

Article

Emotion Recognition in Usability Testing: A Framework for Improving Web Application UI Design

Darius Drungilas , Ignas Ramašauskas and Mindaugas Kurmis 

Marine Research Institute, Klaipeda University, H. Manto Str. 84, LT-92294 Klaipeda, Lithuania

* Correspondence: darius.drungilas@ku.lt

Abstract: Traditional usability testing methods often lack the ability to fully capture different aspects of the user experience (UX). The focus of this research work is to propose a framework and develop its comprehensive prototype to improve usability testing and UX analysis by integrating session recording, interaction logging, and emotion recognition methods. A trained emotion recognition model based on MobileNetV2 architecture in conjunction with Hotjar and Google Analytics is proposed to add more context to the user experience during usability testing. The results obtained during the testing of a developed framework prototype showed that UI testing based on UX principles and integration of emotion recognition can be a powerful tool for improving the UI of web applications. It is recommended to improve UI testing processes by incorporating these aspects and data analysis methods. This would provide a more complete and more objective picture of the usability of the interface.

Keywords: usability; user interface; user experience; affective computing; emotion recognition; machine learning



Citation: Drungilas, D.; Ramašauskas, I.; Kurmis, M. Emotion Recognition in Usability Testing: A Framework for Improving Web Application UI Design. *Appl. Sci.* **2024**, *14*, 4773. <https://doi.org/10.3390/app14114773>

Academic Editor: José Salvador Sánchez Garreta

Received: 30 April 2024

Revised: 25 May 2024

Accepted: 28 May 2024

Published: 31 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

User experience (UX), as defined by the ISO 9241-210 standard [1], encompasses a user's perceptions and responses that result from the use and/or anticipated use of a system, product, or service. This recently developed field of research still faces challenges in defining the scope of UX in general and the application of experiential qualities in particular [2]. The participation of end users is crucial to make interactions as easy and accessible as possible. Usability testing processes help define elements of this area and barriers to implementation [3]. However, for usability—as defined by ISO 9241-210 standard as an extent to which a system, product, or service can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context—use testing is mostly performed manually, where users provide feedback using usability evaluation methods such as cognitive steps with a think-aloud protocol and a heuristic evaluation survey [4].

Effectiveness refers to the accuracy and completeness of achieving goals, while performance considers the resources used to achieve these goals. Moreover, satisfaction reflects the convenience and acceptability of the work system for users and affected individuals [3,5]. These aspects can be effectively evaluated by using affective computing, which involves systems that detect, interpret, process, and simulate human actions. Emotion recognition and the study of behavior resulting from users' emotional states open up new and promising scenarios for a more immersive user experience [6]. The use of this technology in the usability testing process to automate the detection of critical UI areas is not given enough attention, and user research is an important part of the UI/UX design process because it helps designers understand the needs, goals, and preferences of target users. There are several methods designers use to gather user insights [7]: surveys for broad data collection, interviews for detailed individual insights, and usability testing to observe user

interactions with a product. Combining these methods provides a comprehensive assessment of usability and user needs. Researchers often explore measuring and understanding user experience [2] and improving UX practices using data science and process automation, particularly in agile project activities [3]. Marques et al. proposed a UX evaluation technique [8] that helps identify the causes that lead to a negative user experience. It is crucial to understand end users' needs and problems to enhance system usability.

The most common applications of emotion recognition technologies are in the development of computer games to enhance the overall experience [9]; in education, when trying to improve the teaching and accessibility of the program, by taking into account the emotional state of the students [10]; in medicine to detect health problems earlier [11]; in advertising companies trying to understand what kind of product the respective market wants; or in customer service by providing the best possible service quality. Medjden et al. investigated the automatic adaptation of the user interface depending on the multimodal emotion recognition system using the RGB-D sensor [12]. While their focus was on automation rather than UI testing, their work highlights the potential benefits of automating emotion-driven UI adjustments to enhance user experience.

This research addresses a gap in the field by investigating UI testing based on UX principles using emotion recognition technology. Automating UI usability testing is relevant for saving time and improving data collection. Emotion recognition data can enhance the identification of disliked interface features, informing interface improvements. The proposed framework prototype aims to facilitate usability testing by incorporating emotion recognition data to guide interface testing and identify areas needing improvement. This unique approach could benefit developers seeking user feedback to enhance web application UI.

2. Materials and Methods

2.1. Emotion Recognition Methods

In psychology, cognitive science, and neuroscience, there are two main ways to classify and perceive emotions: dimensional or categorical [13]. The dimensional approach usually uses dimensions such as negative vs. positive, calm vs. excited. One of the main dimensions of dimensional models is valence, which reflects the pleasantness or unpleasantness of an emotion [14]. Emotions can range from very positive to very negative, clearly indicating the overall emotional tone. Arousal refers to the intensity or level of activation of an emotion [15]. This dimension ranges from calmness and low arousal to excitement and high arousal. It provides insights into the energy and stimulation associated with different emotional states. Some dimensional models include dominance or control as a third dimension [16]. This dimension reflects the perceived control or influence of the emotion. This adds another layer of complexity to emotional experiences. A categorical approach tends to use separate classes. The first attempts to classify emotions into groups included a large number of emotions in order to represent the complex spectrum of human emotions as accurately as possible. The number of emotions then reached several dozen, but P. Ekman proposed a revolutionary idea [17]: the existence of universal, biologically determined basic emotions shared by various cultures. This laid the groundwork for the development of categorical models, suggesting that there is a set of discrete, cross-cultural emotions that all people experience. A categorical classification of emotions was chosen for this research.

Affective computing—the study and development of systems and devices capable of recognizing, interpreting, processing, and simulating human affect, relates to psychological and cognitive science, particularly the classification of emotions [18]. Affective computing uses machine learning techniques that process and classify various sensory input/output channels between computers and humans. The goal of most of these methods is to create labels that match the categories that a person would perceive in the same situation.

This paper explores the field of facial expression recognition to recognize the emotions of web application users from facial expressions without using any wearable, physiological sensors so that a person feels comfortable and the situation is as close as possible to

everyday use. Facial expression recognition can be used to evaluate the UX of digital products or services. It can also be useful to determine where a person is looking, or what particular object is of great interest to them. Such an approach allows feedback on how users react or enjoy using the system, making it a suitable choice to achieve the objective of this work. The detection and processing of facial expressions is achieved by various methods such as optical flow [19], hidden Markov models [20], neural network processing [21], or active appearance models [22]. Improvements are also being made to facial expression recognition methods [23] to maximize recognition accuracy. It is possible to combine more than one modality (multimodal recognition such as facial expressions and speech prosody [24]; facial expressions and hand gestures [25]; or speech and text [26] for multimodal data and metadata analysis) for a more reliable representation of a subject's emotional state for evaluation.

In this research, the convolutional neural network (CNN) is used for emotion recognition. The CNN is often found in emotion recognition models due to its excellent image processing characteristics (Figure 1).

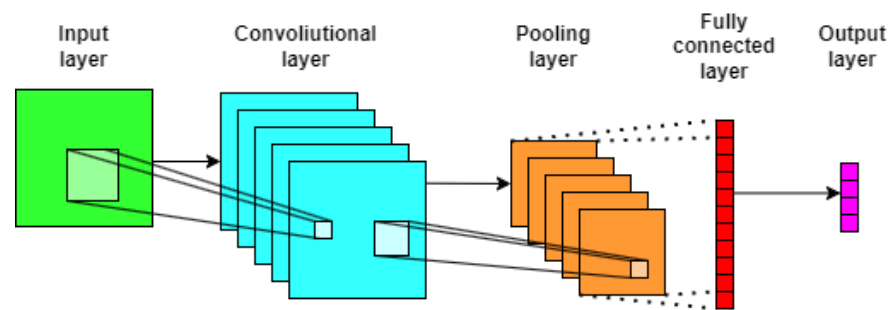


Figure 1. Architecture of CNN for emotion recognition.

The fundamental architecture of CNNs has a special structure that allows efficient processing of grid-structured data such as images [27]. Convolutional layers perform convolution operations in which kernels are swept over the input image, extracting various features such as edges, shapes, and textures [28]. Pooling layers such as maximum pooling reduce dimensions while retaining important information. Activation functions such as ReLU introduce nonlinearity. These operations are defined as follows:

1. The convolution operation in CNNs:

$$Z[i, j] = \sum_m \sum_n I[m, n] K[i - m, j - n], \quad (1)$$

where I is the input image, K is the convolutional filter, $[i, j]$ represents the spatial location in the output feature map, and $[m, n]$ represents the spatial location in the input image.

2. Maximum and average pooling operations:

$$Y[i, j] = \max_{m, n} X[i * s + m, j * s + n], \quad (2)$$

$$Y[i, j] = \frac{1}{mn} \sum_{m, n} X[i * s + m, j * s + n], \quad (3)$$

where X is the input feature map, i and j iterate over the spatial dimensions of the feature map, m and n iterate over the spatial dimensions of the pooling window, and s is the stride of the pooling operation, indicating the step size at which the pooling window moves over the input feature map.

3. ReLU activation function:

$$f(x) = \max(0, x), \quad (4)$$

ReLU activation function returns the input x if it is positive, and zero otherwise.

2.2. Web Application Testing Methodology

Web application testing often incorporates user behavior data collection to gain insights into user interaction. This data includes various aspects such as browsing habits, actions performed on web pages (clicks, text entry, button presses), and dwell time (time spent on specific elements). Session recording and interaction logging tools facilitate this data collection. By analyzing these captured user interactions, researchers can better understand how users navigate and utilize the application.

The implemented emotion recognition module captures an image of the user's facial expression and performs the emotion recognition process. This can be performed using a CNN-trained model that detects and analyzes facial features to predict the categorical state of emotion. The collected data are analyzed to gain insight into the user's behavior and experience with the web application to identify problems or challenges, to determine which elements of the application are most frequently used, or which actions most frequently trigger certain emotions in the user.

Based on the collected data and insights, actions can be taken to improve the usability of the web application and the user experience, including changes to the interface design, the addition of functionality, or changes to improve the user's interaction with the application. These steps form a methodology for testing online applications with an integrated emotion recognition prototype, which allows developers to gain useful insights into user behavior and experience in order to improve the application and ensure a good user experience.

2.3. Architecture of a Web Application Testing Framework Prototype with Integrated Emotion Recognition

In the context of this research, the architecture of the proposed framework prototype has been designed as depicted in Figure 2 to incorporate web application testing integrated with emotion recognition capabilities.

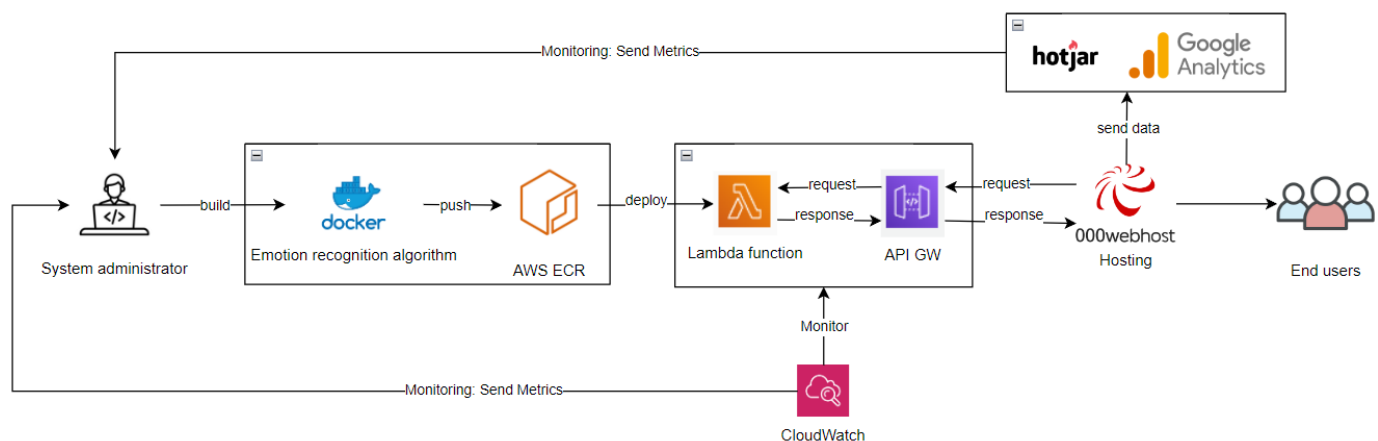


Figure 2. Architecture of the proposed framework prototype for web application testing.

A Docker image (Docker, Inc., Palo Alto, CA, USA—Docker Desktop v.4.25.2; Docker version 24.0.6) was built which is being used to ensure portability and easy deployment of the emotion recognition application. The image is stored in the AWS ECR repository (Amazon Web Services, Inc., Seattle, WA, USA), allowing for easy management and configuration of the application in the cloud. The application is integrated into a Lambda function, which acts as the main component of the system and is activated during API requests from the web page being tested. When the Lambda function receives a request, it uses an emotion recognition algorithm to identify the emotions based on the data provided (Figure 3).

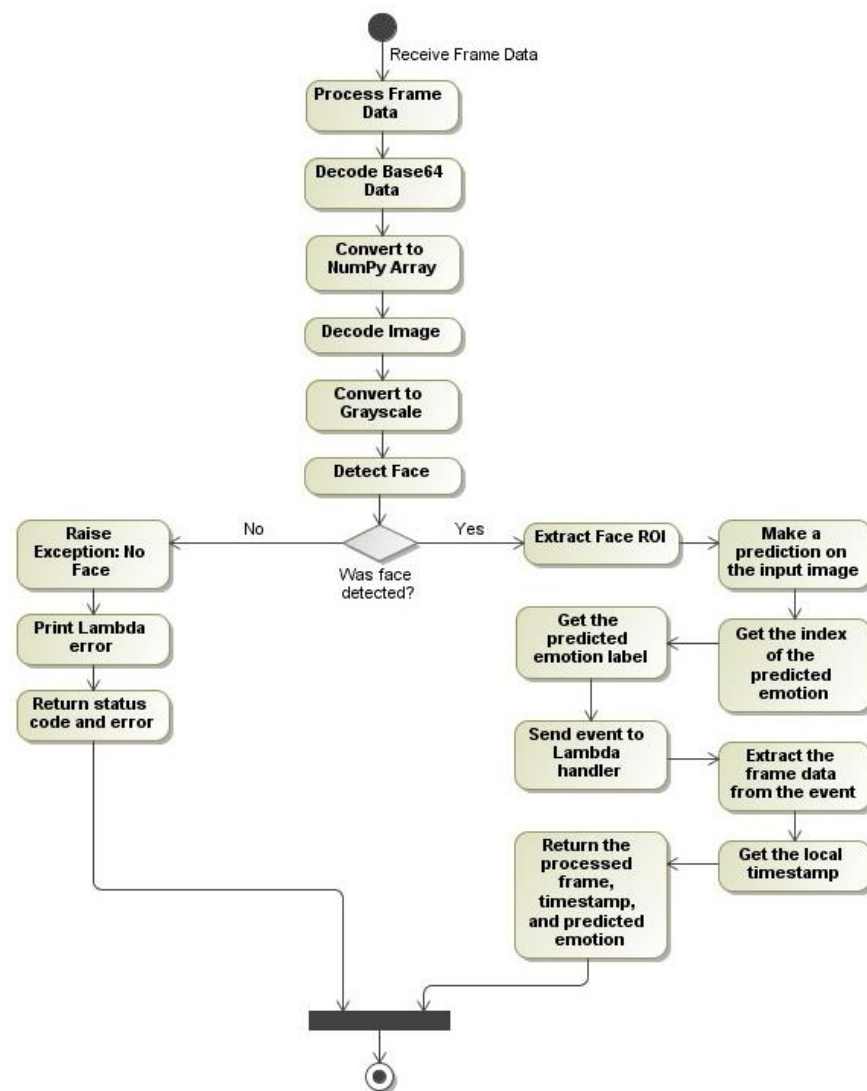


Figure 3. Emotion recognition algorithm activity diagram.

After recognizing the emotion, the function sends a response back to the web page via the API. Lambda function performance can be monitored and analyzed using the AWS CloudWatch platform, which provides detailed logs and metrics about function performance. This information helps to identify problems, optimize performance, and ensure prototype efficiency. Data collected during the user session is sent to the Hotjar (Hotjar Ltd., St Julian's, Malta) and Google Analytics (Google LLC, Mountain View, CA, USA) platforms, providing a more detailed picture of user behavior and interaction with the website. This data allows for a more detailed analysis of user behavior and identification of potential problem areas.

The integrated architecture provides a comprehensive test application process that effectively monitors user behavior, identifies problems, and improves web applications based on data and emotion recognition (Figure 4).

This ensures a reliable usability testing process and optimizes system efficiency and effectiveness using modern cloud computing technologies.

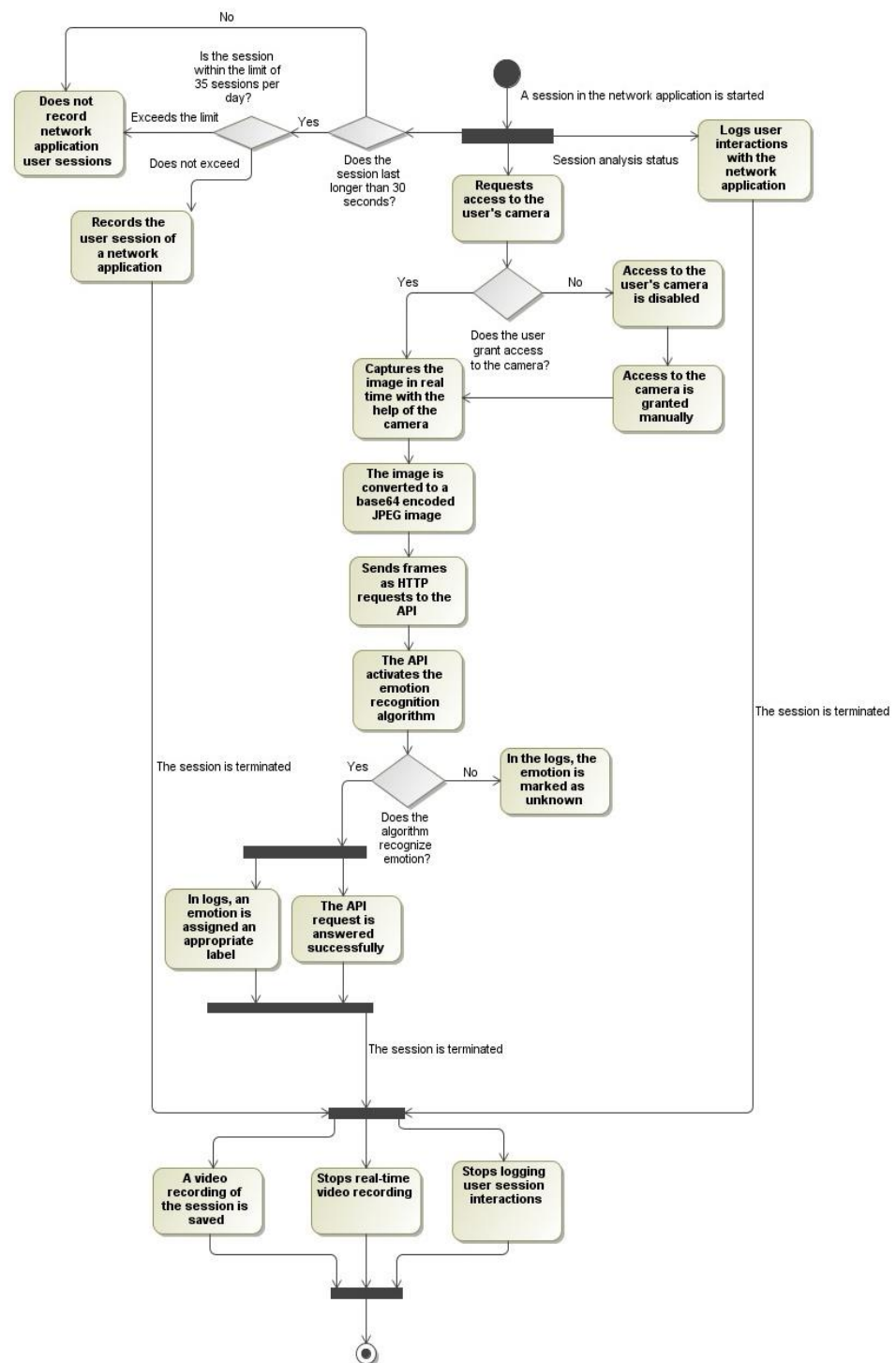


Figure 4. Activity diagram of the proposed framework prototype for web application testing.

3. Results

The emotion recognition model was trained using transfer learning with the MobileNetV2 architecture on the FER-2013 dataset, using a deep learning algorithm. The FER-2013 dataset consists of 28,709 grayscale face images of various ages, genders, and ethnicities, each with a resolution of 48×48 pixels. The dataset is labeled with seven emotion categories: angry, disgusted, scared, happy, sad, surprised, and neutral. The images were pre-processed to ensure that the face was centered and occupied as much space as possible. MobileNetV2 is a deep learning model designed specifically for mobile devices, and implements computer vision tasks in a simple and efficient way [29]. It is

based on the inverted residual structure, where the residual connections are between the bottleneck layers, and lightweight depthwise convolutions are used in the intermediate expansion layer, which is designed to filter functions as a source of nonlinearity [30]. This architecture was chosen for several reasons: it uses few resources, inverted residual blocks allow extracting important features from filters of different sizes, and the linear bottleneck layer contributes to flexibility and efficiency by reducing the number of parameters.

Several emotion recognition models were tested before MobileNetV2 was chosen as the main architecture. The emotion recognition part of the current version of the Deepsight Toolkit is still under development and currently uses a simple “smile” level system. Real-time emotion recognition from facial expressions performed poorly in tests using Deepface. Using the MATLAB (MathWorks, Inc., Natick, MA, USA—MATLAB R2020a) trained models, the classifier perfectly identifies emotions from the test dataset, but when using the real-time test, the classifier often incorrectly identifies the emotion as neutral (possible reasons: limited emotional expressions in the dataset or their bias, limited training data, and overfitting). Three popular pre-trained models, namely GoogleNet, AlexNet, and VGG19, also failed to correctly identify all emotions (Table 1).

Table 1. Results of the evaluation of emotion recognition models.

Neural Network	Training Dataset Size	Mini Batch Size	Max Epochs	Initial Learn Rate (s)	Total Training Time	Accuracy (%)
GoogleNet	20,098	10	6	1×10^{-4}	2928 min	59.6
GoogleNet	20,098	100	12	2×10^{-4}	-	46.1
AlexNet	20,098	50	6	3×10^{-4}	188 min	57.3
Vgg19	20,098	10	1	1×10^{-4}	1177 min	54.0
MobileNetV2	23,891	32	1	1×10^{-3}	34 min	45.8
MobileNetV2	23,891	32	6	1×10^{-3}	184 min	63.1
MobileNetV2	13,932	16	15	1×10^{-3}	374 min	79.5
MobileNetV2	13,932	16	25	1×10^{-3}	435 min	91.8

In further tests, two different algorithms were used to train the emotion recognition model. The first model was trained using the Keras (Keras team (originally developed by François Chollet)—Keras v. 2.15.0) library combined with OpenCV (OpenCV team (Open Source Computer Vision Library)—opencv-python-headless==4.9.0.80) image processing. This particular model was built using the FER-2013 dataset, which contains images with different facial expressions. The CNN model was trained using the Keras Sequential model (a linear stack of layers where one layer can be added at a time starting from the input) with different convolutional, pooling, and fully connected layers. The second model was trained (Figure 5) using the Keras and TensorFlow (Google LLC, Mountain View, CA, USA—tensorflow-cpu==2.15.0) libraries for emotion classification from images and the eINTERFACE_Image_Dataset dataset. The real-time test also failed to successfully detect emotions.

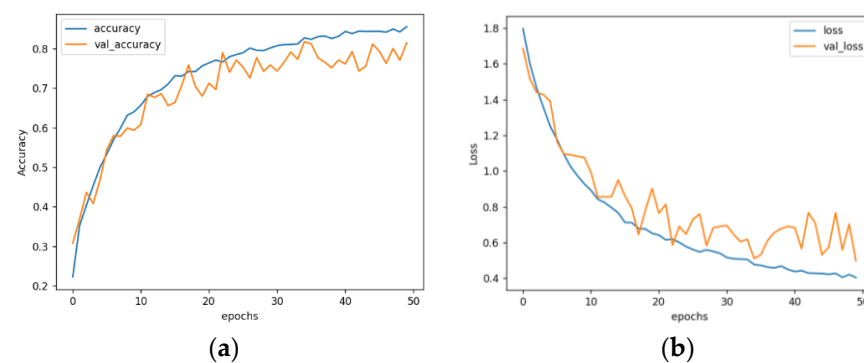


Figure 5. Plots of the results of the second trained model (Keras and TensorFlow): (a) accuracy plot; (b) loss plot.

The proposed MobileNetV2 model was modified by adding additional layers and integrating both input and output layers to create a new version. Learning parameters were set, including the categorical cross-entropy loss function, the Adam optimization algorithm, and the accuracy metric. Finally, the model was trained with the adjusted layer weights using the training data. Training was performed over a selected number of epochs with the available training set size. To prepare the data for training, the samples of the training data set are shuffled to obtain better training results. The model input and output data arrays are then separated into images with facial expression and emotional state labels, respectively. Deep learning models typically process multiple images as a single batch, expecting each image to have a specific format (width, height, and color channels); therefore, the `reshape(-1, img_size, img_size, 3)` method is used, providing the structure of the training data to the model, including the total number of images and the format of each individual image. Deep learning models often benefit from normalization of image pixel intensities. In this case, the image intensities are normalized from the range $[0, 255]$ to $[0, 1]$ by dividing each pixel value by 255. This improves training efficiency and standardizes the input.

The model was trained several times with different values of a set size, epochs, and number of training data depending on the performance of the equipment used to obtain the best result. In order to train the model for more epochs, it was necessary to reduce the number of images used for training and the size of the set, because the equipment used (Lenovo Legion Y540-15IRH 15.6" (Lenovo, Beijing, China)/FHD IPS/i5-9300H/RAM 16 GB/SSD 256GB/Nvidia GeForce GTX1660 Ti/6GB G6/Windows 10 Home) was not able to handle such a complex task. The test results are shown in Table 1.

The first verification test of all four trained models was performed by using AI-generated images containing faces with different emotions. The first three models had difficulty recognizing disgust, sadness, and fear because these emotions have certain facial features in common with other emotions: for example, a frightened person may have a gaping mouth as in the case of surprise, or a disgusted face may have furrowed eyebrows to signal an emotion of anger. The fourth model, with the highest accuracy, showed a significant difference from the previous models, successfully recognizing 7 out of 7 emotions (Figure 6).

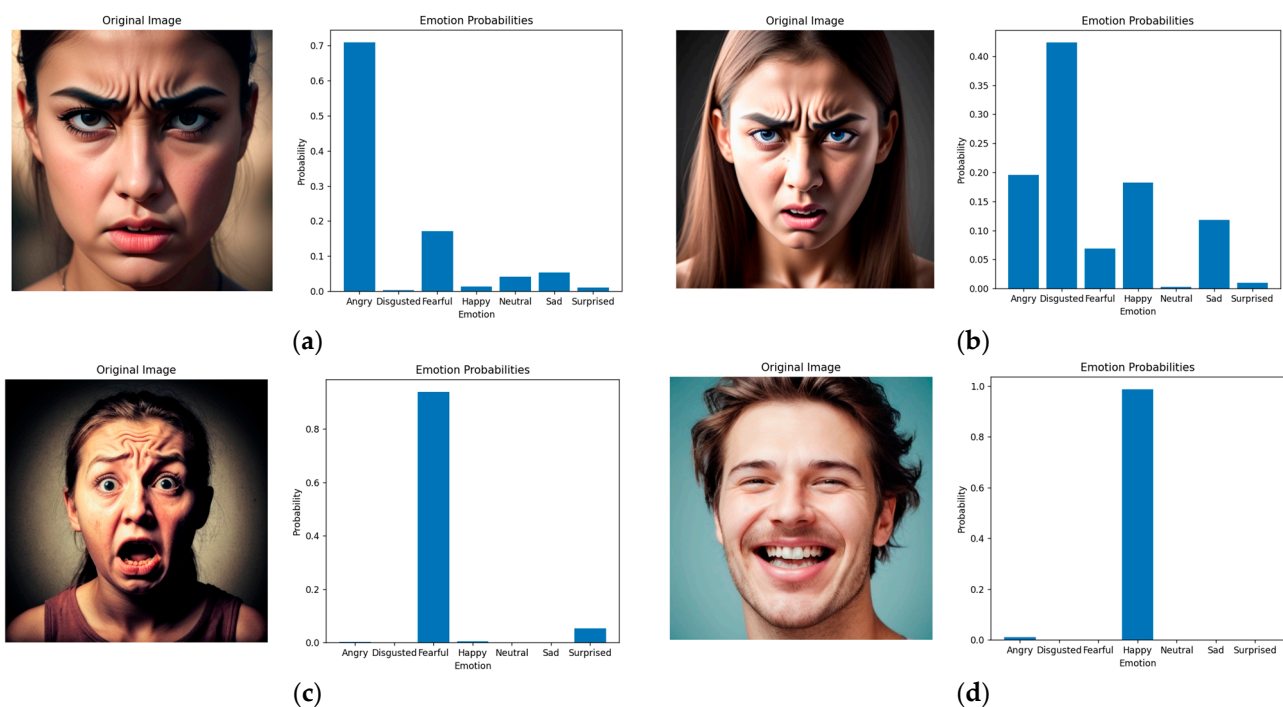


Figure 6. Cont.

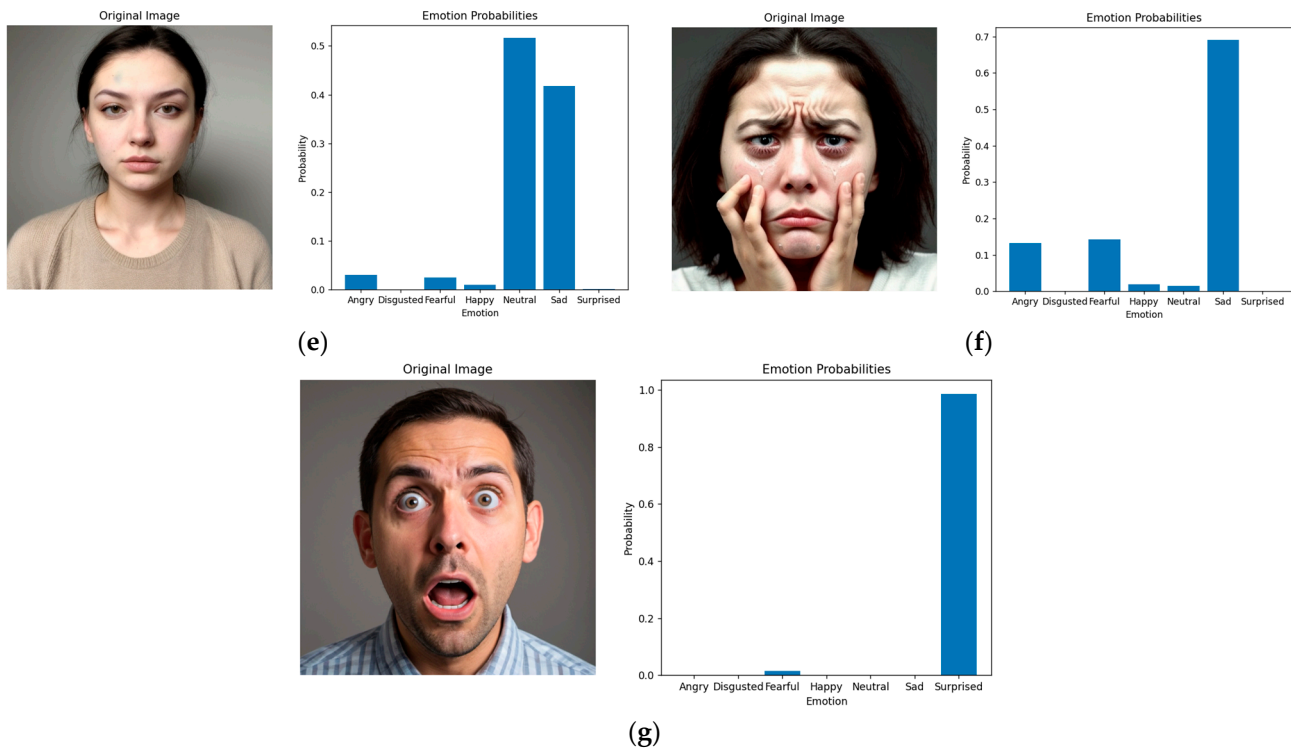


Figure 6. Test results of the fourth trial of the emotion recognition model, recognizing seven out of seven emotions (left side—the image that was used; right side—predicted emotion probabilities: (a) Angry; (b) Disgusted; (c) Fearful; (d) Happy; (e) Neutral; (f) Sad; (g) Surprised).

Since the fourth model showed high accuracy in consistently recognizing seven out of seven emotions using AI-generated images that were not a part of the dataset used to train the model in the first test, a real-time test was conducted to learn how the model would perform in real-world conditions.

The second test was conducted with the following setup: (1) the test subject (author—white male, 31 years old) was facing front of the laptop camera; (2) the laptop camera was 720p and fixed focus was used; (3) there were good lighting conditions on the face using natural light sources. The Haar classifier from OpenCV was used for face detection, and the face region was extracted for the detected face. This region was reduced to 224×224 pixels to prepare the input image for the model. The model then predicted the emotion based on the pre-processed image, and the result is displayed in real time. All seven emotions were recognized during the real-time test, as shown in Figure 7.

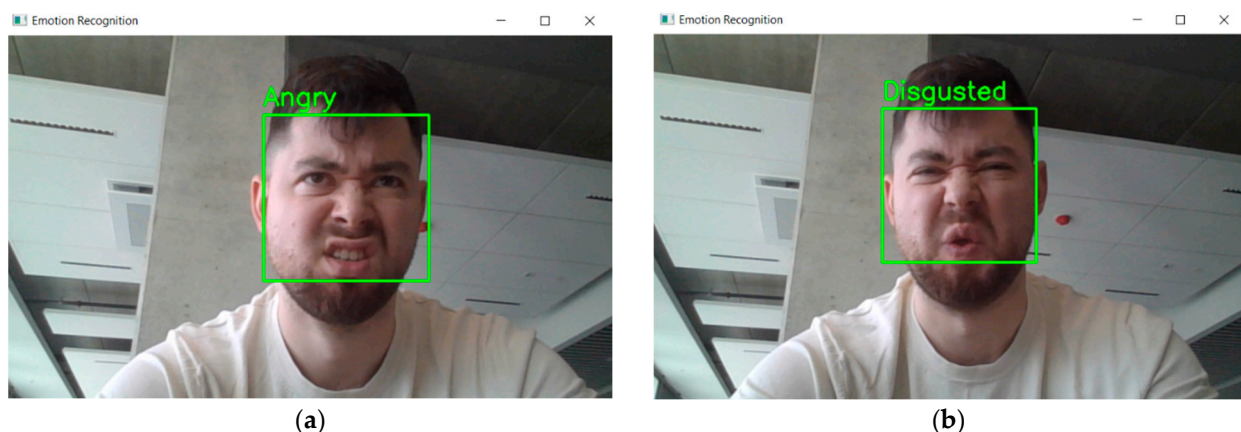


Figure 7. Cont.

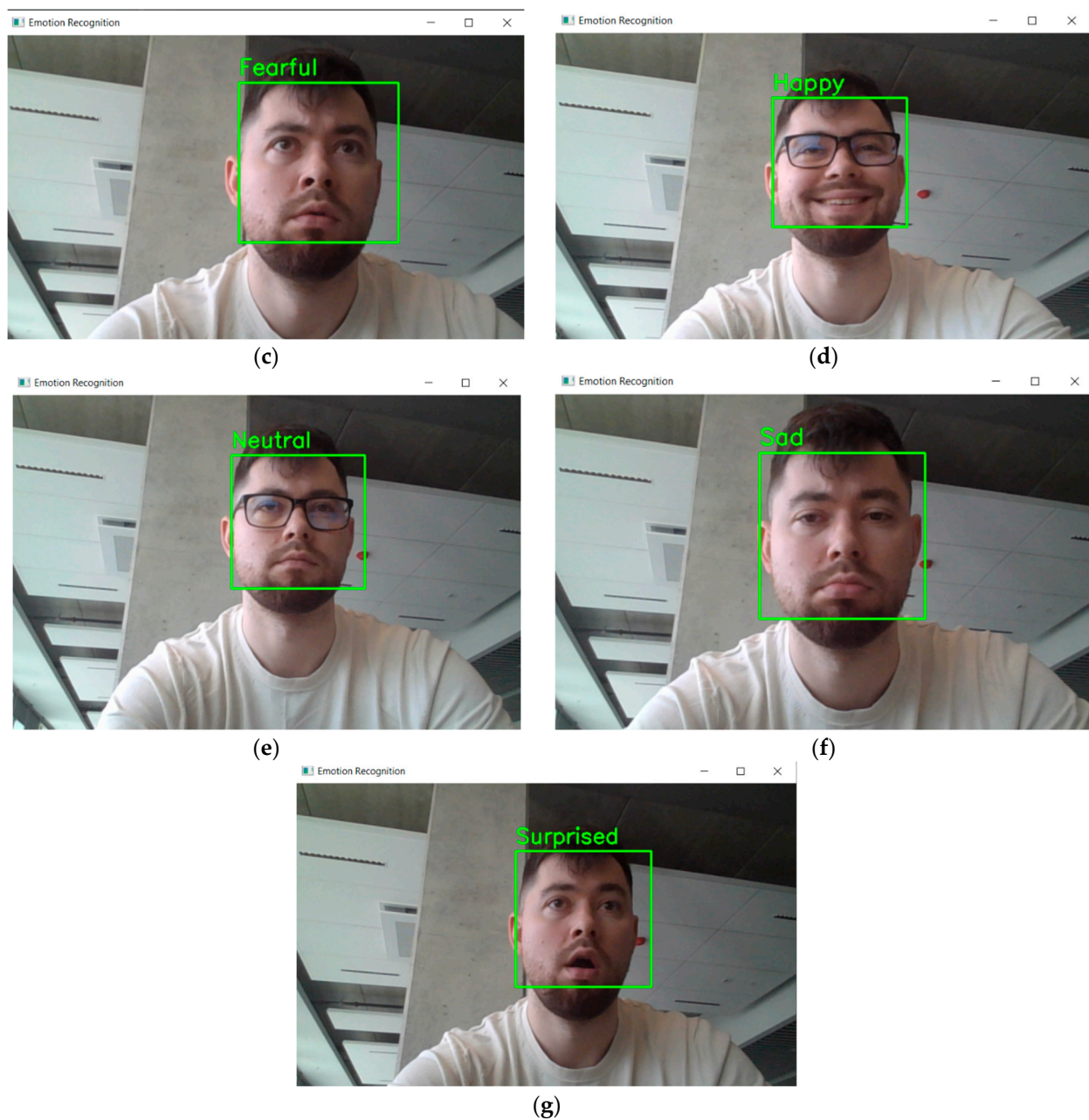


Figure 7. Real-time emotion recognition test results (under normal conditions without the use of special video/lighting equipment): (a) Angry; (b) Disgusted; (c) Fearful; (d) Happy; (e) Neutral; (f) Sad; (g) Surprised.

Since the real-time tests with the trained emotion recognition model showed promising results, the following software was selected to track the user's actions in the web application. For this purpose, the Hotjar tool was used, which allows monitoring the behavior of website visitors, analyzing browsing habits, and obtaining various information about how users use websites. Hotjar offers several features, but the most important one for this research is the recording feature. The recordings section allows us to monitor user sessions and view recordings that show how users move and interact with pages, and where they encounter problems or challenges. In addition, Google Analytics GA4 software was chosen to collect data on the usability of the web application interface. Analyzing user behavior is essential to understanding how visitors interact with the website, what they do, and how their experience can be improved. The integration was carried out by using the tracking script of both applications on the test website.

The emotion recognition program is activated by sending HTTP requests from the web application in use to the API service. To achieve this, the AWS platform was used by configuring the Lambda function using a Docker image that is located in the AWS ECR (Amazon Elastic Container Registry).

Base64 encoded data is decoded during frame capture. The result is a decoded byte object of the frame data. These bytes are then converted to a **numpy** array using an 8-bit unsigned integer type. The resulting array is interpreted as an image array, since each byte corresponds to a pixel value. Finally, the video is decoded using the OpenCV library. The result is an image array that is further used for face detection and emotion recognition. Since the Lambda function is used, the program code requires a Lambda handler to handle events. This method is invoked when the Lambda function is called. The general syntax for a Lambda handler takes two arguments: event and context.

Having a working Lambda function requires a trigger, which in this case is an API gateway. The Amazon API Gateway tool is used for API development. Since the RESTful API has a simpler client–server interaction model and the camera in the web application is configured to send HTTP POST requests to the server with fixed face data, this API type was chosen.

Two use cases were used to test the framework:

- Use case #1—logging recognized user emotions, recording and heat mapping user activity;
- Use case #2—detecting specific user actions (e.g., rage click), reviewing the Google Analytics page to determine the relationship between the recognized emotion and the specific user actions on the page.

A short user session was performed during the usability test. The test session was performed by five users using the web application that was specially developed by the author of this research. The purpose of the test scenarios is to show how parts of the system prototype—i.e., emotion recognition data, session logs, and interaction registration data—can be adapted and how their inclusion in UI usability testing can provide additional context to the UX. During the session, the users used the web application, browsing various pages, clicking buttons, and testing the functionality of UI. The web application design was improved to have more functionality and variety for usability testing. The home page included different types of blocks: headings, paragraphs, images, quotes, additional content sections, tables, and footnotes. Using the editing capabilities provided by WordPress, an options menu was created with two choices: Posts and About pages. Additionally, previously created blog posts were added to the posts page, and a new post was added to the archives and posts section on the right side of the dashboard. The main task for the users was to use all of the functionality offered by the simple blog-type web application, which had parts of poor UI design intentionally added to it, in order to collect emotion data for comparison with the session record. After the session, the CloudWatch logs are checked by selecting group **/aws/lambda/EmotionRecognitionProject**. By specifying the absolute date, it is possible to select the time period during which the user's test session took place. To filter out unnecessary data, such as the start and end of the event and additional warning messages, the search window is used by entering the keyword "emotion" for events where only an emotion was recognized, or "Lambda" for events where emotion recognition failed; for example, when no face was detected. In this way, the necessary emotion recognition data is obtained, which is tracked by timestamps to compare with the information provided by the Hotjar recording. The proposed scenario is more recommended for web applications that are already being visited by users for further improvement, rather than for systems that are still in the development phase.

When examining the data, the general predicted emotion should be neutral, but there may be unwanted noise. At this point, the noise filtering is performed manually by going through different emotions and looking for a cluster of negative emotions which can be used to compare with the Hotjar recording to see if the user encountered any UI problems (Figure 8).

▶ 2024-04-15T12:01:29.026+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:29', 'emotion': 'Neutral'}	2024/04/15/[\$LATEST]a36a3d316f9945328...
▶ 2024-04-15T12:01:30.025+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:30', 'emotion': 'Sad'}	2024/04/15/[\$LATEST]a36a3d316f9945328...
▶ 2024-04-15T12:01:31.128+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:31', 'emotion': 'Sad'}	2024/04/15/[\$LATEST]a36a3d316f9945328...
▶ 2024-04-15T12:01:32.048+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:32', 'emotion': 'Fearful'}	2024/04/15/[\$LATEST]a36a3d316f9945328...
▶ 2024-04-15T12:01:33.050+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:33', 'emotion': 'Fearful'}	2024/04/15/[\$LATEST]a36a3d316f9945328...
▶ 2024-04-15T12:01:34.028+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:34', 'emotion': 'Sad'}	2024/04/15/[\$LATEST]a36a3d316f9945328...
▶ 2024-04-15T12:01:35.048+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:35', 'emotion': 'Fearful'}	2024/04/15/[\$LATEST]a36a3d316f9945328...
▶ 2024-04-15T12:01:36.029+03:00	Lambda success: {'timestamp': '2024-04-15 12:01:36', 'emotion': 'Neutral'}	2024/04/15/[\$LATEST]a36a3d316f9945328...

Figure 8. Use case #1 for identifying problematic negative emotion events, highlighted in the red frame.

In use case #1, the recording section of the Hotjar platform is opened, and a record is selected based on the corresponding timestamp based on the CloudWatch log data. When the corresponding timestamp of the recording is checked, it shows that the user has double-clicked on an image from the image gallery (Figure 9).

The screenshot displays the Hotjar platform interface. On the left, the 'Recordings list' shows a table of recordings with columns for date, time, duration, and a play button. The central video player shows a recording of a user's interaction with a website. A red arrow labeled '1' points to a double-click on a small image in the 'Images testing' gallery. The 'Actions' panel on the right shows a list of actions, with a red box labeled '2' highlighting a 'Clicked: article-post-19-div-entr...' action.

Figure 9. Use case #1: (1) a double-click on a small image is detected; (2) the action list shows the double-click.

With this knowledge from the recording data, we can also compare it to a heatmap to see if the UI has a real problem that users tend to get irritated by. The heatmap of Hotjar shows the top three clicks on the page under investigation, and it is clear that double-clicking on the small image is common, as it is identified as a top click (Figure 10).

The heatmap analysis revealed that double-clicking on small images was a common interaction pattern on the page under investigation. This finding suggests a potential usability issue related to image size and user frustration. Use case #1 helped to identify a bad UI design decision where images are small and difficult to read without an option to change their size, which could cause negative emotions for the user.

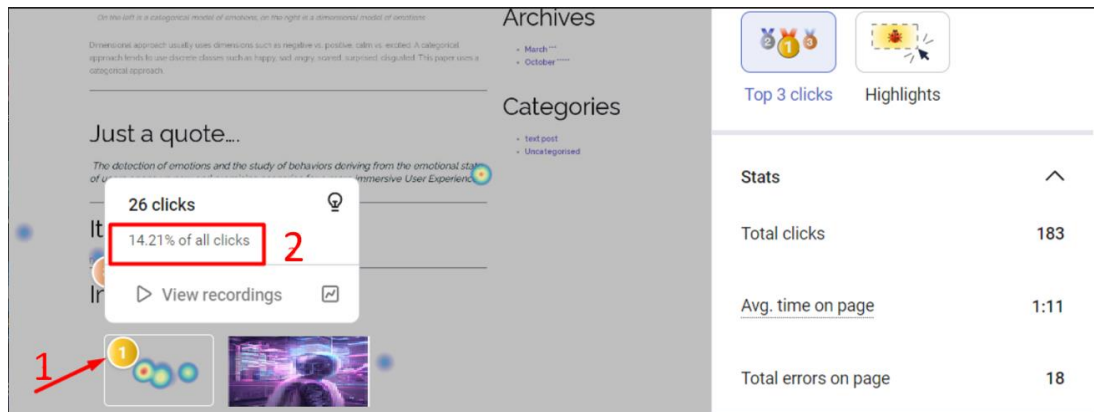


Figure 10. Use case #1: (1) top one click shown on the page by Hotjar heatmap; (2) percentage of this click compared it to all clicks on the page.

In use case #2, the examination of the event log revealed two groups of the Fearful emotion that stood out from the overall data. Following the same procedure, the timestamp of the recording on the Hotjar platform is selected accordingly, highlighting a clear problem in UI design. The Hotjar system itself marks this timestamp as a “rage click”. Rage clicks occur when users repeatedly click on a particular element or area of a web application over a short period of time, usually less than a minute. Multiple clicks on the right arrow button can be seen when viewing the replay and checking the action menu (Figure 11).

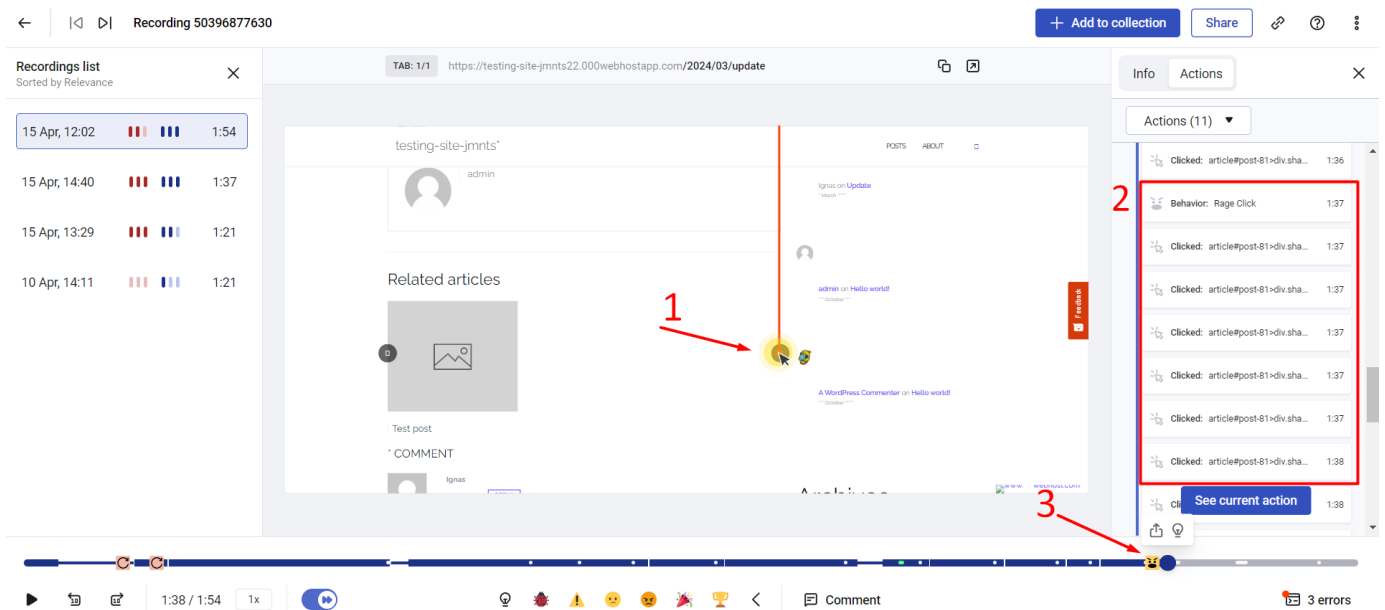


Figure 11. Use case #2: (1) “rage click” identified in recording; (2) multiple clicks in action list; (3) “rage click” marked by Hotjar.

Use case #2 helped to identify a confusing navigation of the page on one of the blog pages. Having this information, a website developer can rethink existing user interface design decisions, such as not using arrow buttons if there are not enough related articles. It is difficult to tell what usability problems may have been overlooked, but the emotion recognition data showed a real time user experience which helps to improve the usability of the web application. Use cases of other users are shown in Figures 12–15.

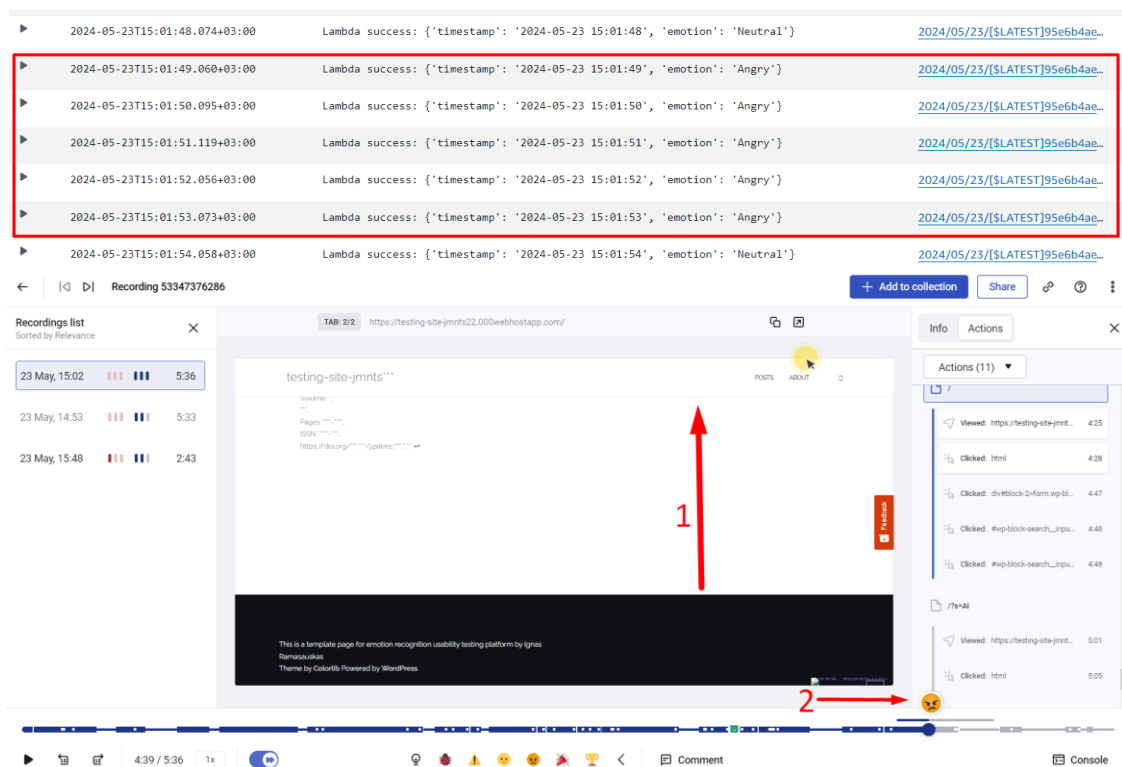


Figure 12. Group of five angry emotions detected (bordered by a red square). Potential problem—no back-to-top button, extensive scrolling (1) marked as frustrated (2) by Hotjar.

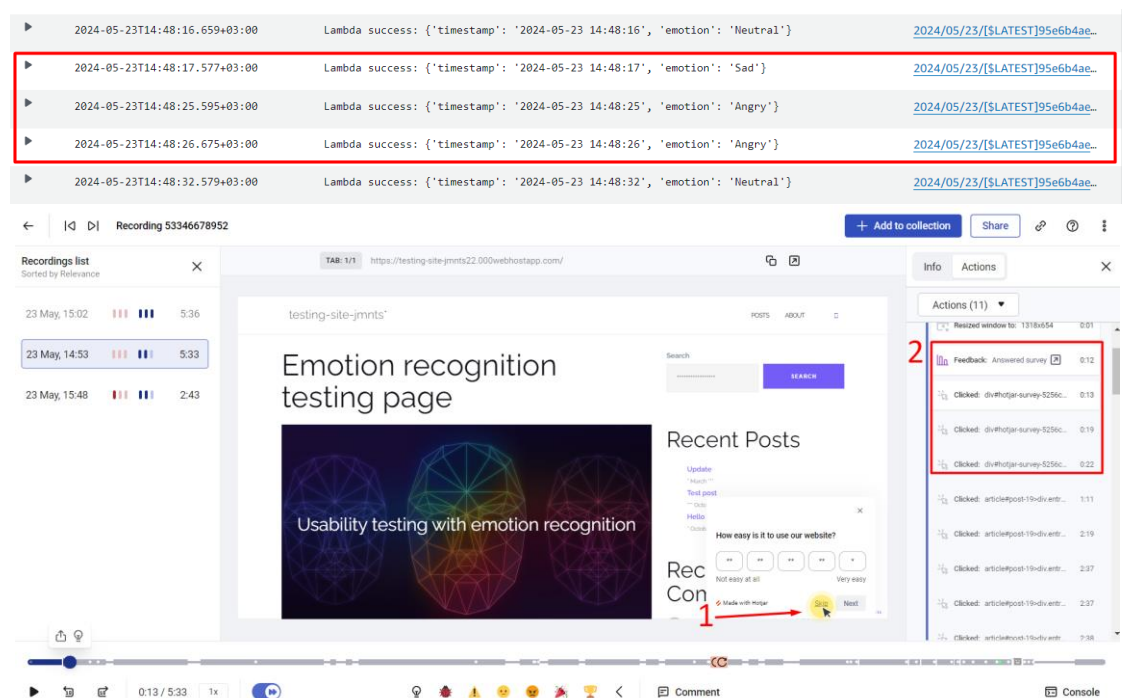


Figure 13. Group of three angry emotions detected (bordered by a red square). Potential problem—user did not like the feedback form and chose to skip (1) and skipping action logged in (2).

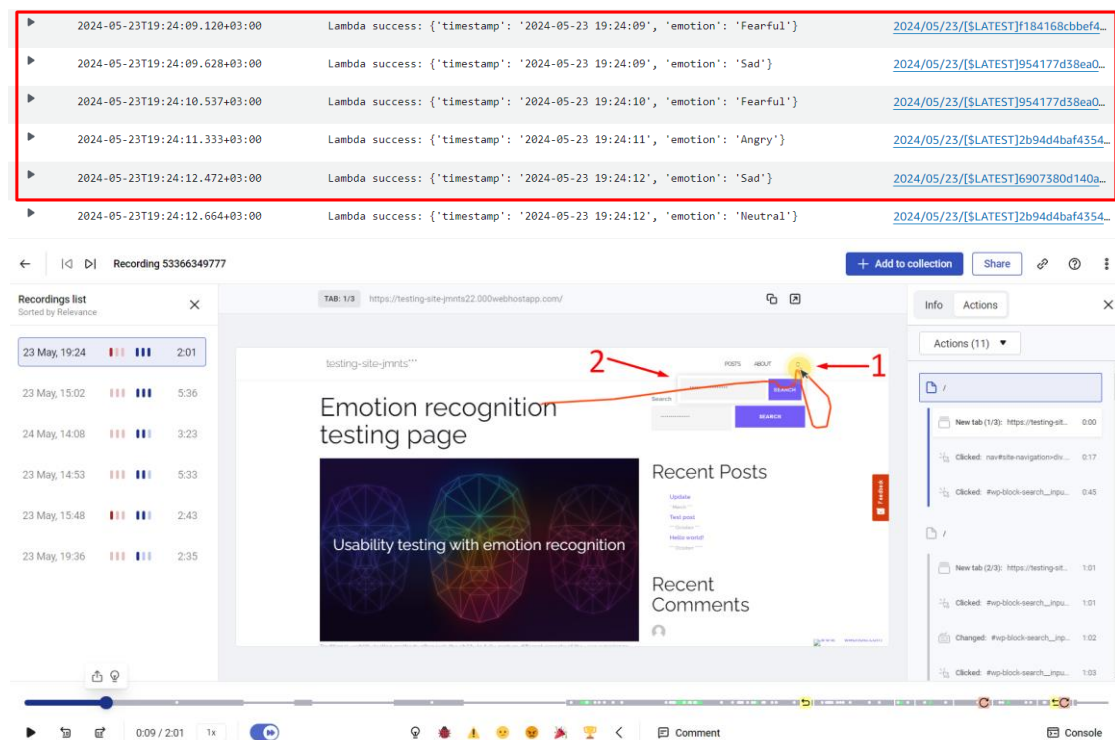


Figure 14. Group of five negative emotions detected (bordered by a red square). Potential problem—the search bar (2) keeps appearing and disappearing when the mouse is over it (1), making it difficult to navigate.

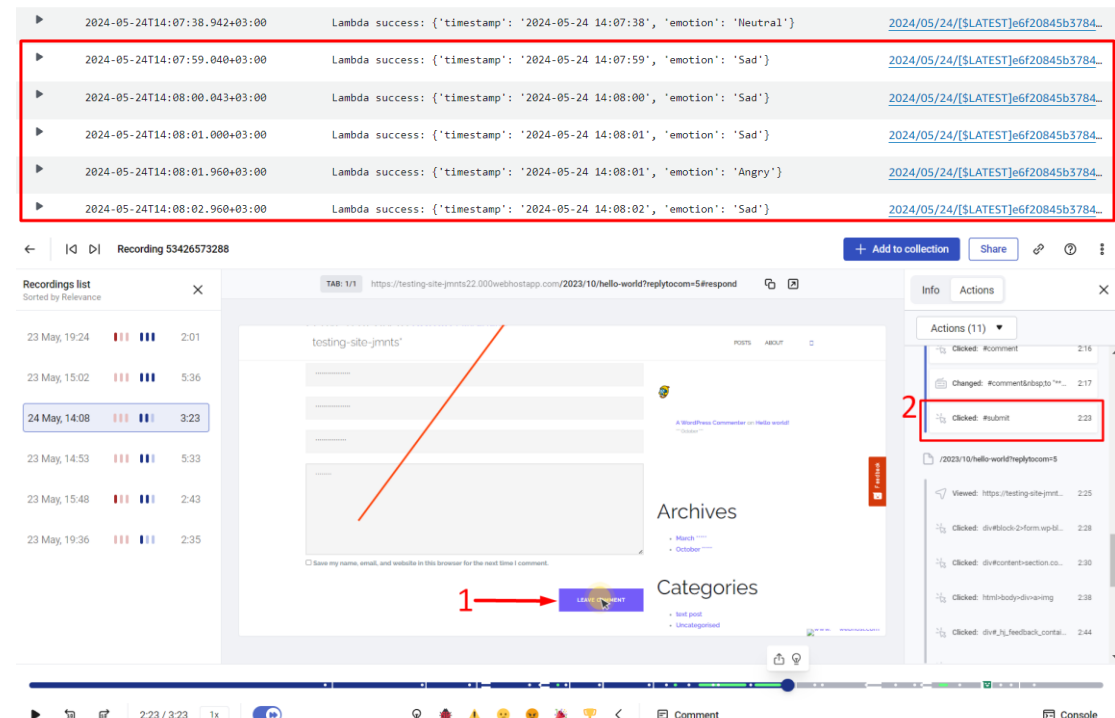


Figure 15. Group of five negative emotions detected (bordered by a red square). Potential problem—leaving a comment (1 and 2) results in a critical error in a Hello Word blog post.

Using the user interaction data collected by the Google Analytics platform, it is possible to decide whether it is worth making design changes if, for example, the page where the problem was found is not popular. In this case, the error was found on the title page and the

“Update” message blog page. In the Google Analytics Events tab, the value “page_view” is selected from the table, showing the statistics of all page views. The report shows that, as expected, the title page is viewed the most, but the “Update” message post on the blog is not visited very often (Figure 16).

User engagement > Page title ▼

TITLE		% TOTAL	AVG. TIME	
testing-site-jmnts22 ...	58.7%	-	18s	-
testing-site-jmnts22	14.7%	↓ 83.1%	14s	↓ 26.1%
About – testing-site-...	6.8%	-	7s	-
Test post – testing-s...	4.8%	↓ 62.8%	7s	↓ 68.5%
Update – testing-sit...	5.2%	-	8s	-
Posts – testing-site-j...	2.1%	-	5s	-
Hello world! – testin...	2.0%	-	6s	-
text post – testing-si...	1.3%	-	14s	-
Sample Page – testi...	1.2%	-	18s	-

1 – 9 of 17 < >

Figure 16. Use case #2: user engagement on different pages data from Google Analytics. The red frames highlight the pages with problematic UI in a use case #1 and #2.

Google Analytics revealed that the title page is highly visited, while the “Update” message blog post is not frequently accessed. This data helps in understanding the impact of UI issues based on the popularity of the affected pages. It indicates that making UI changes on less popular pages might have less of an overall impact compared to changes on highly visited pages. As the problematic area is only embedded in a page that is rarely viewed, the benefit of changing the user interface in this case can be weighed against the cost of the resources required to change and improve it. The testing and data validation shows that such a system can provide useful data for web application usability testing.

4. Discussion

The use of deep learning algorithms and data analysis makes the evaluation more objective, as it is based on mathematical models and actual data rather than the subjective opinion of the evaluators. This shift to data-driven assessments enhances the credibility and reliability of UX research results. Testing takes place in a real-time environment that reflects the actual user experience. This allows for a more accurate assessment of how users interact with the site and to identify real usability weaknesses. This real-time feedback can inform rapid design iterations and continuous improvement efforts. Incorporating emotion recognition into UX research emphasizes a user-centered design approach. By understanding users’ emotions, developers can create more empathetic and intuitive web application interfaces that are tailored to users’ needs and preferences.

One could think about automating the system on a larger scale. Automated notifications when large numbers of negative emotions are captured would allow for quick response and resolution of issues, saving human resources and time. Automated testing processes could easily scale to large data sets and large numbers of users, enabling broader and deeper testing. These benefits show that usability testing with such a framework could be more efficient and more accurate in reflecting user needs and behavior than traditional web application testing methods.

The emotion recognition technology could include more advanced algorithms or use additional sensors to provide more accurate data about users' emotions. Better emotion recognition results could also be achieved by ensuring that the lighting and camera quality are adequate to obtain clearer and better-quality images. It is also very important to balance the angle and composition of the shot to ensure that the angle of the cameras best matches the area of the face. To ensure that the emotion recognition system is universal and includes people of all nationalities and skin colors, the algorithm should be improved or more training data should be added to better recognize the emotions of people of different nationalities and skin colors.

To obtain more information from users—and in line with usability testing methods—using additional features of Hotjar such as surveys and interviews would allow us to obtain valuable feedback from users and better understand their needs and behavioral patterns. As the number of users grows, more time should be spent delving further into Google Analytics and taking advantage of this powerful tool, including a better understanding of on-page analytics, user flow tracking, and conversion analysis, which can provide valuable insights into user behavior and website usage. The integration with tools like Hotjar and Google Analytics enhances the framework's practicality by leveraging existing UX research methodologies. This interoperability streamlines data collection and analysis, making the framework accessible to UX researchers and developers.

Currently, the framework focuses on analyzing facial expressions from images captured by the camera. To extend its capabilities, speech analysis can be integrated by capturing audio at the same time as images. This would involve modifying the client side to capture both image frames and audio data, sending them to the Lambda function for processing. The Lambda function then parses and decodes the audio data and applies a separate audio emotion recognition algorithm trained on labeled data to identify emotional states. Techniques such as CNNs or RNNs that use audio features such as pitch and intensity can be used for this purpose. As datasets and input complexity increase, optimizing image and audio processing becomes critical for scalability. Implementing techniques such as batch processing and parallelization within the lambda function optimizes the efficiency of image and audio processing to handle larger amounts of data. Distributed processing techniques, such as using AWS Batch, can be explored.

The framework's architecture, built on cloud technologies like AWS Lambda and Docker, offers flexibility and adaptability to different types of web applications. Adapting the proposed framework for other applications, such as mobile apps, could involve developing SDKs for iOS and Android. These SDKs would handle tasks such as capturing images or video frames using device cameras and sending those frames to AWS Lambda for emotion recognition. Instead of Hotjar or Google Analytics, which are used for web applications, Firebase Analytics could track mobile-specific interactions such as app usage, touch gestures, and user engagement. It would also be recommended to create custom CloudWatch metrics to monitor specific aspects of Lambda function performance and to set up CloudWatch alarms to notify the system administrator when certain thresholds are crossed, allowing proactive management of resources.

5. Conclusions

This study investigated the development of a usability evaluation framework for web applications based on user emotion recognition in real-world scenarios. It is recommended to explore the development of more robust emotion recognition algorithms capable of accounting for diverse demographic factors, facial expressions, and cultural nuances. Additionally, the incorporation of multimodal interaction analysis, combining emotion recognition with gesture recognition, speech analysis, and physiological signals, could provide a holistic understanding of user engagement and satisfaction. Using the selected technologies, a prototype of the proposed framework was created, which helps to evaluate the usability of the web application based on emotion recognition, session recording, and interaction registration. Exploring applications in areas such as mobile interfaces, virtual

reality, and interactive systems could extend the usability evaluation capabilities of the framework to cross-domain assessments.

During a testing session, the proposed framework prototype demonstrated its ability to effectively capture and analyze user emotions using real-time data monitoring and analysis. Using CloudWatch log analysis and comparing the obtained data with Hotjar recordings, it was possible to identify problematic areas of a web application where users experience negative emotions. For future work, the integration with emerging technologies such as natural language processing (NLP) to expand the framework's capabilities in novel HCI contexts could be explored. Future work could explore the integration of AI-driven insights and automation within the framework to provide actionable recommendations for optimizing user experience based on emotion analysis. Additionally, developing customizable modules within the framework may accommodate specific user demographics, cultural contexts, and application domains.

Based on the successful testing, data validation, and complementary strengths of emotion recognition, session recording, and user interaction analysis, the proposed framework prototype holds great promise for evaluating web application usability based on user emotions in real-world scenarios.

Despite its strengths, this framework has certain limitations, such as reliance on specific hardware for emotion recognition. Future research could focus on enhancing the framework's adaptability to different platforms. Nonetheless, this approach empowers developers to make data-driven decisions for optimizing user experience.

Author Contributions: Conceptualization, D.D., I.R. and M.K.; data curation, D.D., I.R. and M.K.; formal analysis, D.D., I.R. and M.K.; methodology, D.D., I.R. and M.K.; resources, D.D., I.R. and M.K.; software, D.D., I.R. and M.K.; supervision, D.D., I.R. and M.K.; validation, D.D., I.R. and M.K.; visualization, D.D., I.R. and M.K.; writing—original draft, D.D., I.R. and M.K.; writing—review and editing, D.D., I.R. and M.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Lithuanian Research Council and the Ministry of Education, Science, and Sports of the Republic of Lithuania (Project No. S-A-UEI-23-9).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. ISO 9241-210; Ergonomics of Human-System Interaction, Part 210: Human-Centred Design for Interactive Systems. ISO: Geneva, Switzerland, 2019.
2. Law, L.-C.; Schaik, P.; Roto, V. Attitudes towards User Experience (UX) Measurement. *Int. J. Hum.-Comput. Stud.* **2014**, *72*, 526–541. [\[CrossRef\]](#)
3. Ferreira, B.; Marques, S.; Kalinowski, M.; Lopes, H.; Barbosa, S.D.J. Lessons Learned to Improve the UX Practices in Agile Projects Involving Data Science and Process Automation. *Inf. Softw. Technol.* **2023**, *155*, 107106. [\[CrossRef\]](#)
4. Alomari, H.W.; Ramasamy, V.; Kiper, J.D.; Potvin, G. A User Interface (UI) and User eXperience (UX) Evaluation Framework for Cyberlearning Environments in Computer Science and Software Engineering Education. *Heliyon* **2020**, *6*, e03917. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Galera, K.M.; Vilela-Malabanan, C. Evaluating on User Experience and User Interface (UX/UI) of EnerTrApp a Mobile Web Energy Monitoring System. *Procedia Comput. Sci.* **2019**, *161*, 1225–1232. [\[CrossRef\]](#)
6. Bisogni, C.; Cascone, L.; Castiglione, A.; Passero, I. Deep Learning for Emotion Driven User Experiences. *Pattern Recogn. Lett.* **2021**, *152*, 115–121. [\[CrossRef\]](#)
7. Walji, M.F.; Kalenderian, E.; Piotrowski, M.; Tran, D.; Kookal, K.K.; Tokede, O.; White, J.M.; Vaderhobli, R.; Ramoni, R.; Stark, P.C.; et al. Are Three Methods Better than One? A Comparative Assessment of Usability Evaluation Methods in an EHR. *Int. J. Med. Inform.* **2014**, *83*, 361–367. [\[CrossRef\]](#) [\[PubMed\]](#)

8. Marques, L.; Matsubara, P.G.; Nakamura, W.T.; Ferreira, B.M.; Wiese, I.S.; Gadelha, B.F.; Zaina, L.M.; Redmiles, D.; Conte, T.U. Understanding UX Better: A New Technique to Go beyond Emotion Assessment. *Sensors* **2021**, *21*, 7183. [\[CrossRef\]](#)
9. Setiono, D.; Saputra, D.; Putra, K.; Moniaga, J.; Chowanda, A. Enhancing Player Experience in Game With Affective Computing. *Procedia Comput. Sci.* **2021**, *179*, 781–788. [\[CrossRef\]](#)
10. Liu, J.; Tong, J.; Han, J.; Yang, F.; Chen, S. Affective Computing Applications in Distance Education. In Proceedings of the 2013 the International Conference on Education Technology and Information System (ICETIS 2013), San Francisco, CA, USA, 23–25 October 2013. [\[CrossRef\]](#)
11. Yao, K.; Huang, W.-T.; Chen, T.-Y.; Wu, C.-C.; Ho, W.-S. Establishing an Intelligent Emotion Analysis System for Long-Term Care Application Based on LabVIEW. *Sustainability* **2022**, *14*, 8932. [\[CrossRef\]](#)
12. Medjden, S.; Ahmed, N.; Lataifeh, M. Design and Analysis of an Automatic UI Adaptation Framework from Multimodal Emotion Recognition Using an RGB-D Sensor. *Procedia Comput. Sci.* **2020**, *170*, 82–89. [\[CrossRef\]](#)
13. Martinez, A.; Du, S. A Model of the Perception of Facial Expressions of Emotion by Humans: Research Overview and Perspectives. *J. Mach. Learn. Res.* **2012**, *13*, 1589–1608. [\[PubMed\]](#)
14. Kauschke, C.; Bahn, D.; Vesker, M.; Schwarzer, G. The Role of Emotional Valence for the Processing of Facial and Verbal Stimuli-Positivity or Negativity Bias? *Front. Psychol.* **2019**, *10*, 1654. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Höfling, T.T.A.; Gerdes, A.B.M.; Föhl, U.; Alpers, G.W. Read My Face: Automatic Facial Coding Versus Psychophysiological Indicators of Emotional Valence and Arousal. *Front. Psychol.* **2020**, *11*, 1388. [\[CrossRef\]](#) [\[PubMed\]](#)
16. Plutchik, R. The Nature of Emotions: Human Emotions Have Deep Evolutionary Roots, a Fact That May Explain Their Complexity and Provide Tools for Clinical Practice. *Am. Sci.* **2001**, *89*, 344–350. [\[CrossRef\]](#)
17. Ekman, P. An Argument for Basic Emotions. *Cogn. Emot.* **1992**, *6*, 169–200. [\[CrossRef\]](#)
18. Tao, J.; Tan, T. Affective Computing: A Review. In Proceedings of the International Conference on Affective Computing and Intelligent Interaction, Beijing, China, 22 October 2005; pp. 981–995.
19. He, S.; Zhao, H.; Juan, J.; Dong, Z.; Tao, Z. Optical Flow Fusion Synthesis Based on Adversarial Learning from Videos for Facial Action Unit Detection. In *Proceedings of the International Conference on Image, Vision and Intelligent Systems (ICIVIS 2021)*; Yao, J., Xiao, Y., You, P., Sun, G., Eds.; Springer Nature: Singapore, 2022; pp. 561–571.
20. Inthiam, J.; Mowshowitz, A.; Hayashi, E. Mood Perception Model for Social Robot Based on Facial and Bodily Expression Using a Hidden Markov Model. *J. Robot. Mechatron.* **2019**, *31*, 629–638. [\[CrossRef\]](#)
21. Muhammad Aamir, M.A.; Ali, T.; Shaf, A.; Irfan, M.; Saleem, M. ML-DCNNNet: Multi-Level Deep Convolutional Neural Network for Facial Expression Recognition and Intensity Estimation. *Arab. J. Sci. Eng.* **2020**, *45*, 10605–10620. [\[CrossRef\]](#)
22. Cheon, Y.; Kim, D. Natural Facial Expression Recognition Using Differential-AAM and Manifold Learning. *Pattern Recognit.* **2009**, *42*, 1340–1350. [\[CrossRef\]](#)
23. Karbauskaitė, R.; Sakalauskas, L.; Dzemyda, G. Kriging Predictor for Facial Emotion Recognition Using Numerical Proximities of Human Emotions. *Informatica* **2020**, *31*, 249–275. [\[CrossRef\]](#)
24. Ullah, M.; Li, X.; Hassan, M.A.; Ullah, F.; Muhammad, Y.; Granelli, F.; Vilcekova, L.; Sadad, T. An Intelligent Multi-Floor Navigational System Based on Speech, Facial Recognition and Voice Broadcasting Using Internet of Things. *Sensors* **2023**, *23*, 275. [\[CrossRef\]](#)
25. Verma, B.; Choudhary, A. Affective State Recognition from Hand Gestures and Facial Expressions Using Grassmann Manifolds. *Multimed. Tools Appl.* **2021**, *80*, 14019–14040. [\[CrossRef\]](#)
26. Sailunaz, K.; Dhaliwal, M.; Rokne, J.; Alhaji, R. Emotion Detection from Text and Speech: A Survey. *Soc. Netw. Anal. Min.* **2018**, *8*, 28. [\[CrossRef\]](#)
27. Namatevs, I. Deep Convolutional Neural Networks: Structure, Feature Extraction and Training. *Inf. Technol. Manag. Sci.* **2017**, *20*, 40–47. [\[CrossRef\]](#)
28. Yamashita, R.; Nishio, M.; Do, R.; Togashi, K. Convolutional Neural Networks: An Overview and Application in Radiology. *Insights Into Imaging* **2018**, *9*, 611–629. [\[CrossRef\]](#)
29. Dong, K.; Zhou, C.; Yihan, R.; Li, Y. MobileNetV2 Model for Image Classification. In Proceedings of the 2020 2nd International Conference on Information Technology and Computer Application (ITCA), Guangzhou, China, 1 December 2020; pp. 476–480.
30. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE: Salt Lake City, UT, USA; pp. 4510–4520.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.