Non-isomorphic Interaction Techniques for Controlling Avatar Facial Expressions in VR

Marc Baloup, Thomas Pietrzak, Martin Hachet, Géry Casiez

Non-isomorphic Interaction Techniques for Controlling Avatar Facial Expressions in VR

Marc Baloup
Inria, Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL
Lille, France
marc.baloup@inria.fr

Thomas Pietrzak
Univ. Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRIStAL
Lille, France
thomas.pietrzak@univ-lille.fr

Martin Hachet
Inria, Univ. Bordeaux, UMR 5800 LaBRI
Talence, France
martin.hachet@inria.fr

Géry Casiez*
Univ. Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRIStAL
Lille, France
gery.casiez@univ-lille.fr

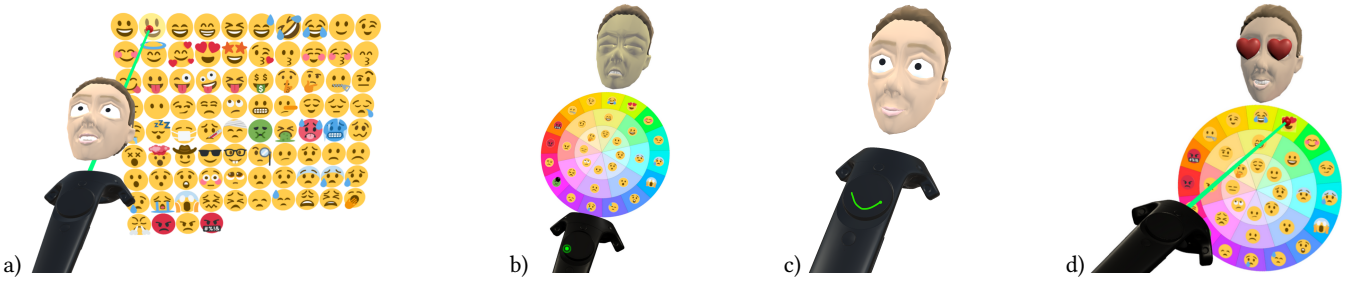

Figure 1: Four of the interaction techniques designed: a) RayMoji presents a grid menu in front of the user with raycasting used to select an expression, b) EmoTouch presents a circular menu above the controller and selection is made using the controller's touchpad, c) EmoGest is based on gestures on the controller's touchpad to define an expression, and d) EmoRay is a result of our first experiment, which presents a circular menu in front of the user with raycasting used to select an expression. All techniques present a feedforward or feedback of the expression, using a miniature version of the avatar's face.

ABSTRACT

The control of an avatar's facial expressions in virtual reality is mainly based on the automated recognition and transposition of the user's facial expressions. These isomorphic techniques are limited to what users can convey with their own face and have recognition issues. To overcome these limitations, non-isomorphic techniques rely on interaction techniques using input devices to control the avatar's facial expressions. Such techniques need to be designed to quickly and easily select and control an expression, and not disrupt a main task such as talking. We present the design of a set of new non-isomorphic interaction techniques for controlling an avatar facial expression in VR using a standard VR controller. These techniques have been evaluated through two controlled experiments to help designing an interaction technique combining the strengths of each approach. This technique was evaluated in a final ecological study showing it can be used in contexts such as social applications.

CCS CONCEPTS

• **Human-centered computing** → **Virtual reality**; **Interaction techniques**.

KEYWORDS

VR, Avatar, Facial expression, Emotion, Emoticons, Emoji

ACM Reference Format:
Marc Baloup, Thomas Pietrzak, Martin Hachet, and Géry Casiez. 2021. Non-isomorphic Interaction Techniques for Controlling Avatar Facial Expressions in VR. In *27th ACM Symposium on Virtual Reality Software and Technology (VRST '21), December 8–10, 2021, Osaka, Japan.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3489849.3489867

1 INTRODUCTION

Social apps are becoming mainstream in immersive environments. For example, Facebook [6], VRChat [28], and Rec Room [21] enable users to communicate with each other in virtual chat rooms. Communication in immersive environments is fundamentally different from mobile or desktop messaging, as it essentially relies on speech due to the difficulty to enter text. This also opens interesting opportunities: it is more engaging and favors social presence [7]. The use of facial expressions is an essential aspect of communication in virtual environments to enrich speech communication [9, 16], from professional contexts like VR meetings and conferences [13] to entertainment like VR role-playing games. In these situations, controlling the facial expression of an avatar should be a secondary task and not disturb a main task like talking, manipulating objects,

*Also with Institut Universitaire de France.

VRST '21, December 8–10, 2021, Osaka, Japan

or navigating in the environment. As such, users need to be able to control facial expressions in real time, to be consistent with the pace of a discussion, for example.

Facial expressions can first be controlled by isomorphic techniques, providing a direct mapping between users and their avatar expressions, using video [11, 25, 29] or voice analysis [6] to infer a facial expression to represent on the avatar. It helps focus on a primary task because users do not have to explicitly control their facial expression. However, users cannot choose the facial expression they would like their avatar to show. They may want to express something different from what they express with their actual face or voice. For instance, users may not want to have to laugh out loud to make their avatar animate accordingly. The second category of interaction techniques are based on non-isomorphic control, where users control parameters of the avatar's facial expression using interaction techniques based on input devices. This is an alternative solution that can overcome the limitations of isomorphic-based techniques. However current techniques are limited to a small set of facial expressions with a limited control of their parameters [21], opening opportunities to design new interaction techniques and control mechanisms.

We contribute a set of non-isomorphic interaction techniques for controlling the facial expressions of an embodied avatar, from the selection of an expression to the control of its intensity over time. The techniques have been evaluated through two controlled experiments to evaluate their performance and subjective preferences. This allowed us to propose a new technique that leverages the strength of each approach. Finally, we validated this technique in an ecological experiment.

2 RELATED WORK

In non-VR contexts, Emojis have become an essential aspect of non-verbal communication to enrich text messaging and social media posts on mobile and desktop applications. Users typically select emojis in grid layouts but alternatives exist in the literature [1, 20].

The current Unicode standard (v13) features 3292 emojis in nine categories, among which the *Smileys and emotions* category contains facial expressions and other face-related emojis [26]. Considering the wide use of emojis in mobile and desktop messaging, they can be leveraged to design interaction techniques for controlling facial expressions in VR, to favor a learning transfer from one context to others. Furthermore, people are increasingly accustomed to using emojis for purposes other than emotions, which is to be expected as well in the VR setting and needs to be considered when designing interaction techniques.

2.1 Isomorphic control of face expressions

The isomorphic control of face expressions is primarily based on the use of cameras. Several setups were studied: like a Kinect tracking the user's facial expressions [29], a combination of a PrimeSense camera and a Kinect [11], or embedded sensors in the headset [10, 25]. Another solution, used in Facebook Spaces [6], consists in analyzing the user's voice with a microphone and inferring the face expression. This method has privacy issues, because the users have no guarantee of what is recorded, and how it is interpreted. Also, this technique is limited to audible expressions (laughing loud for instance). Last, because of recognition algorithms performance, users may have to exaggerate the expressions.

Mobile applications use the same approach. Animoji [2] leverage the Face ID sensors to track the user's face in real time and transpose it to the avatar displayed on the screen. Other applications like Snapchat or Instagram provide filter-based face tracking to augment the expression of the user's face with decorations. These techniques provide a manual control of the avatar's appearance but aim at providing an isomorphic mapping between the users and their avatar's face pose, orientation and expression.

Above all, the issue with isomorphic control is that it assumes users would like to transpose their actual facial expression to their avatar in the virtual environment. However, users sometimes want to show a different expression, or with a different intensity. Taken together, these disadvantages mitigate the benefits of automatic detection in the general case. Non-isomorphic control of face expressions is an alternative to overcome these issues.

2.2 Non-isomorphic control of face expression

Facial expressions involve the movements from numerous muscles. To reduce the number of degrees of freedom, the Facial Action Coding System (FACS) [5] describes 24 Action Units (AU), representing atomic face movements and the MPEG-4 standard defines 68 Facial Animation Parameters (FAP) [17]. Even with this simplification of the degrees of freedom, manually controlling 24 or 68 integrated degrees of freedom in real time does not seem possible and further simplification is necessary.

Instead of trying to control each individual degree of freedom of a face, an approach is to let the user define and control the target face expression. Since face expressions are often associated with emotions, a solution consists in selecting an emotion instead of a face expression. Models of emotions such as Plutchik [18] or PAD [12] define sets of emotions with different levels of intensity. The Plutchik wheel of emotion [19] shows eight basic emotions (anger, fear, sadness, disgust, surprise, anticipation, trust, and joy) with three levels of intensities. For example, the intensity for joy ranges from serenity to ecstasy. The Plutchik model allows interpolating between some emotions [3] while the PAD model defines a 3D space to describe a large set of typical emotions.

Designers may also want to complement facial expressions related to emotions with abstract expressions, such as *money-mouth face* 🤑, *heart eyes* 😍, *exploding head* 🤯, *smiling with sunglasses* 😎, *face with medical mask* 😷, *cold face* 🥶 and *hot face* 🥵. However, this is not possible with an emotion model alone. To overcome this limitation, an alternative is to select a face expression among a predefined set of face expressions. These face expressions could either represent emotions or abstract expressions.

Rec Room, a VR social application, uses this approach to enable users to show a face expression on their avatar [21]. The user opens a marking menu with the menu button on the controller. The main menu contains 2 submenus proposing 3 facial expressions each. The first sub-menu contains positive expressions (*laughing* 🤣, *heart eyes* 😍 and *tongue out* 😛) and the second one the negative expressions (*crying* 😢, *angry* 😠 and *neutral face* 😐). When activated, the expression stays on the avatar's face between 2 and 3 seconds. Compared to emoji menus, the choice is limited, and the user cannot control other parameters such as the duration or the intensity of the expression.

Several interfaces have been proposed in the context of desktop interaction to allow users to select an emotion and then let the system translate this emotion to a facial expression. For example, *EmoCoW* [24], is a GUI showing a configurable *Emotional Color Wheel* to adjust the facial expression in real time. This interface allows movie animators to animate a face in real time using a low-cost equipment. However, the real-time control is limited to a small predefined set of emotions. Thus, the control of the face expression using all possible emotions is not possible in real-time. Bittorf and Wuethrich [3] used tangible interfaces like a MIDI keyboard, or a mixer with faders to select an emotion in a representation of Plutchik's classification of emotions [18]. Each basic emotion is mapped to one continuous input, for instance the slider's position. They may also combine the basic emotions by activating multiple inputs at the same time. While users have a fine control of the emotion they express, the keyboard-based techniques require them to learn the mapping. As a consequence, authors report a high cognitive load. This makes their techniques less suitable as a secondary task and inappropriate to VR due to inadequate input devices.

Techniques that let the user choose an emotion, using either an emotion model like Plutchik's as seen above, or the Pleasure-Arousal-Dominance model of Mehrabian [12], limit the choice to face expressions related to the expression of an emotion. Our techniques combine these approaches but also make it possible to select other expressions, representing facial expressions with artifacts, and not only combinations of base emotions.

3 PROPOSED INTERACTION TECHNIQUES

We present hereafter the interaction techniques we designed to select and control an avatar's facial expressions in VR. Our design is based on a task decomposition and consideration of a number of requirements.

3.1 Design rationale

We decompose the task of defining a face expression into sub-tasks that can be considered as independent. These sub-tasks follow a chronological order that consist in first selecting an expression, then controlling its intensity and finally controlling its duration or end. This approach has the advantage of using building blocks that can fit together to design new techniques. Each building block can also more easily be evaluated individually.

We posit that the interaction techniques should interfere as little as possible with a main task such as talking or interacting with the environment. This means that selecting and controlling a facial expression should require minimal cognitive effort and that selection should be fast to interfere as little as possible with the main task. In addition, feedforward and feedback are important aspects to consider given that users do not see their avatar's face. Feedforward is important during the selection step to help users choose an appropriate expression. Feedback is then important to help control the intensity of an expression or remind users about their avatar's current face expression.

We chose to use emojis to represent the expressions for two of our techniques based on menus. Our initial design of menus used representations of the corresponding avatar's face but the expressions were not always easy to differentiate. Emojis instead appear easier to distinguish, due to the well-visible face attributes, and they can be presented in smaller sizes than faces, allowing to fit more into view. In addition, users are familiar with emojis, which might facilitate skill transfer from social applications.

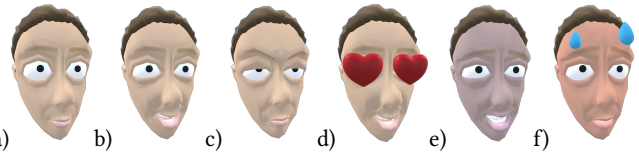

Figure 2: Examples of face expressions. a) *neutral* 😐; b) *grin* 😀; c) *unamused* 😒; d) *heart eyes* 😍; e) *cold* 🥶; f) *hot* 🥵

3.2 Implementation

We implemented a rendering and animation system within the Unity game engine version 2019.4. We chose rendering and animation techniques allowing us to implement our interaction techniques while providing convincing results and being easy to replicate.

We adapted a 3D model of an avatar face[1] to use it in Unity (Figure 2). We chose this model for its similarity with the non-realistic style used in VR games and social applications. Using more realistic models would require to use more advanced rendering and animation techniques [15, 22], which is beyond the scope of this paper. The Unity engine uses *blend shapes* (also called *shape keys* in Blender) to control the degrees of freedom for animating 3D models with C# code or the Unity GUI. We modified several shape keys of the Blender model to make them correspond to 21 of the Action Units of the FACS [5].

We augmented the avatar face with additional features. The color of the skin can change accordingly to some expressions, like *angry* 😡, *nauseated* 🤢, *hot* 🥵 or *cold* 🥶. We also created artefacts such as tears for *laugh* 🤣, *crying* 😢 or *sweat* 😥, and hearts for *heart eyes* 😍. Figure 2 shows a preview of our avatar with some of these expressions. We also animated expressions, typically *laugh* 🤣 or *cold* 🥶 for which the jaw is shaking accordingly. All these additions allow us to cover 85 emojis in the *Smiley* category [26]. Finally, we smooth the transition from one expression to another using exponential smoothing, with $\alpha = 0.2$ and a sample rate of 90Hz. We adjusted these values with informal pilot studies.

To provide feedforward and feedback, we represent a miniature version of the avatar's face, that we call *PuppetFace*, in the field of view of the user (for instance, above the controller, Figure 1). All techniques use it to provide feedback and some of them to also provide feedforward during the selection. The code for our techniques is available at ns.inria.fr/loki/AvatarFacialExpressions/.

3.3 Selection of expressions

This section presents four interaction techniques for the selection of expressions. The rationale is to explore a wide range of modalities, from menus to gestures and voice commands.

3.3.1 RayMoji. RayMoji shows a panel 1m in front of the user, and the PuppetFace appears over the virtual controller (Figure 1-a). The panel is fixed in the 3D space, and contains a grid of up to 10×9 emojis (∅ 7 cm). The size is a trade-off between the ease of emoji selection and the obstruction of the user's field of view. We set suitable values through informal pilot studies.

[1] Model by *cgcookie*, CC-BY (https://www.blendswap.com/blend/22625)

The layout is similar to emoji menus in current non-VR social apps. The presentation order is similar, and defined by the emoji standard [26]. Therefore, we assume users are accustomed to this layout, which can favor learning transfer. We used 84 emojis, which are all in the *Smileys and emotion* category of the emoji standard. The number of displayed emojis is adjustable, according to the application's needs. It allows to easily provide a wide range of expressions, from those representing emotions to less realistic ones.

To select a face expression, the user aims at the corresponding emoji on the menu with Raycasting. The PuppetFace provides feedforward information as the ray hovers emojis. When the user presses the trigger button, the menu disappears, and the avatar's face shows the selected expression.

3.3.2 EmoTouch. EmoTouch displays a circular menu above the virtual representation of the controller (Figure 1-b). It is designed to represent only expressions corresponding to emotions. The user controls a cursor on this menu with the circular touchpad of the controller to select one of the 24 facial expressions, using an absolute mapping. The PuppetFace appears above this menu to provide feedforward information. The facial expression is selected once the user presses the trigger button or the touchpad[2]. The menu disappears, but the PuppetFace remains visible. The avatar's face and PuppetFace display the selected expression until the user releases the pressed input.

The layout of the circular menu is similar to the Plutchik's wheel of emotions [18], with emojis to represent the expressions. In the original wheel, the strongest emotions are located at the center [19]. We reversed the intensity level so that the neutral emotion is at the center of the wheel, and the strongest emotions are on the periphery. The layout of this model conveniently places similar emotions next to each other, which, we hypothesis, facilitates visual search and helps memorize the location of each expression. However, the choice of emoji slightly differs from the Plutchik's model, because we did not find emojis representing *vigilance*, *submission*, *trust*, *admiration* and *optimism*. We replaced them with emojis that have a close meaning with the surrounding ones. We set the size of the menu and the number of emojis after informal pilot studies. It is a trade-off between a high number of items and selection accuracy. Each basic emotion uses a distinct hue (red, orange, yellow, green, cyan, blue, purple, pink), and saturation codes intensity.

3.3.3 EmoGest. EmoGest enables users to select a facial expression with unistroke gestures on the controller touchpad. The avatar directly shows the selected facial expression. The PuppetFace is always visible, and shows the avatar's current face in real time. The current expression remains active until the user disables EmoGest with the button above the touchpad[3] or selects another expression. To show all the available gestures, we provide a help menu accessible with the grip buttons[4] of the controller.

Unlike previous techniques, this technique does not display emojis. Therefore this technique has the advantage of being eyes-free, and this would help to focus on the main task.

We choose to implement the 8 expressions as listed in Figure 3.

[2]On VR controllers with a touchpad, the system can differentiate when the user touch or press the touchpad.
[3]The Menu button on the HTC Vive, and the B button on the Valve Index.
[4]The grip buttons detect when squeezing the controller with the fingers.

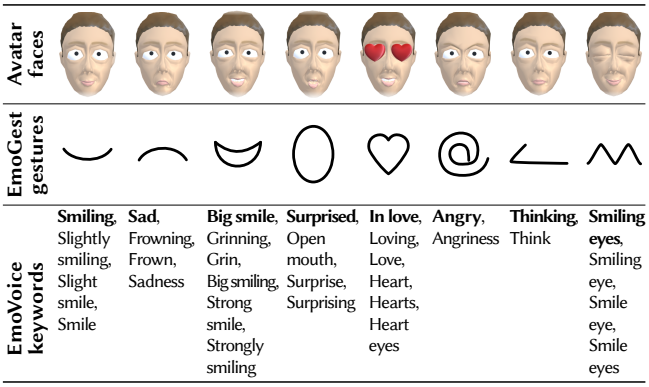

Figure 3: Implemented gestures for EmoGest and keywords for EmoVoice, associated with their corresponding facial expression. Keywords written in bold are the one given to the participants in the help menu.

To decide on the gestures to implement for each expression, first, we conducted an informal elicitation study with 6 participants. We asked them to draw a gesture for each expression. At least 2 participants agreed on a gesture for *slight smile* (6), *frown* (2), *big smile* (5), *open mouth* (4), *smiling eyes* (2), *heart eyes* (5), that are also inspired by ASCII emoticons `:-)`, `:-(`, `:-D`, `:-O`, `^^` and the shape of the heart. The *angry* face expression is represented by a `@` shape, like the ASCII emoticon `:-@` that is often used to express anger. Since there is no ASCII emoticon equivalent for the *thinking* face expression, we use the gesture as shown in Figure 3, that follows the shape of the hand in the emoji 🤔. The set of gestures is limited to a small number (8) to favor learnability, and reduce recognition errors. Also, we limited our technique to unistroke gestures to reduce selection time.

We implemented a custom version of the $1 recognizer algorithm [30]. We disabled the rotation to zero sub-algorithm to differentiate shapes that are identical but that have different orientations (like *smile* and *frown*, see Figure 3) but we kept the rotation correction at ±45° to allow slightly rotated inputs. We also enhanced the scale normalization step to keep the aspect ratio of the original stroke. It enables the recognition of shapes like horizontal and vertical strokes. Finally, our version of the recognizer also attempts to match the strokes with their backward version to allow users to draw shapes in one direction or the other.

After informal pilot studies of 3 participants, to reduce detection collision, we adjusted the shapes of *big smile* from a `D` (16% error rate) shape to a crescent moon shape (6%), and *open mouth* from a circle (33%) to a vertical ellipse (14%).

3.3.4 EmoVoice. EmoVoice uses speech recognition to control the facial expression of the avatar. When the user activates the technique with a press of the touchpad, it listens to the microphone until the touchpad is released. During this period, it detects specific keywords and activates the corresponding expression on the avatar's face.

Similarly to EmoGest, users can press the grip button[4] to access a help menu that shows the available expressions. This technique does not use emojis either, thus it has the advantage of being eyes-free.

For each implemented expression, we associated multiple keywords that fit the expression. It is also possible to support multiple languages, with different sets of keywords and voice recognizer configurations. For instance, for the 🙂 *slight smile* face, we recognize the words `slight smile`, `smile`, and `smiling`. We used the same principle to implement the other expressions of the technique : *frown* (4 keywords in English), *thinking* (2), *big smile* (6), *open mouth* (4), *heart eyes* (6), *smiling eyes* (4) and *angry* (2). Some of the keywords were suggested by participants during informal pilot studies.

We used the Speech-to-text tool from Google API, since it showed the best performance (compute time and recognition error) against CMU Sphinx. We used the python script SpeechRecognition to interface with these tools [31].

3.4 Control of expression's intensity, duration and ending

Following our task decomposition, we present five techniques to control the intensity, the duration and the ending of the face expression. We choose to control the intensity in real time, which indirectly allows for controlling the expression duration and end when the intensity reaches 0. We found the direct manipulation of the intensity more appropriate in social situations where expressions are more spontaneous. In addition, this allows to easily control the animation of the face.

Controller trigger button. The first technique maps the controller elastic trigger position to the intensity. The ending of the expression is triggered after a timeout of 1 second after the release of the trigger button. This timeout allows the user to adjust the intensity without releasing the control of the expression by accident. We also added 10 haptic detents to help users adjust the intensity. The intensity of the haptic pulse is proportional to the current intensity of the facial expression.

Controller orientation. The second technique maps the controller's tilt to the intensity. The expression ends when the user releases the selection button. This technique implements the same haptic feedback as detailed for the previous technique.

Controller shaking. The third technique maps the magnitude of the controller acceleration to the intensity. The higher the acceleration, the greater the intensity. We filter this value with a 0.5s moving average to avoid undesirable intensity oscillations. We set the parameters of this technique after pilot studies. The expression ends when the user releases the selection button.

Virtual elastic band. The fourth technique maps the length of a virtual white segment to the intensity. One end of the segment is attached to the visual representation of the selected expression in the menu. The other end is affixed to either the virtual controller, or to the extremity of a ray (*e.g.* in RayMoji), so that the user can manipulate the length of the segment. We reduce the segment thickness as it gets longer, similarly to an elastic band. This technique implements the same haptic feedback as previously presented.

Circular menu pointer position. The fifth technique requires a pie menu, such as EmoTouch with only one level (8 expressions). After the users have selected an expression, they can adjust its intensity by moving around the cursor in the menu. The intensity is null at the center and maximum on the edge. This technique implements the same haptic feedback as previously presented.

4 EXPERIMENT 1: SELECTION OF A FACIAL EXPRESSION

In this first experiment, we compare the non-isomorphic facial expression selection techniques presented in section 3.3. We study the following hypotheses. (H1) The selection time increases with the number of expressions available as the users may spend more time finding the correct expression. (H2) EmoGest and EmoVoice are more error-prone due to the limitations of recognition algorithms. (H3) EmoGest and EmoVoice require more training compared to other techniques because they require learning and performing gesture and voice commands. (H4) EmoTouch is faster than the other techniques thanks to the spatial mapping of the expressions that facilitates expression finding. (H5) Techniques having fewer errors and requiring less learning are preferred. We are also interested in finding relative differences between the techniques in terms of performance and subjective ratings.

4.1 Apparatus

The experiment used an HTC Vive VR headset on a PC [27]. Participants manipulated a Vive controller in their dominant hand. The experiment application was coded in C# with Unity 3D and the OpenVR plugin.

4.2 Task

Participants sat on a chair positioned in front of two virtual screens. The first one was directly in front of the participant to display the instructions. The second one was on the right to display additional information (Figure 4). The participants embodied the avatar face described in section 3.2.

Each trial consisted in replicating a given face expression using one of the techniques. The expression to replicate was first displayed on the left of the instructions screen using a representation of the avatar's face with the expected expression. Participants could then activate the technique by pressing the touchpad of the controller. After selecting an expression, a representation of the avatar's face with the resulting expression was displayed on the right of the instructions screen. A message "Correct" or "Wrong" appeared on the same screen during 1.5 s to indicate the correct selection of the expression. In addition, if the selection was wrong, the controller vibrated during 500 ms, and the participant had the opportunity to try the selection again. If the participant failed to select the right expression 4 times in a row, the experiment proceeded to the next trial.

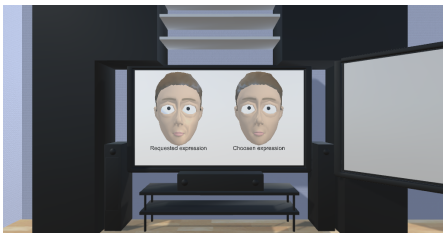

Figure 4: Point of view of the participants in the first experiment.

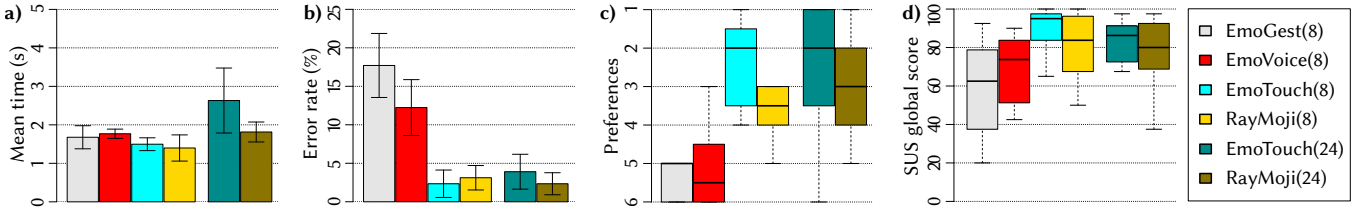

Figure 5: a) Mean times, b) error rates, c) participant preferences and d) SUS scores for the selection of an expression. Mean times and error rates results are with 95% confidence intervals.

4.3 Methodology

Twelve participants took part in this experiment (2 female, age mean = 26.2, σ = 5.3). Two of them never experienced Virtual Reality before. They had to select facial expressions using 6 Techniques: *EmoGest* (*EG8*), *EmoVoice* (*EV8*), *EmoTouch* with 8 expressions (*ET8*), *RayMoji* with 8 expressions (*RM8*), *EmoTouch* with 24 expressions (*ET24*) and *RayMoji* with 24 expressions (*RM24*). Both *EmoGest* and *EmoVoice* provided a set of 8 expressions for the experiment. *EmoTouch* and *RayMoji* were tested with both 8 and 24 expressions. The 8 expressions versions allow a fair comparison with *EmoGest* and *EmoVoice*. The 24 expressions versions enable measuring the benefits and cost of a larger set of face expressions. We used a Latin square to balance the order of the techniques between participants. Each block consisted of 8 trials. For each technique and block, the same following face expressions were requested to the participants, in the same order: *thinking* 🤔, *slightly smiling* 🙂, *frowning* 🙁, *angry* 😠, *grinning* 😀, *smiling with heart eyes* 😍, *open mouth* 😮 and *smiling eyes* 😊.

For each technique, participants first performed a training block in which participants selected the 8 expressions and a second training phase of 2 minutes, during which they were free to use the technique. During these two training phases, the right screen displayed instructions on how to use the current technique. Finally, the experimental phase had 4 blocks of 8 expressions in which we measured performance.

In summary the experiment followed a within-subjects design: 12 participants × 6 Techniques × 4 Blocks × 8 expression selections = 2,304 total trials. The experiment lasted around 1 hour and 15 minutes per participant.

4.4 Results

Our dependent variables are *selection time* and *error rate* for performance measures, as well as SUS [4] and preferences for qualitative measures.

Selection time. We define selection time as the interval between the activation of the technique and the validation of the expression selection. Trial with errors were discarded from the analysis of selection time.

A Box-Cox transformation of $\lambda = -0.75$ was applied to correct non-normal data. A repeated measures Anova shows a significant effect of Block ($F_{3,33} = 15.1$, $p < 0.0001$, $\eta^2_G = 0.13$), but no interaction with other factors, suggesting a similar learning rate for each technique. Pairwise comparisons with Bonferroni correction show significant differences between the block 1 and the blocks 3 and 4, and between the blocks 2 and 4. We assume this difference is due to a learning effect, so we removed the first two blocks from the remaining analysis. Further analysis shows a significant effect of Technique ($F_{5,55} = 6.8$, $p < 0.0001$, $\eta^2_G = 0.28$). Pairwise comparisons show that *RM8* (1.40s) is significantly faster to use than *RM24* (1.81s, $p = 0.003$) ; *ET8* (1.50s) is significantly faster than *ET24* (2.63s, $p = 0.01$). These results confirm H1 but H3 cannot be confirmed due to the lack of significant interaction between Block and Technique. We also did not find evidence for *EmoTouch* being faster than the other techniques so H4 cannot be confirmed.

Error rate. The error rate is the percentage of trials where the facial expression was not successfully selected at the first attempt. The error rate is computed per participant, technique and block and aggregated between conditions. The overall error rate is 6.9%. Data was transformed using an Aligned Rank Transform (ART) to correct non-normal distributions. A repeated-measures Anova shows a significant effect of Technique ($F_{2.1,23.0} = 19.0$, $p < 0.001$, $\eta^2_G = 0.58$). Post-hoc analysis show that *ET8* (2.3%), *RM24* (2.3%), *RM8* (3.1%) and *ET24* (3.9%) are significantly less error-prone than *EV8* (12.2%) and *EG8* (17.7%). These results confirm H2.

SUS Questionnaire. A Friedman analysis shows a significant effect of Technique on the global SUS score ($\chi^2(5) = 23.6$, $p = 0.0002$). A Wilcoxon post-hoc analysis shows that *ET8* ($M_{ET8\ S} = 95$) has a significantly higher SUS score than *EG8* ($M_{EG8\ S} = 62.5$, $p = 0.016$) and *EV8* ($M_{EV8\ S} = 73.75$, $p = 0.036$).

Participant preferences. After testing all the techniques, the participants had to rank them according to their preference. A Friedman analysis shows a significant effect of technique on user preferences ($\chi^2(5) = 17.0$, $p = 0.0045$). A Wilcoxon post-hoc analysis shows that participants significantly prefer *ET8* ($M = 2$), *ET24* ($M = 2$) and *RM24* ($M = 3$) over *EG8* ($M = 5$) and *EV8* ($M = 5.5$) ($p < 0.030$). These results together with the SUS analysis tend to validate H5.

4.5 Discussion

EmoVoice and EmoGest generally have lower performance compared to the other techniques. The error rate of both techniques is significantly higher than the others and EmoVoice is slower than RayMoji (8). In addition, participants reported they are more likely to use EmoTouch (8) than EmoVoice in their social VR experiences and they considered EmoGest is harder to use than all other techniques. They felt less confident using EmoGest than EmoTouch (8) and both EmoGest and EmoVoice are less preferred than EmoTouch (8 and 24) and RayMoji (24). According to the participant comments, the accuracy of the recognition system and the need to learn the words or gestures affected the overall performance of EmoVoice and EmoGest. Two participants had trouble when using EmoGest on the touchpad, due to the shape of the controller and

the morphology of their hands. Three of the participants suggested that using EmoVoice would interrupt their speech when talking to other people, due to the use of the microphone. Finally, two participants explained they do not want to use EmoVoice because they would feel embarrassed when speaking the keywords out loud. All these results show that these two techniques are less suitable to be used to control the facial expression of an avatar.

The results also indicate that showing only 8 expressions in the interface reduces the selection time for EmoTouch, compared to showing 24 since the user takes less time to find the desired expression. Despite these results, no other significant difference was found between the techniques with 24 expressions and their equivalents with 8 expressions. Informal feedback from participants shows that for EmoTouch, one of them would prefer to have more choices, and two others found having less choice easier to complete the task. For RayMoji, one participant would have preferred to have more expressions, but 3 others would have preferred having fewer items. According to these results, the number and the set of expressions displayed for a technique has to be set according to the needs of the target applications.

The statistical analysis did not show any significant difference between RayMoji and EmoTouch, but the informal feedback from the participants helped us identify the characteristics of both techniques that they liked or disliked. For EmoTouch, three participants found helpful the organization of the expressions on the circular menu and the use of colors, even if three participants found some colors were incoherent and needed some adjustments. One participant liked that EmoTouch takes up little space in their field of view, compared to RayMoji. The downside of EmoTouch, according to the participants, is that they need to look at the controller below their field of view, or to raise the controller to be able to see the circular menu. RayMoji does not have this issue, since the grid menu appears in front of the user, and the user points at the menu with the controller and a virtual ray. Also, one participant could not reach the upper side of the touchpad of the controller, due to their small hands, making EmoTouch difficult to use. The use of Raycasting in RayMoji avoids this issue.

All this feedback shows that participants prefer a colored menu, organized based on the Plutchik wheel of emotion, rather than a grid menu with no specific organization. However, the use of the touchpad to indicate an expression, and the position of the menu above the controller, are limiting factors for EmoTouch. Instead, the usage of Raycasting and the menu appearing in front of the user are better choices for the selection of an expression. Finally, the feedback from the participants helped us to design a new technique that we called EmoRay, which combines the strengths of RayMoji and EmoTouch. It uses the circular menu of EmoTouch with the Raycasting of RayMoji. The menu is placed closer to the user (from 1 m to 0.6 m) and is reduced in size to take less space in the field of view. The PuppetFace is moved from above the controller to above the menu, so the user does not have to look at the controller to preview the pointed expression and the placement of the menu is based on the direction of the ray when the menu is activated, instead of the direction of the HMD, to avoid the menu blocking the center of the field of view.

5 EXPERIMENT 2: CONTROL OF THE EXPRESSION'S INTENSITY

In this second experiment, we compare the techniques for the non-isomorphic control of the expression's intensity, as presented in section 3.4. We hypothesize that the participants would prefer to use the *trigger* button (H1), because it allows to validate the choice of the expression and control its intensity with the same input. We also hypothesis that the *Shaking* technique could be more exhausting to use, so participants would prefer other techniques (H2). The apparatus was the same as in the first experiment.

5.1 Task

Participants sat on a chair, as well as their avatar with a virtual mirror in front of them to see their avatar's face. The participant embodied the avatar face described in section 3.2, which moved along the user's HMD. The avatar face showed the face expression the user adjusted with the tested interaction technique. In this experiment we used EmoRay to select the facial expression.

The experimenter explained participants how the techniques work. Then, participants were instructed to practice the techniques at their convenience. Participants were encouraged to think aloud, in particular to compare techniques. Each technique was tested for 3 minutes.

5.2 Methodology

Ten participants took part in this experiment (4 females, age mean = 26.7, σ = 1.75). We used a within-subjects design, with the factor Technique. The techniques are the *Trigger* button, the controller's *Orientation*, the controller *Shaking*, the virtual *Elastic* and the *Pie* shaped menu. We used a Latin square to balance the order of the techniques between participants.

In summary the experiment followed a within-subjects design: 10 participants × 5 Techniques = 50 total test sessions. The experiment lasted around 45 minutes per participant.

5.3 Results

We asked the participant to rank the tested techniques according to their preference. A Friedman analysis shows a significant effect of technique on user preferences ($\chi^2(2) = 8.13$, $p = 0.017$). A Wilcoxon post-hoc analysis shows that participants significantly prefer *Elastic* (M = 1.5), *Trigger* (M = 2.5) and *Orientation* (M = 3) over *Shake* (M = 5) ($p < 0.049$). The analysis also shows that they prefer *Elastic* over *Pie* (M = 3.5, $p = 0.020$).

5.4 Discussion

Shaking the controller is the least preferred technique. Although some participants found this technique fun to use, seven of them

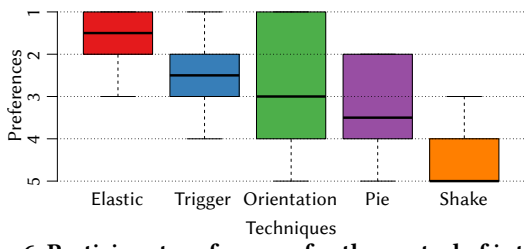

Figure 6: Participant preferences for the control of intensity experiment. Higher on the graph (lower value) is better.

reported that this technique lacks precision and four participants found this technique exhausting. The results, along with the participants' feedback, confirm H2. Three participants also think that it is only suitable for a few expressions, like *very angry* 😡 or *laughing out loud* 🤣, and it is disturbing to use for other expressions.

Despite *Trigger* being preferred over *Shake*, it does not have the highest median rank. Therefore, we cannot confirm H1. Five participants found that *Trigger* allows precise control of the intensity, but 4 other participants said the precision of this technique is not enough. Two participants liked the implementation of the timeout, when releasing the trigger by mistake, since it avoids the expression to end instantly. In contrast, one participant stressed the fact that this timeout prevents releasing the control instantly.

The haptic feedback was implemented in all techniques except *Shake*. Some participants noticed and found helpful the presence of a haptic feedback while using the techniques (4 participants for *Orientation*, 3 for *Trigger*, 2 for *Elastic* and 2 for *Pie*).

Overall, participants preferred to control the intensity with *Elastic*, with a median rank of 1.5. Five participants liked the visual feedback of the virtual elastic band. Also, four participants stated that the technique allows a precise control of the intensity.

As a result, we decide to keep the *Virtual elastic* technique to control the expression's intensity, along with EmoRay for the selection of the expression. The combination of these two techniques is called EmoRayE in the following.

6 EXPERIMENT 3: ECOLOGICAL STUDY

This experiment was designed to evaluate EmoRayE in an ecological context, similar to social VR applications. The participants had to control their facial expression while they discussed with the experimenter. We studied the following hypothesis. (H1) Our technique has a good usability. (H2) It is pleasant to use. (H3) The technique does not disturb users when they are talking. (H4) It does not interrupt users when they are talking. (H5) The technique does not disturb users when they are listening to somebody.

6.1 Apparatus

The participant and the experimenter were in different rooms during the experiment. The participant wore a Valve Index VR headset and hold two controllers, one in each hand. The experimenter wore an HTC Vive VR Headset and hold also two controllers [27]. Each headset used a dedicated computer. The experiment application was coded in C# with Unity 3D, the OpenVR plugin, and the network API Mirror [14]. The application instance on the participant's computer served as the host and client. The experimenter's computer was connected to the server with an Ethernet cable, to minimize latency and maximize the connection stability. Although this configuration does not reflect a standard internet connection, it minimizes the risk of disconnection and it ensures consistent experimental conditions across the participants. Audio was transmitted in real time over the network.

6.2 Task

Participants took a seat in a chair. Their avatar was also sitting in front of a virtual mirror, which allowed participants to see themselves. The participants embodied the avatar face described in section 3.2, as well as a full body as shown in Figure 7.

Figure 7: View of the ecological experiment.

The avatar face and hands were moving according to the movement of the user, using inverse kinematics, with the hips and legs fixed in the scene. The face expression of their avatar changed according to the manipulation of the tested technique. The participant's and experimenter's avatars were facing each other 3m apart. Figure 7 shows the virtual room.

The conversation was semi-directed, so the two persons could talk about any subject, like in any social situation. However, the experimenter prepared specific discussion topics to suggest that participants use facial expressions. For example, the experimenter asked participants about dishes they like or not (😋, 😋, 🤢, ...), their thoughts about recent sports events (😀, 😢, 😠, ...), and thoughts about their work (🤔). During the discussion, both the participant and the experimenter had to control their facial expression, to support their own words or to react to what the other was saying.

6.3 Methodology

Nine participants took part in this experiment (4 female, age mean = 25.8, σ = 2.8). Three of them never experienced Virtual Reality before. The technique tested was EmoRayE, the combination of EmoRay to select the expression, and the *Virtual elastic* to control the expression's intensity.

During the experiment, the participants went through 3 phases. First, the experimenter explained to the participant how to use EmoRayE, while the participant was able to use the technique to explore its possibilities. In the second phase, the participant and the experimenter went through the semi-directed discussion, as described in the previous section. Finally, the participant completed a paper questionnaire. The experiment lasted around 25 minutes per participant.

6.4 Results

Our dependent variables are the number of times the participants used EmoRayE, the duration of the conversation, the number of different expressions they used, the SUS [4] and AttrakDiff [8] questionnaires, and additional questions with Likert scales.

The participant and the experimenter talked on average for 10min 25s (σ = 1min 56s) and the participants used EmoRayE on average 26.1 times (σ = 14.3).

SUS Questionnaire. The overall mean SUS Score is 80.6 (σ = 14.2). It corresponds to the *A* scale according to Sauro's Percentile rankings of SUS scores ($\geq$ 90th percentile) [23], showing a very good usability and confirming H1.

AttrakDiff Questionnaire. EmoRay was rated above average (Pragmatic quality: mean = 0.71, CI ±0.60 ; Hedonic quality: mean = 0.86, CI = 0.43). As a result EmoRayE is positioned in the region between "neutral" and "desired" on the AttrakDiff scale (Figure 8a). While

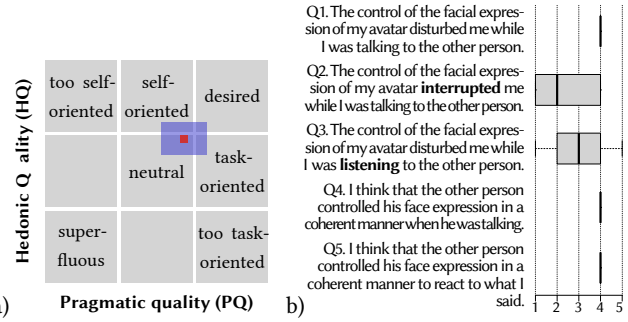

Figure 8: a) Portfolio-presentation of the AttrakDiff Questionnaire results; b) Box plot of the Likert-scale questionnaire's results, from 1 (strongly disagree) to 5 (strongly agree).

not in the "desired" region, EmoRayE appears overall pleasant to use (H2).

Usage while talking. We asked the participants five questions about how they felt when using EmoRayE while discussing with the experimenter. Each question presented a sentence to which the participant had to indicate their level of agreement or disagreement, using a five-level Likert scale. The sentences are listed in Figure 8. In the analysis, the answers were scaled from 1 (strongly disagree) to 5 (strongly agree).

The participants mostly agreed to the first sentence ($M_{Q1} = 4$), so we cannot confirm H3. At least half of the participants disagreed with the second sentence ($M_{Q2} = 2$). Further experiments would be required to clearly confirm H4. Participants either agreed or disagreed on the third sentence ($M_{Q3} = 3$), so we cannot conclude about H5. The participants mostly agreed to the last two sentences ($M_{Q4} = 4$, $M_{Q5} = 4$), so the potential disturbance and interruption of the discussion does not seem to be noticeable for an external person. From the experimenter point of view, the disturbance and interruption of the participants were hardly noticeable.

Discussion. Taken together, these results suggest that EmoRayE appears easy to use in an ecological context and that it can disturb users while they are talking or listening but with interruptions that appear limited. Moreover, the participants found the experimenter controlled his face in a coherent manner when he was talking, so we can expect that more training would help users master the technique and reduce disturbance and interruptions.

Informal observations of the experiment shows that four participants often kept the menu open, and were using it only when necessary. This was probably to reduce the time taken to activate an expression, since the menu was already open when it was needed.

7 DISCUSSION

The results from the different experiments allow us to make recommendations for the design of non-isomorphic techniques to control facial expressions. We first designed two menu-based techniques and two other techniques relying on gestures and voice. According to the results of the first experiment, menu-based techniques show greater preferences and perform better than the two other techniques, particularly in terms of error rate that remains high for the voice recognition system, in spite of the use of a commercial system. Even if the techniques based on gestures and voice recognition allow eyes-free interaction, they require to learn words and gestures and the use of the voice technique is more difficult to use while talking.

Regarding the layout of the menu-based techniques, the circular menu was preferred over the grid menu. It helps to organize expressions based on the emotions they represent. However, abstract expressions hardly fit into this layout. In this case, a grid representation remains an appropriate solution.

Participants prefer menus in front of them rather than attached to the controller. The latter condition forces the users to look down to the controller or to raise it to comfortably see the menu, forcing them to look away the main subject of interest (*e.g.* another person) in the VR application. The size of the items on the menu and the distance between the user and the menu were tweaked during informal pilot studies, to balance between a good visibility of the items and the occlusion of the user's field of view. Displaying the menu at 0.6 m from the user's eyes and using target size of 9 cm, could be used as a good starting point, considering their low error rate. As expected, selection time increases with the number of expressions in the menu, but not in a linear fashion, as the increase of selection time was only 30% when transitioning from 8 and 24 items for RayMoji. Overall designers should favor fewer items in the menus to reduce selection time.

For the control of the expression intensity in real time, we recommend the use of the *Virtual elastic band*, but the *Trigger button* and *Controller orientation* are also good alternatives. Participants found these techniques allow a precise control of the intensity over *Shake* and *Pie shaped menu*.

The final technique we designed, EmoRayE showed a very good usability in the ecological study. Even if the participants felt the technique could disrupt them while talking, this could not be noticed by the experimenter. Taken together the results suggest EmoRayE could be readily used in VR social applications.

Due to the COVID-19 situation, our experiments have limitations in terms of diversity and number of participants. Only two out of twelve participants of the first experiment were female. Should gender have an effect on the results, it cannot be investigated. The third experiment has only nine participants, which is a lower bound for this type of experiment. Including more participants would provide more insights.

8 CONCLUSION

We presented a set of non-isomorphic interaction techniques to control facial expressions of avatars in VR. A controlled experiment allowed us to compare their performance and user preferences for the selection of an expression. This let us combine the technical components of the best performing techniques to propose EmoRay. We also proposed different techniques to control the intensity of the expressions that were evaluated in a controlled experiment showing that Virtual elastic was among the best techniques. EmoRay with Virtual elastic was evaluated in an ecological study, showing a good usability and minimal disturbance of participants, confirming it can be used in social-related applications.

As future work, we would like to explore the combined used of isomorphic and non-isomorphic techniques, where isomorphic techniques would be used for simple expressions and non-isomorphic

techniques supplementing or replacing them when necessary. We also plan to evaluate the impact of the use of facial expressions in applications like meetings in VR and virtual conferences.

ACKNOWLEDGMENTS

This work was supported by the Inria IPL Avatar project.

REFERENCES

[1] Jessalyn Alvina, Chengcheng Qu, Joanna McGrenere, and Wendy E. Mackay. 2019. MojiBoard: Generating Parametric Emojis with Gesture Keyboards. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI EA '19)*. Association for Computing Machinery, Glasgow, Scotland Uk, 1–6. https://doi.org/10.1145/3290607.3312771

[2] Apple. 2017. Animoji. https://support.apple.com/en-us/HT208190 https://hubs.mozilla.com/.

[3] Bernhard Bittorf and Charles Wuethrich. 2012. EmotiCon Interactive emotion control for virtual characters. In *Proceedings of the 20th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG' 2012)*.

[4] John Brooke. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[5] Paul Ekman and Wallace V. Friesen. 1978. *Facial action coding systems*. Consulting Psychologists Press.

[6] Facebook. 2019. Facebook Spaces. https://web.archive.org/web/20191005002238/https://www.facebook.com/spaces Last accessed July 15th, 2021.

[7] Scott W. Greenwald, Zhangyuan Wang, Markus Funk, and Pattie Maes. 2017. Investigating Social Presence and Communication with Embodied Avatars in Room-Scale Virtual Reality. In *Immersive Learning Research Network*, Dennis Beck, Colin Allison, Leonel Morgado, Johanna Pirker, Foaad Khosmood, Jonathon Richter, and Christian Gütl (Eds.). Springer International Publishing, 75–90.

[8] Marc Hassenzahl, Michael Burmester, and Franz Koller. 2003. AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität. In *Mensch & computer 2003*. Springer, 187–196.

[9] Jennifer Hyde, Elizabeth J. Carter, Sara Kiesler, and Jessica K. Hodgins. 2015. Using an Interactive Avatar's Facial Expressiveness to Increase Persuasiveness and Socialness. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1719–1728. https://doi.org/10.1145/2702123.2702465 event-place: Seoul, Republic of Korea.

[10] Hao Li, Laura Trutoiu, Kyle Olszewski, Lingyu Wei, Tristan Trutna, Pei-Lun Hsieh, Aaron Nicholls, and Chongyang Ma. 2015. Facial Performance Sensing Head-Mounted Display. *ACM Trans. Graph.* 34, 4, Article 47 (July 2015), 9 pages. https://doi.org/10.1145/2766939

[11] J. Lugrin, D. Zilch, D. Roth, G. Bente, and M. E. Latoschik. 2016. FaceBo: Real-time face and body tracking for faithful avatar synthesis. In *2016 IEEE Virtual Reality (VR '16)*. 225–226. https://doi.org/10.1109/VR.2016.7504735

[12] Albert Mehrabian. 1996. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in Temperament. *Current Psychology* 14, 4 (Dec. 1996), 261–292. https://doi.org/10.1007/BF02686918

[13] Mozilla. 2020. Hubs by Mozilla. https://hubs.mozilla.com/ Last accessed July 15th, 2021.

[14] Mirror Networking. 2021. Open Source Networking for Unity. https://mirror-networking.com/, retrieved July 12th, 2021.

[15] M. Obaid, R. Mukundan, M. Billinghurst, and C. Pelachaud. 2010. Expressive MPEG-4 Facial Animation Using Quadratic Deformation Models. In *2010 Seventh International Conference on Computer Graphics, Imaging and Visualization (CGIV '10)*. 9–14. https://doi.org/10.1109/CGIV.2010.11

[16] Soo Youn Oh, Jeremy Bailenson, Nicole Krämer, and Benjamin Li. 2016. Let the Avatar Brighten Your Smile: Effects of Enhancing Facial Expressions in Virtual Environments. *PLoS ONE* 11, 9 (2016), 18 pages. https://doi.org/10.1371/journal.pone.0161794

[17] Igor S. Pandzic and Forchheimer Robert (Eds.). 2003. *MPEG-4 Facial Animation: The Standard, Implementation And Applications*. John Wiley & Sons, Inc. https://doi.org/10.1002/0470854626

[18] Robert Plutchik. 1980. Chapter 1 - A General Psychoevolutionary Theory of Emotion. In *Theories of Emotion*, Robert Plutchik and Henry Kellerman (Eds.). Academic Press, 3–33. https://doi.org/10.1016/B978-0-12-558701-3.50007-7

[19] Robert Plutchik. 2001. The Nature of Emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American Scientist* 89, 4 (2001), 344–350. https://www.jstor.org/stable/27857503 Publisher: Sigma Xi, The Scientific Research Society.

[20] Henning Pohl, Dennis Stanke, and Michael Rohs. 2016. EmojiZoom: emoji entry via large overview maps. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. Association for Computing Machinery, Florence, Italy, 510–517. https://doi.org/10.1145/2935334.2935382

[21] Rec Room. 2020. Rec Room. https://recroom.com/ Last accessed July 15th, 2021.

[22] Fiorella de Rosis, Catherine Pelachaud, Isabella Poggi, Valeria Carofiglio, and Berardina De Carolis. 2003. From Greta's mind to her face: modelling the dynamics of affective states in a conversational embodied agent. *International Journal of Human-Computer Studies* 59, 1 (July 2003), 81–118. https://doi.org/10.1016/S1071-5819(03)00020-X

[23] Jeff Sauro. 2011. *A practical guide to the system usability scale: Background, benchmarks & best practices*. Measuring Usability LLC.

[24] Clemens Sielaff. 2010. EmoCoW: An Interface for Real-time Facial Animation. In *ACM SIGGRAPH ASIA 2010 Sketches (SA '10)*. ACM, New York, NY, USA, 40:1–40:2. https://doi.org/10.1145/1899950.1899990

[25] Katsuhiro Suzuki, Fumihiko Nakamura, Jiu Otsuka, Katsutoshi Masai, Yuta Itoh, Yuta Sugiura, and Maki Sugimoto. 2016. Facial Expression Mapping Inside Head Mounted Display by Embedded Optical Sensors. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16 Adjunct)*. ACM, New York, NY, USA, 91–92. https://doi.org/10.1145/2984751.2985714

[26] Unicode. 2020. Unicode® Emoji Charts v13.0. https://unicode.org/emoji/charts/ Last accessed July 15th, 2021.

[27] Vive. 2019. HTC Vive VR headset. https://www.vive.com/us/product/vive-virtual-reality-system/, retrieved January 7th, 2019.

[28] VRChat Inc. 2020. VRChat. https://vrchat.com/ Last accessed July 15th, 2021.

[29] Thibaut Weise, Sofien Bouaziz, Hao Li, and Mark Pauly. 2011. Realtime Performance-based Facial Animation. In *ACM SIGGRAPH 2011 Papers (SIGGRAPH '11)*. ACM, New York, NY, USA, 77:1–77:10. https://doi.org/10.1145/1964921.1964972

[30] Jacob O. Wobbrock, Andrew D. Wilson, and Yang Li. 2007. Gestures without Libraries, Toolkits or Training: A $1 Recognizer for User Interface Prototypes. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology* (Newport, Rhode Island, USA) *(UIST '07)*. Association for Computing Machinery, New York, NY, USA, 159–168. https://doi.org/10.1145/1294211.1294238

[31] Anthony Zhang, Arvind Chembarpu, Kevin Smith, Kamus Hadenes, Sarah Braden, Bohdan Turkynewych, Steve Dougherty, and Broderick Carlin. 2021. SpeechRecognition. https://pypi.org/project/SpeechRecognition/ Last accessed July 15th, 2021.