PREPRINT - Accepted at the 10th International Conference on Affective Computing and Intelligent Interaction (ACII 2022)

Sensor-Based Emotion Recognition in Software Development: Facial Expressions as Gold Standard

Nicole Novielli, Daniela Grassi, Filippo Lanubile
University of Bari
Bari, Italy
{nicole.novielli, filippo.lanubile}@uniba.it, d.grassi9@studenti.uniba.it

Alexander Serebrenik
Eindhoven University of Technology
Eindhoven, The Netherlands
a.serebrenik@tue.nl

***Abstract*—Early identification of emotions of software developers can enable timely intervention in order to support developers' well-being and prevent burnout. We present a machine learning experiment aimed at recognizing emotions during programming tasks using wearable biometric sensors, tracking electrodermal activity and heart-related metrics. As a gold standard for supervised learning, we rely on a state-of-the-art tool for emotion recognition based on facial expression analysis. We design, implement and evaluate an approach that combines the output of two classifiers for neutral valence recognition and positive/negative polarity classification. Our findings suggest that biometric sensors in a wristband can be used to identify emotions whose recognition would otherwise need an intrusive webcam.**

***Index Terms*—Emotion awareness, emotion detection, biometric sensors, empirical software engineering, human factors, facial expression analysis**

I. Introduction

Recent years have seen an increasing interest in investigating the role played by emotions on software engineers' productivity and well-being [1]–[7]. Indeed, software development is an intellectual activity requiring creativity and problem-solving skills, which are known to be influenced by emotions [8]. In recent years, companies have adopted strategies to support emotion awareness [9]–[11], e.g., by enriching agile retrospective meetings with self-reported information on perceived emotions so as to better identify what are the activities and events associated with them. However, self-report might provide a partial or biased perspective because the communication of emotions might be influenced by cognitive processing as well as by emotion regulation tendencies [12], [13]. Along this line, we envisage the adoption of sensor-based emotion recognition as a way to enrich emotion awareness beyond self-reporting, both at an individual and the team level.

Emotion recognition through facial expression analysis has reached the required level of maturity for commercialization. State-of-the-art approaches are implemented in commercial tools such as Affectiva[1] or Face Reader,[2] available at accessible costs. However, using facial recognition involves capturing video through webcam, which might be perceived as highly intrusive by developers. Hence we ask:

RQ *To what extent can we use biometric sensors embedded in non-invasive wearable devices, such as wristbands, as a proxy for emotions that would be otherwise recognized through facial expression analysis during programming tasks?*

To address our research question, we performed a study with 23 participants engaged in a programming tasks while wearing a wristband equipped with biometric sensors. We used a webcam to capture the participants' emotions based on the analysis of their facial expressions, which we used as a gold standard for training a sensor-based supervised classifier for emotional valence, e.g. the positive, negative, or neutral pleasantness of the emotion stimulus.

As a main contribution, this is the first study proposing facial expression analysis as a gold standard to train a sensor-based emotion classifier for software development. Our findings complements empirical evidence provided by previous research investigating approaches to sensor-based emotion detection using developers' self-reported emotions as gold standard. Specifically, we experiment with different experimental settings to provide a better understanding of the impact of several factors, – including imbalanced train data and the choice of algorithm for machine learning – on the emotion recognition performance. As a further contribution, we release a lab package to replicate and build upon this study.

II. Background

A. Sensor-based Emotion Recognition

The link between emotions and physiological feedback—measured using biometric sensors—has been widely investigated in the field of affective computing [14]–[16] demonstrating association of several physiological measures with emotions. Changes in the electrical activity of the brain (EEG) act as a successful predictor for pleasantness of the emotion stimulus [16], [17] and can be used to identify discrete emotions, such as happiness and sadness [18]. The electrical activity of the skin (EDA) is an indicator for emotional intensity [19] and has been used for identifying excitement, stress, interest, attention as well as anxiety and frustration [20], [21]. Heart-related metrics such as blood volume pressure (BVP), heart rate (HR) and its variability (HRV) have been successfully used for emotion detection [22], [23]. Facial electromyography (EMG) was also used [15], [24], as it captures the movements of facial muscles due to facial expressions of emotions.

In recent years, software engineering research has investigated the feasibility of emotion detection using lightweight

[1] www.affectiva.com
[2] www.noldus.com/facereader

biometric sensors that can be comfortably worn in a natural setting such as the work environment [2]–[4], [25]. In our study we consider EDA, BVP, and HR metrics as they can be collected using low-cost non-invasive sensors [3], [4], [25], [26] that can be comfortably used by developers during programming tasks (see Section III-A). This choice is in line with current research investigating the use of lightweight biometric sensor for emotion recognition in software development. Müller and Fritz [3] attempted to recognize emotional valence of 17 programmers. They use self-report collected while coding as gold-standard to train a sensor-based classifier able to distinguish between positive and negative emotions with an accuracy of .71. They use multiple sensors including EEG, EDA, HR, and eye tracking metrics. Along the same line, Girardi et al. [2] use biometrics to classify developers emotional valence and arousal. They trained two supervised classifiers for valence and arousal using as a gold standard the emotions self-reported by the participants during a Java programming task. They identify a minimum set of sensors including EDA, BVP, and HR measured using the Empatica E4 wristband, which they also employ in a field study [4].

B. Facial Expression Analysis for Emotion Recognition

Facial expression analysis (FEA) studies the task of analysing facial expressions to infer affective and cognitive mental states. FEA models facial expression using Facial Action Coding System (FACS), originally defined by Ekman [27]. FACS defines the facial expressions by coding them into small action units corresponding to facial muscles that usually contribute to create the various expression patterns associated to the emotion experienced by an individual [28]–[30]. Nowadays, this research field has reached maturity and available tools are capable to reliably identify people's emotions based on FACS analysis [31].

As for software engineering, recent studies proposed to use FACS-based emotion recognition for usability studies. Johanssen et al. [32] propose *EmotionKit*, a framework for identification of user emotions through facial expression analysis that interprets negative emotion as an indication of usability problems. Schmitd at al. [33] investigate the relationship between emotions and usability metrics. Specifically, they perform a study in a multi-modal setting where emotions are assessed through text-based sentiment analysis, speech-based emotion analysis, and FACS-based emotion analysis using OpenFace [34], an open source tool for face analysis. Filho et al. [35] propose an approach to automated usability tests for mobile devices that leverages live emotion logging using the device front camera. To recognize the user emotions they rely on the Intel RealSense SDK.[3] To the best of our knowledge, this is the first study proposing to use the facial emotion recognition as a gold standard for training a classifier of software developers' emotions while programming. Among the tools currently available, in this study we rely on emotion labels provided by Affectiva [28], which is widely used for labeling facial expressions of emotions [36].

[3]https://software.intel.com/en-us/intel-realsense-sdk

III. Study Design

A. Instrumentation and Participants

Development Task. The task, originally designed by Mueller and Fritz [3], consisted in implementing a Java program to retrieve all answers posted by a StackOverflow user and sum up the scores the user earned for them. Participants could use the StackExchange API[4] and were provided with a skeleton code to start with.

Measurement Tools and Devices. We use the Empatica E4 wristband to collect EDA- and hearth-related biometrics, a webcam to collect the video of participants during the programming task, and Affectiva for emotion recognition using facial expression analysis.[5] The choice of using a minimal set of sensors, i.e., the EDA sensor and the plethysmograph for heart-related metrics embedded in the E4 device, is justified by the results of previous work identifying this as the minimal set of non-invasive biometric sensors for reliable emotion recognition while programming [2]. Empatica E4 measures EDA with a sample frequency of 4Hz. It features a plethysmograph for collecting BVP sampled at a frequency of 64Hz. BVP is used to derive the HR and HRV. Following the Empatica guidelines,[6] we excluded HRV as it is unreliable in dynamic conditions (i.e., when typing).

In line with our long-term research goal, i.e., enabling the early recognition of negative feelings impairing software engineers' well-being, we focus on the *valence* of emotions, i.e. the (un)pleasantness of the emotion stimulus [37]. Affectiva uses the Facial Action Coding Systems and implements a supervised classifier based on deep-learning, trained on million data points extracted from webcam videos recorded in real-world conditions [28], [36]. It takes as input raw videos and provides as output, for each second of the video, a set a score for valence in the range $[-100, 100]$ as well as a timestamp which we can be used to synchronize with other data sources such as the raw signal collected by the biometric sensors. At the time of the study, Affectiva offered a free six-months license for academia.

Participants. We recruited 23 CS students following a convenience sampling strategy [38] by inviting participants from a pool of volunteers. We recruited only volunteers that could provide evidence they cleared exams where Java programming was used for capstone projects. By doing so, we could mitigate any threats to validity due to the lack of familiarity with the language adopted for the programming task.

B. Experimental Protocol

The experimental protocol is organized in four phases.

Pre-experimental briefing. The experimenter invites the participant to enter the laboratory, sit in a comfortable position

[4]https://api.stackexchange.com

[5]www.affectiva.com

[6]https://support.empatica.com/hc/en-us/articles/360030058011-E4-data-IBI-expected-signal

and adjust the monitor height. The experimenter summarizes the study steps and explains the programming task. Then, the participant is invited to sign the consent form to allow anonymous treatment of the data collected through the biometric sensors and the webcam.

Acquisition of neutral baseline. The participant wears the Empatica E4 and the experimenter checks that the wristband correctly acquires and record the raw EDA and heart-related signals. Then, we collect the participant's biometrics in a neutral condition. To this aim, the participant watches a 2-minute relaxing video of a nature scenery capable of inducing relaxation and a neutral emotional state [39]. The raw biometrics collected in the neutral emotional condition, i.e. in absence of emotions, are used for normalization of the biometrics during the preprocessing, for each participant (see Section IV-B). By doing so, we mitigate the threats to validity due to the differences between participants' biometrics that might affect the performance of the emotion classifiers, in line with previous research [2], [40]

Software development task. The core of the study is a 30-minute coding session. We collect the subjects' biometrics and record their videos while programming. Upon completing the coding task, the participants are invited to watch again the relaxing video to ward-off possible induced negative emotions that could arise, for example, from not being able to solve the programming task. At the end of the experiment, we award the participants a meal voucher.

Data Cleaning. Once the experiment is completed, but before analyzing the data, we check the quality of the collected data. We manually search for possible malfunctioning of the wristband that could have introduced noise. Also, we check for any possible discontinuity in the raw data due to interruption of the recording by the device. Furthermore, we discard those participants for which we could not rely on a good quality for the video. From the original pool of 23 participants, we removed three videos. As a result, after this data-cleaning step, our dataset includes 20 raw videos that we used for study described in the following (see Section IV-C).

IV. Machine Learning

A. Gold Standard Dataset

After the data-cleaning step described in the previous Section, our initial dataset included 20 raw videos. In fact for two participants, the quality of the video captured by the webcam was not good enough to ensure a reliable labeling of emotions because their face was only partially captured as the participants changed their position after the initial setting. A third participant was discarded because we could not synchronize the webcam and sensors timestamps.

For each participant, we manually isolated the part of the raw video regarding the programming tasks by removing the initial part including the briefing and the acquisition of the neutral baseline. Thus, we ran Affectiva to obtain a valence score for each second of the videos recorded during the programming task. We used the Affectiva SDK for analysing our videos [41]. The only limit imposed by the tool was the overall duration of the input raw videos that could be no longer than 3 minutes. As such, we segmented the raw videos in chunks of 3 minutes each, using a video editor. We discretize the valence scores in order to distinguish between the *positive*, *negative*, and *neutral* valence. To this aim, we used the k-means clustering algorithm on the continuous valence scores as implemented by the *a-rules* R package [42], [43].

Whenever reusing machine-learning classifiers *off-the-shelf*, we might experience a drop in performance as we run them on new unseen data collected in different conditions with respect to the training set. Hence, we performed a basic check to assess the reliability and suitability of the valence annotation provided by Affectiva with respect to our research goals. To this aim, two researchers manually inspected a sample of videos from our dataset to check for correctness of valence labels. For each of the 20 participants, we extracted five 10-second video excerpts (100 segments overall). The two researchers independently verified the consistency of the valence label with the facial expression shown in the video. The two researchers achieved full agreement on this labeling task. As a result of this step, 8 videos were removed because of clear disagreement between Affectiva and human labels. Problems were due to the presence of elements disturbing the facial expression analysis such as dark glasses, long hair bangs covering the eyes or the eyebrows, and dark beard and mustache erroneously identified as an open mouth.

As a results of this step, the gold standard dataset finally includes videos of 12 participants corresponding to 17,710 labeled video-chunks of 1 second, of which 1,581 (9%) were labeled as *positive*, 1,420 (8%) as *negative*, and the remaining 14,709 (83%) items as *neutral*. We use these labels as gold standard for training our valence classifier and we align them with the biometrics measurements using the Empatica E4 timestamps.

Given the unbalanced distribution of the valence label, we experiment with different datasets resulting from random sampling neutral cases, as reported in Table I. Specifically, we consider the full dataset (the first line of Table I) as well as a balanced dataset (the second line). To further investigate the impact of the unbalanced data distribution on the training, we experiment with four additional settings with increasing number items for the neutral class. All the scripts used for the creation of the gold standard dataset and for sampling the neutral cases are included in our replication package.[7]

TABLE I
THE GOLD STANDARD DATASET WITH DISTRIBUTION OF VALENCE LABELS. *Line 1*: THE FULL DATASET DISTRIBUTION. LINES 2–6: VARIOUS ROUNDS OF THE MACHINE LEARNING STUDY.

Neutral	Positive	Negative	Total
14709 (83%)	*1581 (9%)*	*1420 (8%)*	*17710*
1400 (32%)	1581 (36%)	1420 (32%)	4401
2800 (48%)	1581 (27%)	1420 (24%)	5801
4200 (58%)	1581 (22%)	1420 (20%)	7201
5600 (65%)	1581 (18%)	1420 (17%)	8601
7000 (70%)	1581 (16%)	1420 (14%)	10001

[7]https://figshare.com/s/0dff61db72be2bde3292

B. Preprocessing and Features extraction

The raw biometric signals were recorded during the entire experimental session for all the participants. However, for the purpose of reliably capturing biometrics associated to emotional valence labels, we only consider the signals recorded in proximity of the stimuli of interest, i.e., the biometrics collected in the 10 seconds before the valence label is assigned by Affectiva. This choice is in line with consolidated practices in related research on sensor-based classification of affective states of software developers [2]–[4], although the aforementioned studies use self-reports as gold standard. To synchronize the measurement of the biometric signals with the labeled emotion, we (i) save the timestamp of the label provided by Affectiva (`t_label`), (ii) calculate the timestamp for relevant timeframe for each interruption—i.e., 10 seconds before the labeled frame (`t_start`), and (iii) select each signal samples recorded between `t_start` and `t_label`.

For each participant, we normalize the signals to their baseline calculated based on the last 30 seconds of the video used to elicit a neutral state before starting the task, in line with previous research [2], [40]. To maximize the signal information and reduce noise caused by movements, we applied multiple filtering techniques, by reusing the script distributed in the replication package by Girardi et al. [2]. Regarding BVP, we extract frequency bands using a band-pass filter algorithm at different intervals [22]. The EDA signal consists of a tonic component (i.e., the level of electrical conductivity of the skin) and a phasic one representing phasic changes in electrical conductivity or skin conductance response (SCR) [44]. We extract the two components using the cvxEDA algorithm [45].

TABLE II
MACHINE LEARNING FEATURES.

Signal	Features
EDA	- mean tonic - phasic AUC - phasic min, max, mean, sum peaks amplitudes
BVP	- min, max, sum peaks amplitudes - mean peak amplitude (diff. between baseline and task)
HR	-mean (diff . between baseline and task) - heart-rate variance (diff . between baseline and task)

After signal pre-processing, we extracted the features presented in Table II, which we use to train the classifier. We select features based on previous studies using the same signals [2]–[4], [26], [46] and reuse publicly available scripts for feature extraction [2].

C. Modeling

We experiment with a pipeline approach, depicted in Figure 1, which combines the output of two classifiers: the first one distinguishes neutral vs. non-neutral items. Then, the non-neutral items are provided in input to the second classifiers that distinguishes negative vs. positive polarity. Figure 1.b). The two classifiers in the pipeline approach are trained independently using the training set. In particular, for training the first classifier (neutral vs. non-neutral) the positive and negative items are mapped to the non-neutral class. We train the classifiers using supervised machine learning. Based on previous findings [2], [47], we select the two best-performing machine learning algorithms on sensor-based emotion recognition, that is Random Forest (RF) and Support Vector Machine (SVM).

We train and evaluate the pipeline for the valence classifier in several different settings corresponding to various combinations of different parameters. Specifically, we tested with different *train-test splits* (70-30,80-20,90-10), the two *machine learning algorithms* (RF and SVM), and different *size of the neutral class* including an increasing number of neutral valence cases, randomly sampled from the full dataset described in Section IV-A. To split the gold standard into train and test sets we used the stratified sampling strategy implemented in the *R* package `caret` [48]. For each setting, we search for the optimal hyper-parameters using 10-fold cross-validation The resulting model is then evaluated on the hold-out test set to assess its performance on unseen data. We repeat this process 10 times to further increase the validity of the results. The performance is evaluated by computing the mean for precision, recall, F-measure, over the different runs. This setting is directly comparable to the one implemented by Müller and Fritz [3] and Girardi et al. [2].

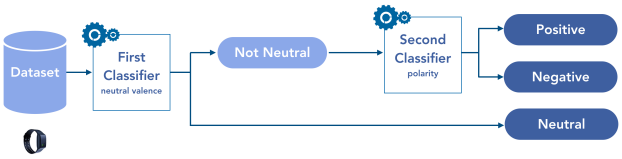

Fig. 1. The valence classifier as a pipeline of two supervised classifiers.

D. Evaluation Metrics

We assess the performance of our valence classifier in terms of precision, recall, and F1. Given the unbalanced distribution of the gold standard dataset, we rely on the macro-average i.e., precision and recall are first evaluated locally for each class, and then globally by averaging the results of the different categories. For comparison with related work, we also report the Accuracy, computed as the percentage of correctly classified cases. In line with previous research [49] we also compute the weighted kappa (κ) [50], [51] to distinguish between mild disagreement, that is the disagreement between negative/positive and neutral valence, and severe disagreement, that is the disagreement between positive and negative valence. We assign weight = 1 to mild disagreement cases and a weight = 2 to severe disagreement. Following a consolidated practice [51], we interpret κ as follows: κ values less or equal to zero indicate that agreement is less than chance; the agreement is slight if $0.01 \leq \kappa \leq 0.20$, fair if $0.21 \leq \kappa \leq 0.40$, moderate if $0.41 \leq \kappa \leq 0.60$, substantial if $0.61 \leq \kappa \leq 0.80$ and almost perfect if $0.81 \leq \kappa \leq 1$.

V. RESULTS

In Table III, we report the performance achieved in all settings by the valence classifier. Due to space constraint, we

only report the performance obtained with Random Forest as we observed it outperforms SVM in all settings. In line with consolidated research practice [2], we use as a baseline the trivial classifier that always predict the majority class, i.e., the classifier predicting *neutral* valence for all settings except the one with 1400 neutral cases, for which *positive* is the majority class. In Table IV we report the baseline performance.

TABLE III
PERFORMANCE OF THE CLASSIFIERS (BEST PERFORMANCE IN BOLD).

#neu.	Prec	Rec	F1	Acc.	κ	Perfect agreem.	Disagreement Severe	Disagreement Mild
Split: 70% train (10-fold) - 30% test								
1400	.61	.62	.60	.63	.54	63%	6%	31%
2800	.59	.60	.59	.60	.44	60%	3%	37%
4200	.61	.57	.59	.65	.4	65%	2%	33%
5600	.62	.54	.57	.69	.36	69%	1%	29%
7000	.62	.52	.55	.72	.33	72%	1%	27%
14709	.65	.41	.44	.84	.18	84%	<1%	16%
Split: 80% train (k-fold) - 20% test								
1400	.67	.66	.64	.67	.60	67%	5%	28%
2800	.61	.61	.61	.62	.46	62%	4%	35%
4200	.66	.62	.63	.69	.47	69%	1%	30%
5600	.63	.54	.57	.69	.37	69%	1%	30%
7000	.65	.52	.56	.73	.35	73%	1%	26%
14709	.61	.44	.47	.83	.22	83%	<1%	17%
Split: 90% train (k-fold) - 10% test								
1400	**.66**	**.67**	**.66**	**.68**	**.61**	**68%**	**4%**	**28%**
2800	.63	.64	.64	.64	.49	64%	3%	33%
4200	.61	.57	.59	.64	.40	64%	1%	35%
5600	.62	.55	.58	.68	.38	68%	1%	31%
7000	.68	.56	.60	.75	.41	75%	1%	24%
14709	.61	.44	.48	.83	.22	83%	<1%	17%

TABLE IV
MAJORITY CLASS BASELINES FOR EACH RUN.

# neutral	Prec	Rec	F1	# neutral	Prec	Rec	F1
1400	.12	.33	.18	5600	.19	.33	.25
2800	.16	.33	.22	7000	.22	.33	.26
4200	.19	.33	.25	14709	.23	.33	.27

In all settings and runs we observe a substantial improvements over the baseline classifiers (see Table IV). In particular, we obtain the best performance in the 90-10 split condition. The best performance (precision = .66, recall = .67, f1 = .66, accuracy = .68) is obtained using 1400 neutral cases. As for weighted κ, we observe a substantial agreement (κ = .61) between the gold label and the predictions of the pipeline classifier. A low percentage of severe disagreement (4%) is observed, which indicates that confusion between positive and negative emotion rarely occurs, compared to confusion between neutral and positive or neutral and negative (mild disagreement = 28%).

To compare the performance of the two approaches, we provide the confusion matrix and the class-based performance for the best run (see Table V). In line with the mild and severe disagreement rates observed for the two best performing classifiers (see Table III), the performance by class suggests that the pipeline classifier is able to distinguish between positive from negative valence, thus rarely introducing a severe disagreement.

TABLE V
PERFORMANCE BY CLASS.

	Confusion matrix			Performance by class		
	Classifier prediction					
Class	neg	neu	pos	Prec	Rec	F1
neg	120 (85%)	14 (10%)	8 (6%)	.69	.85	.76
neu	45 (31%)	52 (37%)	43 (31%)	.58	.37	.45
pos	9 (6%)	23 (15%)	126 (80%)	.71	.80	.75

VI. DISCUSSION

A. Comparison with Related Studies

Valence classifier performance. Our study is built on top of two studies that used self-reports rather than facial expression analysis as a gold standard to classify developers' emotions [2], [3] along the valence dimension. Specifically, we use the same Java programming task used by Girardi et al [2] and originally defined by Müller and Fritz [3] and we leverage the Empatica E4 wristband used in their studies. Our best performing classifier achieves performance (f1 = .66 and accuracy = .68) comparable to the one achieved by the valence classifiers of the two studies. In particular, Müller and Fritz [3] report a classification accuracy of .71% while Girardi et al. [2] report an accuracy = .71 and f1 = .59). This is a very promising result especially considered that this performance is obtained by a classifiers that predict valence in $\{positive, neutral, negative\}$. Conversely, the two previous studies trained binary classifiers that were able to distinguish between positive and negative cases, without modeling the neutral condition in absence of emotions. The study of Girardi et al. [2] conducted in the ab setting has been further extended in a study of 21 professional developers at their workplace [4].

Richer sensor settings. The topic of sensing developers' emotions was also investigated by field studies involving professional developers at the workplace. Vrzakova et al. [25] leveraged EDA and eye gaze for classifying developers' valence and arousal of 37 developers performing code review during an in-situ experiment. They trained a supervised machine-learning classifiers considering features of each signal separately and features of all signals in combination. Again, as a ground truth, they relied on binarized self-reported scores for valence (positive vs. negative) and arousal (low vs. high). Their findings show that the eye gaze is the most predictive measurement for emotional valence, achieving accuracy of 85.8%. When combining both eye-gaze and EDA, authors achieve 90.7% accuracy for valence and 83.9% for arousal. We are not including measures based on eye-tracking because too much invasive for being considered in the workplace except for experimental purposes.

Need for individual training? Girardi et al. [4] investigated the use of sensors for classifying the self-reported emotions of 21 software developers from 5 different companies, as reported over a time span of 2 or 3 weeks, depending on the duration of the agile iteration. They attempt to distinguish negative from non-negative emotions based on EDA- and heart-related metrics using the Empatica E4 wristband. While achieving promising performance (f1=.75 for the best run), their valence

classifier reported lower average performance in the leave-one-subject-out condition (f1 = .46), i.e. its performance vary depending on the individual's biometrics. Thus, they suggest strengthening the approach through future replications involving further data collection and fine-tuning of emotion models on an individual basis, to account for differences in biometrics between study participants. Due to the restricted pool of participants we could not investigate further in such direction, which we plan to explore in future studies.

B. Implications

Sensor-based recognition of developers' emotion: self-report vs. facial emotion recognition. Findings from affective computing research suggest that multiple emotion assessment methods (e.g., self-report vs. recognition of emotions based on facial expressions) might not necessarily align at a particular moment in time [52]. In his study on the role of emotion in learning and cognitive development, Pekrun [12] describes the emotional as a coherent response among different components. At the cognitive level, the emotion is triggered by the assessment of a situation (i.e., worrying about something threatening my goals). At a physical level, emotions reflect in biometrics changes (e.g, EDA changes due to sweating and heart rate rising in presence of anxiety) and might be also visible through facial expressions.

In the context of software development, the self-reported emotions might be influenced by cognitive processing as well as by emotion regulation tendencies. It is the case of emotional labor, that is the "process by which workers are expected to manage their feelings in accordance with organizationally defined rules and guidelines" [53], which might reduce the disclosure of negative emotions considered not acceptable in collaborative software development [13].

To fully support emotion awareness during software development a combination of multiple approaches for emotion assessment is needed. Specifically, we envisage the emergence of tools and practices including both self-reporting through experience sample and emotion recognition through facial expression analysis, as they might provide complementary information on the emotional status of an individual. However, this could not be always feasible in practice: continuously asking to self-report emotions might be perceived as annoying and might interfere with daily activities while video recording is problematic in terms of privacy.

This study represents the first attempt to assess the emotions of software developers using facial expression analysis as a gold standard. Previous studies investigated and demonstrated the feasibility of sensor-based emotion detection using non-invasive biometric devices while programming [2], [3], code-review [25], or during a wider range of developers' daily activities performed at the workplace [4], [25]. What these studies have in common is that they rely on self-reported emotions as gold standard, i.e. such classifiers enable recognizing the emotions developers explicitly reported during the data collection stage. In this study, we advance and complement the state of the art by providing evidence that it is possible to build sensor-based classifiers to predict also the emotions that would be identified through facial analysis.

Supporting the developers' emotional awareness at the individual and at the team level. Happy developers solve problems better [54]. Recent research demonstrated that a relationship exists between developers' productivity and their well-being [55], [56]. Indeed, companies are recently implementing strategies to support emotion awareness [9], [10] of developers both at an individual and at the team level. For example, during agile retrospective meetings, developers could self-report their emotions and use them as a proxy for problems and triggers for discussion [11], [57].

Along the same line, we envision the emergence of tools that combine multiple approaches to support emotional awareness at the team level, e.g. by including the emotional feedback in agile meetings, as well as at the individual level, e.g. by identifying negative emotions and suggesting just-in-time corrective actions to restore positive mood and focus. The findings of our machine learning experiment suggest that biometrics can be used to identify emotions that would be recognized by a state-of-the-art tool for facial expression analysis. As such, we envision the adoption of sensor-based emotion classifiers based on wearable devices because less invasive than webcams and less annoying than self-reporting.

C. Threats to Validity

Construct validity. Our study suffers from threats to construct validity that is the reliability of our measures in capturing emotions. In this study, we employed lightweight, low-cost sensors in line with our long-term vision of enabling emotional awareness at the work place using sensors that are comfortable to wear during the developers' daily activities. This might have lowered the quality of data collected by the sensors with respect to those collected in a controlled setting as in previous lab studies in the affective computing field. To mitigate this threat, we performed a careful quality assessment of the collected data for all participants.

Internal validity. Threats to internal validity concern confounding factors that can influence the results. Factors existing in our laboratory settings, such as the absence of real consequences when failing or succeeding in the task compared to real task professional developers deal with while at work, can influence the triggered emotions. Also the choice of the programming task might have an impact on richness and variety of emotions experienced by the participants during the task. In line with our long-term goal of supporting emotion awareness and well being of software developers, the test should have been feasible but also difficult enough to elicit both positive and negative emotions. To ensure feasibility, we selected only volunteers that successfully cleared exams where Java was used for capstone projects.

Further threats might be due to the choice of the Affectiva tool for creating the gold standard. When re-using supervised classifiers, we should be aware that the classifiers are built for specific goals and the datasets might reflect different conceptualizations of affect. To mitigate this threat, two researcher

independently evaluate the correctness of the emotions predicted by Affectiva by manually inspecting a subset of labeled video frames. Finally, due to constraints on the input format posed by the tool, we segmented the raw videos in chunks of 3 minutes each, which might have introduced the risk of data loss. However, Affectiva uses frames of 1 second as unit of analysis, which mitigates this risk.

External validity Threats to external validity relate to the generalizability of the results. To enable fair comparison with related work, we chose the same task used in a previous study [2], [3]. Regarding participants, given the limited amount of participants included in our gold standard, we cannot claim a large generalization power. Nevertheless, we covered different levels of academic experience (by including Bachelors, Masters, and PhD students). Further replications should engage more participants, including also professional developers as they might experience a different range of emotions compared to students. Moreover, our previous study of biometric recognition of emotions [2] has been successfully confirmed in the industrial setting [4]. However, the application of sensor-based emotion recognition in industrial practice can be limited by the ability of the companies to purchase suitable wearable devices for their employers.

Conclusion validity The validity of our conclusions relies on the robustness of machine learning models. We mitigated this threat by running two different algorithms (Random Forest and SVM) addressing the same classification task, applying hyper-parameters tuning, and reporting results from different evaluation settings. However, we acknowledge the validity of our findings can be limited by the sample size.

VII. Conclusions

We investigated to what extent we can use non-invasive biometric sensors embedded in a wristband as a proxy for the identification of emotions of software developers while programming. Specifically, we experimented with machine learning using biometric features to predict the gold labels for emotions assigned by a tool that implements emotion recognition through facial expression analysis. We achieved a performance comparable to the one reported by a previous study relying on self-report as a gold standard. This study represents the first attempt to recognize the developers' emotions using facial expression recognition as a gold standard. Our findings represents a further step towards the implementation of tools that support emotion awareness and well-being of software developers at the workplace.

Ethical Impact Statement

Potential negative applications. We acknowledge the potential misuse of emotion detection classifiers when embedded in technology to monitor people's behavior. We do not advocate in favor of the implementation of a monitoring technology that might have an impact on privacy. Conversely, we advocate in favor of using sensor-based emotion detection to gain self-awareness of developers' own emotions. In the context of software development, the emotional feedback could be shared with the colleagues, e.g. during retrospective meetings in Agile development, on a voluntary basis.

Deceptive applications and failure modes. While demonstrating promising performance, we are aware that our classifier is not yet robust enough to be deployed for daily use without running the risk of incorrect classification. Nevertheless, the effect of misclassifying emotions is limited to loosing users' confidence that the classifier can serve their emotion awareness goals.

Risks to privacy. The protocol we use to collect data was carefully explained to the participants at the beginning of the experiment. According to the rules at the university that hosted the data collection procedure, participants were requested to sign a consent form where they give consent to the anonymous storage and treatment of the data collected during the experimental session. We are not releasing the data and we are not planning to do so in the next future.

Generalizability and biases. Due to the restricted size of the pool of participants, we cannot claim the generalizability of the classifier performance or the absence of biases due to differences in the individual biometrics.

References

[1] D. Ford and C. Parnin, "Exploring causes of frustration for software developers," in *CHASE*, 2015, pp. 115–116.

[2] D. Girardi, N. Novielli, D. Fucci, and F. Lanubile, "Recognizing developers' emotions while programming," in *ICSE*, 2020, p. 666–677.

[3] S. C. Müller and T. Fritz, "Stuck and frustrated or in flow and happy: Sensing developers' emotions and progress," in *ICSE*, 2015, pp. 688–699.

[4] D. Girardi, F. Lanubile, N. Novielli, and A. Serebrenik, "Emotions and perceived productivity of software developers at the workplace," *IEEE Transactions on Software Engineering*, pp. 1–1, 2021.

[5] D. Graziotin, F. Fagerholm, X. Wang, and P. Abrahamsson, "What happens when software developers are (un)happy," *J. of Systems and Software*, vol. 140, pp. 32–47, 2018.

[6] B. Lin, N. Cassee, A. Serebrenik, G. Bavota, N. Novielli, and M. Lanza, "Opinion mining for software development: A systematic literature review," *ACM Trans. Softw. Eng. Methodol.*, vol. 31, no. 3, pp. 38:1–38:41, 2022. [Online]. Available: https://doi.org/10.1145/3490388

[7] N. Cassee, F. Zampetti, N. Novielli, A. Serebrenik, and M. Di Penta, "Self-admitted technical debt and comments' polarity: An empirical study," *Empir. Softw. Eng.*, vol. 28, 2022.

[8] T. M. Amabile, S. G. Barsade, J. S. Mueller, and B. M. Staw, "Affect and creativity at work," *Admin Sci. Quart.*, vol. 50, no. 3, pp. 367–403, 2005.

[9] N. Novielli and A. Serebrenik, "Sentiment and emotion in software engineering," *IEEE Software*, vol. 36, no. 5, pp. 6–23, Sep. 2019.

[10] E. Marcos, R. Hens, T. Puebla, and J. M. Vara, "Applying emotional team coaching to software development," *IEEE Software*, pp. 1–8, 2020.

[11] Y. Andriyani, R. Hoda, and R. Amor, "Reflection in agile retrospectives," in *Agile Processes in Software Engineering and Extreme Programming*. Cham: Springer, 2017, pp. 3–19.

[12] R. Pekrun, *Emotions as Drivers of Learning and Cognitive Development*. New York, NY: Springer New York, 2011, pp. 23–39.

[13] A. Serebrenik, "Emotional labor of software engineers," in *Proc.of the 16th Ed. of the BElgian-NEtherlands software eVOLution symposium*, 2017, pp. 1–6.

[14] J. Kim and E. André, "Emotion recognition based on physiological changes in music listening," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 30, no. 12, pp. 2067–2083, 2008.

[15] S. Koelstra, C. Mühl, M. Soleymani, J. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, and I. Patras, "DEAP: A database for emotion analysis using physiological signals," *IEEE Trans. on Affective Comp.*, vol. 3, no. 1, pp. 18–31, 2012.

[16] M. Soleymani, S. Asghari-Esfeden, Y. Fu, and M. Pantic, "Analysis of EEG signals and facial expressions for continuous emotion detection," *IEEE Trans. Aff. Comp.*, vol. 7, no. 1, pp. 17–28, 2016.

[17] B. Reuderink, C. Mühl, and M. Poel, "Valence, arousal and dominance in the eeg during game play," *Int'l Journal of Autonomous and Adaptive Communications Systems*, vol. 6, no. 1, pp. 45–62, 2013.

[18] M. Li and B.-L. Lu, "Emotion classification based on gamma-band eeg," in *2009 Annual Int'l Conf. of the IEEE Engineering in Medicine and Biology Society*, Sep. 2009, pp. 1223–1226.

[19] P. J. Lang and M. Bradley, "The int'l affective picture system (iaps) in the study of emotion and attention," in *Handbook of Emotion Elicitation and Attention*, J. A. Coan and J. J. B. Allen, Eds. Oxford University Press, 2007, ch. 2, pp. 29–46.

[20] W. Burleson and R. W. Picard, "Affective agents: Sustaining motivation to learn through failure and state of "stuck"," in *Social and Emotional Intelligence in Learning Environments Workshop*, 8 2004.

[21] A. Kapoor, W. Burleson, and R. W. Picard, "Automatic prediction of frustration," *Int'l J. Human-Computer Studies*, vol. 65, no. 8, pp. 724–736, 2007.

[22] F. Canento, A. Fred, H. Silva, H. Gamboa, and A. Lourenço, "Multimodal biosignal sensor data handling for emotion recognition," in *SENSORS*. IEEE, 2011, pp. 647–650.

[23] J. Scheirer, R. Fernandez, J. Klein, and R. W. Picard, "Frustrating the user on purpose: a step toward building an affective computer," *Interacting with Computers*, vol. 14, pp. 93–118, 2002.

[24] P. A. Nogueira, R. A. Rodrigues, E. C. Oliveira, and L. E. Nacke, "A hybrid approach at emotional state detection: Merging theoretical models of emotion with data-driven statistical classifiers," in *IAT 2013*. IEEE, 2013, pp. 253–260.

[25] H. Vrzakova, A. Begel, L. Mehtätalo, and R. Bednarik, "Affect recognition in code review: An in-situ biometric study of reviewer's affect," *J. Syst. Softw.*, vol. 159, 2020.

[26] D. Girardi, F. Lanubile, and N. Novielli, "Emotion detection using noninvasive low cost sensors," in *ACII 2017*, 2017, pp. 125–130.

[27] P. Ekman, W. V. Friesen, and S. Ancoli, "Facial signs of emotional experience," *J. of Pers. & Social Psych.*, vol. 39, pp. 1125–1134, 1980.

[28] D. McDuff, R. El Kaliouby, and R. Picard, "Crowdsourcing facial responses to online videos," *IEEE Trans. Affect. Comput.*, vol. 3, no. 4, p. 456–468, jan 2012.

[29] J. Lien, T. Kanade, J. Cohn, and C.-C. Li, "Automated facial expression recognition based on facs action units," in *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, 1998, pp. 390–395.

[30] Z. Zeng, M. Pantic, and T. S. Huang, *Emotion Recognition Based on Multimodal Information*. London: Springer London, 2009, pp. 241–265.

[31] D. Dupré, E. G. Krumhuber, D. Küster, and G. J. McKeown, "A performance comparison of eight commercially available automatic classifiers for facial affect recognition," *PLOS ONE*, vol. 15, no. 4, pp. 1–17, 04 2020.

[32] J. O. Johanssen, J. P. Bernius, and B. Bruegge, "Toward usability problem identification based on user emotions derived from facial expressions," in *Proc. of the 4th Int'l. Workshop on Emotion Awareness in Software Engineering*, ser. SEmotion '19. IEEE Press, 2019, p. 1–7.

[33] T. Schmidt, M. Schlindwein, K. Lichtner, and C. Wolff, "Investigating the relationship between emotion recognition software and usability metrics," *i-com*, vol. 19, no. 2, pp. 139–151, 2020.

[34] 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)*, 2018, pp. 59–66.

[35] J. F. Filho, T. Valle, and W. Prata, "Automated usability tests for mobile devices through live emotions logging," in *Proc. of the 17th Int'l. Conf. on Human-Computer Interaction with Mobile Devices and Services Adjunct*, ser. MobileHCI '15. ACM, 2015, p. 636–643.

[36] D. McDuff, R. El Kaliouby, J. F. Cohn, and R. W. Picard, "Predicting ad liking and purchase intent: Large-scale analysis of facial responses to ads," *IEEE Trans. Affect. Comput.*, vol. 6, no. 3, p. 223–235, jul 2015.

[37] J. Russell, "A circumplex model of affect," *Journal of Personality and Social Psychology*, vol. 39, pp. 1161–1178, 1980.

[38] S. Baltes and P. Ralph, "Sampling in software engineering research: a critical review and guidelines," *Empir. Softw. Eng.*, vol. 27, no. 4, p. 94, 2022. [Online]. Available: https://doi.org/10.1007/s10664-021-10072-8

[39] J. Rottemberg, R. D. Ray, and J. J. Gross, "Emotion elicitation using films," in *Handbook of Emotion Elicitation and Assesment*, J. Coan and J. J. Allen, Eds. Oxford University Press, 2007, ch. 1, pp. 9–28.

[40] T. Fritz, A. Begel, S. C. Müller, S. Yigit-Elliott, and M. Züger, "Using psycho-physiological measures to assess task difficulty in software development," in *36th Int'l Conf. on Software Engineering, ICSE '14*, 2014, pp. 402–413.

[41] D. McDuff, A. Mahmoud, M. Mavadati, M. Amr, J. Turcot, and R. e. Kaliouby, "Affdex sdk: a cross-platform real-time multi-face expression recognition toolkit," in *Proceedings of the 2016 CHI conference extended abstracts on human factors in computing systems*, 2016, pp. 3723–3726.

[42] M. Hahsler, B. Gruen, and K. Hornik, "arules – A computational environment for mining association rules and frequent item sets," *J. of Statistical Software*, vol. 14, no. 15, pp. 1–25, October 2005.

[43] M. Hahsler, S. Chelluboina, K. Hornik, and C. Buchta, "The arules r-package ecosystem: Analyzing interesting patterns from large transaction datasets," *Journal of Machine Learning Research*, vol. 12, pp. 1977–1981, 2011.

[44] J. J. Braithwaite, D. G. Watson, R. Jones, and M. Rowe, "A Guide for Analysing Electrodermal Activity (EDA) & Skin Conductance Responses (SCRs) for Psychological Experiments," University of Birmingham, UK, University of Birmingham, UK, Tech. Rep., 2015.

[45] A. Greco, G. Valenza, A. Lanata, E. P. Scilingo, and L. Citi, "cvxEDA: A Convex Optimization Approach to Electrodermal Activity Processing," *IEEE Trans. on Biom. Eng.*, vol. 63, no. 4, pp. 797–804, 2016.

[46] D. Fucci, D. Girardi, N. Novielli, L. Quaranta, and F. Lanubile, "A replication study on code comprehension and expertise using lightweight biometric sensors," in *ICPC 2019*, 2019, pp. 311–322.

[47] N. Jaques, S. Taylor, A. Azaria, A. Ghandeharioun, A. Sano, and R. Picard, "Predicting students' happiness from physiology, phone, mobility, and behavioral data," in *ACII 2015*, vol. 2015, 09 2015, pp. 222–228.

[48] M. Kuhn, "The caret package," http://topepo.github.io/caret/index.html, 2009.

[49] N. Novielli, F. Calefato, D. Dongiovanni, D. Girardi, and F. Lanubile, *Can We Use SE-Specific Sentiment Analysis Tools in a Cross-Platform Setting?* New York, NY, USA: Association for Computing Machinery, 2020, p. 158–168.

[50] J. Cohen, "Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit." *Psychological Bulletin*, vol. 70, no. 4, p. 213, 1968.

[51] A. Viera and J. Garrett, "Understanding interobserver agreement: The kappa statistic." *Family medicine*, vol. 37, no. 5, pp. 360–363, 2005.

[52] J. M. Harley, F. Bouchet, M. S. Hussain, R. Azevedo, and R. Calvo, "A multi-componential analysis of emotions during complex learning with an intelligent multi-agent system," *Computers in Human Behavior*, vol. 48, pp. 615–625, 2015.

[53] R. Hochschild, *The managed heart: Commercialization of human feeling*. The University of California Press, 1983.

[54] D. Graziotin, X. Wang, and P. Abrahamsson, "Happy software developers solve problems better: psychological measurements in empirical software engineering," *PeerJ*, 2014.

[55] M. Storey, T. Zimmermann, C. Bird, J. Czerwonka, B. Murphy, and E. Kalliamvakou, "Towards a theory of software developer job satisfaction and perceived productivity," *IEEE Trans. Softw. Eng.*, pp. 1–1, 2019.

[56] A. Meyer, E. T. Barr, C. Bird, and T. Zimmermann, "Today was a good day: The daily life of software developers," *IEEE Trans. on Software Eng*, pp. 1–1, 2019.

[57] M.-A. A. El-Migid, D. Cai, T. Niven, J. Vo, K. Madampe, J. Grundy, and R. Hoda, "Emotimonitor: A trello power-up to capture and monitor emotions of agile teams," *JSS*, vol. 186, p. 111206, 2022.