and R. Ali, "The design of adaptive acquisition of users feedback: An empirical study," in *2014 IEEE Eighth International Conference on Research Challenges in Information Science (RCIS)*, May 2014, pp. 1–12.

[36] J. O. Johanssen, "Continuous user understanding for the evolution of interactive systems," in *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, ser. EICS '18. ACM, 2018, pp. 15:1–15:6.

[37] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*. John Wiley & Sons, 2012.

[38] J. Nielsen and T. K. Landauer, "A mathematical model of the finding of usability problems," in *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, ser. CHI '93. New York, NY, USA: ACM, 1993, pp. 206–213.

[39] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques (Third Edition)*, third edition ed. Boston: Morgan Kaufmann, 2011.

[40] J. Feijó Filho, T. Valle, and W. Prata, "Automated usability tests for mobile devices through live emotions logging," in *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. ACM, 2015, pp. 636–643.

[41] J. O. Johanssen, A. Kleebaum, B. Bruegge, and B. Paech, "Feature Crumbs: Adapting Usage Monitoring to Continuous Software Engineering," in *19th International Conference on Product-Focused Software Process Improvement.*, M. Kuhrmann, K. Schneider, D. Pfahl, S. Amasaki, M. Ciolkowski, R. Hebig, P. Tell, J. Klünder, and S. Küpper, Eds. Cham: Springer International Publishing, 2018, pp. 263–271.

[42] A. Begel, "Invited talk: Fun with software developers and biometrics," in *2016 IEEE/ACM 1st International Workshop on Emotional Awareness in Software Engineering (SEmotion)*, May 2016, pp. 1–2.

[43] P. Rozin and A. B. Cohen, "High frequency of facial expressions corresponding to confusion, concentration, and worry in an analysis of naturally occurring facial expressions of americans." *Emotion*, vol. 3, no. 1, pp. 68–75, 2003.

[44] T. Shen, H. Fu, J. Chen, W. K. Yu, C. Y. Lau, W. L. Lo, and Z. Chi, "Facial expression recognition using depth map estimation of light field camera," in *2016 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, Aug 2016, pp. 1–4.

[45] M. Z. Uddin, "Facial expression recognition using depth information and spatiotemporal features," in *2016 18th International Conference on Advanced Communication Technology (ICACT)*, Jan 2016, pp. 726–731.

[46] J. Feijó Filho, W. Prata, and J. Oliveira, "Affective-ready, contextual and automated usability test for mobile software," in *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. ACM, 2016, pp. 638–644.

[47] ——, "Where-how-what am i feeling: User context logging in automated usability tests for mobile software," in *Design, User Experience, and Usability: Technological Contexts*, A. Marcus, Ed. Springer International Publishing, 2016, pp. 14–23.

[48] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, "Openface 2.0: Facial behavior analysis toolkit," in *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, May 2018, pp. 59–66.

[49] D. McDuff, R. E. Kaliouby, J. F. Cohn, and R. W. Picard, "Predicting ad liking and purchase intent: Large-scale analysis of facial responses to ads," *IEEE Transactions on Affective Computing*, vol. 6, no. 3, pp. 223–235, July 2015.

Definitive version available at https://doi.org/10.1109/SEmotion.2019.00008