Controlling Media Player with Hands: A Transformer Approach and a Quality of Experience Assessment

ALESSANDRO FLORIS, SIMONE PORCU, and LUIGI ATZORI, DIEE, University of Cagliari, Italy and CNIT, University of Cagliari, Italy

In this article, we propose a Hand Gesture Recognition (HGR) system based on a novel deep transformer (DT) neural network for media player control. The extracted hand skeleton features are processed by separate transformers for each finger in isolation to better identify the finger characteristics to drive the following classification. The achieved HGR accuracy (0.853) outperforms state-of-the-art HGR approaches when tested on the popular NVIDIA dataset. Moreover, we conducted a subjective assessment involving 30 people to evaluate the Quality of Experience (QoE) provided by the proposed DT-HGR for controlling a media player application compared with two traditional input devices, i.e., mouse and keyboard. The assessment participants were asked to evaluate objective (accuracy) and subjective (physical fatigue, usability, pragmatic quality, and hedonic quality) measurements. We found that (i) the accuracy of DT-HGR is very high (91.67%), only slightly lower than that of traditional alternative interaction modalities; and that (ii) the perceived quality for DT-HGR in terms of satisfaction, comfort, and interactivity is very high, with an average Mean Opinion Score (MOS) value as high as 4.4, whereas the alternative approaches did not reach 3.8, which encourages a more pervasive adoption of the natural gesture interaction.

CCS Concepts: • **Human-centered computing → HCI design and evaluation methods**; **Gestural input**; **Laboratory experiments;**

Additional Key Words and Phrases: Human–computer interface, hand gesture recognition, quality of experience, transformer neural network, media player

ACM Reference Format:
Alessandro Floris, Simone Porcu, and Luigi Atzori. 2024. Controlling Media Player with Hands: A Transformer Approach and a Quality of Experience Assessment. *ACM Trans. Multimedia Comput. Commun. Appl.* 20, 5, Article 132 (January 2024), 22 pages.
https://doi.org/10.1145/3638560

1 INTRODUCTION

The digital world is becoming increasingly heavily present in our everyday activities, which is heading towards the "metaverse", a complete virtual environment linked to the real physical world [30]. Key technologies, such as **extended reality (XR)**, **artificial intelligence (AI)**, and

This work has been partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, "Sustainable Mobility Center" (Centro Nazionale per la Mobilit Sostenibile [CNMS], grant no. CN_00000023), and by the PON "Ricerca e Innovazione" 2014–2020 (PON R&I)"Azione IV.4 Dottorati e contratti di ricerca su tematiche dell'innovazione" with D.M. 1062 on 10.08.2021.

Authors' address: A. Floris, S. Porcu, and L. Atzori, DIEE, University of Cagliari, via Marengo 2, Cagliari, 09123, Italy and CNIT, University of Cagliari, via Marengo 2, Cagliari, 09123, Italy; e-mails: {alessandro.floris84, simone.porcu, l.atzori}@unica.it.

ACM 1551-6857/2024/01-ART132
https://doi.org/10.1145/3638560

computer vision (CV), have been extensively used to enable satisfactory interactions of humans with digital objects in the virtual world. Human–computer interaction technologies are of utmost importance in this regard, since they account for the research in the design, utilization, and implementation of computer systems by humans as well as for the interaction between the two worlds [23].

In particular, the last few years have seen increasing interest in the tools for multimodal interaction as an alternative to traditional **human–computer interfaces (HCIs)**, e.g., touch screens, mice, and keyboards. Multimodal interaction provides the users with choices in how they can interact with a computer by supporting different forms of natural inputs, such as speech, gestures, touch, facial expressions, and eye gaze. These interfaces are also called ***perceptual user interfaces (PUIs)*** because they support unobtrusive interaction through sensors placed in the physical environment rather than on the user [21].

Among these *non-tangible* interfaces, in this study, we specifically focus on hand gestures, which is a natural form of human communication used as an alternative or as a support of verbal communication [4]. Using the hands as a device provides an attractive and natural alternative to traditional HCIs. It can also help users communicate with computer systems in a more intuitive way and with no need for input devices [34]. CV-based systems are commonly used to design touchless HCIs relying on vision-based **hand gesture recognition (HGR)** applications. Despite the advancements provided by modern powerful AI-based systems, automatic HGR is still a challenging task, mostly because of the diversity in how people perform the gestures, which makes gesture classification difficult [25]. However, the performance of hand gesture–based systems is promising. For this reason, they have been considered for different applications, such as interacting with **three-dimensional (3D)** objects [40], commanding in-vehicle systems [38], playing video games [32], and controlling media players [2, 33].

The contribution of this article is twofold. Firstly, we propose a novel **Deep Transformers neural network HGR system (DT-HGR)** for controlling a media player. We were motivated by the limitations of existing HGR systems for controlling a media player, which are trained using poor datasets (in terms of the number of samples and image quality), do not consider hand skeleton data for hand detection (lack of robustness), and do not employ state-of-the-art deep neural networks for hand recognition [2, 3, 18, 29, 33]. The proposed DT-HGR performs hand gesture detection by processing each acquired video frame to identify the hand skeleton and extract hand features. The subsequent gesture recognition algorithm relies on a completely novel approach that exploits separate transformers for each hand's finger feature set to better identify the finger characteristics exploited in the following classification activity. The performance of the proposed DT-HGR, in terms of HGR accuracy, is evaluated on the popular NVIDIA Dynamic Hand Gesture Dataset [25] and compared with state-of-the-art HGR approaches tested on the same dataset.

Secondly, we present a subjective quality assessment to evaluate the performance of the proposed DT-HGR in a practical application for controlling a media player using six selected hand gestures. Despite the central role assumed in the last few years by the **Quality of Experience (QoE)** to measure the subjective quality perceived for multimedia-based services [22], the studies evaluating the QoE for CV-based HCIs are limited. Indeed, the majority of these studies only evaluated objective performance related to the HGR task (e.g., accuracy), by neglecting the subjective QoE perceived by the users that may heavily impact the acceptability of the new HCIs in these application scenarios. For these reasons, we carefully designed and conducted a subjective quality assessment involving 30 people to evaluate the QoE of the users regarding the control of the media player using three different HCIs, two relying on traditional well-known input devices (i.e., mouse and keyboard) and one based on the proposed DT-HGR. During the assessment, the participants were asked to evaluate four subjective metrics: physical fatigue, usability, pragmatic quality, and hedonic quality. An objective metric was also considered to measure the accuracy of the proposed HGR system. Finally, we analyzed the collected data and discussed the obtained results.

The article is structured as follows. Section 2 discusses the related work on HGR and QoE assessment of hand gesture–based HCIs. In Section 3, we present the considered scenario, describe the proposed DT-HGR solution, and discuss the achieved performance. Section 4 describes the designed quality assessment methodology. Section 5 discusses the QoE results. Section 6 presents our conclusions.

2 RELATED WORK

2.1 Hand Gesture Recognition

Hand gesture recognition (HGR) refers to the process of identification and tracking of explicit human gestures to their representation for commanding computer systems [34]. There are two main methods to perform HGR: contact-based devices and CV systems.

Contact-based devices are devices that need physical interaction with the user, such as data gloves, wearables, accelerometers, and multi-touch screens. Being in contact with the user, these devices are very precise in detecting the user's movement, but they can be intrusive and uncomfortable, and the users need to be accustomed to them [23].

CV systems, instead, use one or more cameras to capture the user's movements, which are then analyzed to detect hand motion and gestures. Although these systems are more user friendly, there are several challenges to overcome, such as robustness (detecting the hand within the image under different lighting conditions and background), accuracy (detecting the correct hand gesture), and real-time processing (ability to analyze the image at the frame rate of the input video) [14].

The two main components of CV-based HGR systems are the hand detector and the gesture recognizer. The hand detector detects the hand image within the original captured images and extracts hand gesture features. Traditional image-based algorithms (e.g., the Viola-Jones algorithm) were commonly used to extract image-based hand gesture features, such as shape and color. However, these methods have shown some major limitations because (i) appropriate lighting conditions are needed; (ii) they are computationally slow; and (iii) the nature of hand gestures is linked to the spatial motion relationships of the hand joints [14].

Therefore, state-of-the-art methods for hand detection utilize AI-based techniques to extract hand skeleton information [5], which can also be fused with color and depth image information to enhance gesture recognition performance [14]. The features provided by the hand detector are the input of the gesture recognizer, which uses AI-based techniques to classify the extracted features for the recognition of different hand gestures. **Convolutional Neural Networks (CNNs)** were commonly used for gesture recognition because they are very fast and effective, in particular, compared with static gestures whose pattern does not change during the execution time [16, 46].

However, CNNs are not as effective for the recognition of dynamic hand gestures, which, in addition to spatial features (as for the static gestures), also include temporal features. For this reason, recurrent neural networks, such as **Long Short-Term Memory (LSTM)**, are preferred for HGR of dynamic gestures in that they can keep track of arbitrary long-term dependencies in the input sequences. Indeed, most of the state-of-the-art HGR methods rely on the combination of a CNN with an LSTM [14, 27].

Recently, transformer-based neural networks have started to be used in the CV field to replace traditional recurrent modules (e.g., LSTM) because they are designed to process sequential input data all at once; the attention mechanism provides context for any position in the input sequence [24]. The authors of [12] are among the first to propose a transformer-based architecture to classify dynamic hand gestures. The proposed model, which can process an input sequence of variable length and outputs the gesture classification, achieved state-of-the-art results.

Concerning HGR solutions specifically designed for controlling media players, the literature is limited and based on outdated methods, as follows. The approaches in [2, 3, 29, 33] lack robustness because the hand detection task was implemented using image-based analysis (e.g., color thresholding [2], Viola-Jones algorithm [3]) for the extraction of the hand features. Therefore, the hands may not be detected if the light setting is different from that used to train the classifier.

Traditional classifiers were also used for the HGR task, such as **k-Nearest Neighbor (k-NN)** [33], **support vector machine (SVM)** [2], decision tree [29], and Haar classifier [3]. A CNN architecture was used in [18, 26], but only a few static hand gestures (2 in [18] and 6 in [26]) were considered to control the media player. Moreover, none of these studies has evaluated its approach on state-of-the-art datasets.

It is evident that the HGR systems for controlling media players found in the literature have significant limitations concerning both hand detection and hand recognition tasks. For this reason, inspired by the most recent approaches for HGR [12, 14], we propose a novel transformer-based HGR application for controlling media players. In contrast to [12], in which a single transformer is used for all hand gesture features, our DT-HGR system exploits 5 separate transformers, one for each hand's finger feature set, to better identify the finger characteristics and to better feed the following classification procedure. The proposed approach is described in Section 3.

2.2 QoE Assessment of Hand Gesture-Based HCIs

The QoE is defined as *"the degree of delight or annoyance of the user of an application or service"* [22]. By reflecting the subjective quality perceived by the user, the evaluation of the QoE for multimedia services is of utmost importance although it is not straightforward. Indeed, the QoE is affected by three types of influence factors, namely, system, human, and context, which, in turn, are composed of several different **perceptual dimensions (PDs)** [6, 22, 43].

The studies assessing the QoE of HCIs based on hand gestures are limited in the literature. Also, there is not a standard methodology for their assessment as, for example, the ITU-T P.910 [17] guidelines for the assessment of video quality. For this reason, each literature study assessed the QoE following a different methodology and considering different PDs.

The usability PD and acceptability PD were considered in [11] to evaluate the proposed gesture-controlled video player interface for mobile devices compared with standard buttons. A subjective experiment was conducted involving 15 people, who were asked to indicate the preferred interaction method and to judge how satisfying, tiring, and responsive the new interface was. The performance of the new interface was also measured objectively in terms of task execution time and the number of erroneous gestures performed.

In [40], the user experience is assessed by conducting two different experiments involving a total of 19 subjects. The first experiment involved the mouse versus near field hand gestures as the HCIs and consisted of the exploration of 3D objects presented on a stereoscopic display using actions such as rotation, zooming in and out, or pointing to a specific part of the volume. The second experiment involved the Wii controller versus far field hand gestures and consisted of controlling an icon by moving the hand in 4 directions. User experience was assessed in terms of physical fatigue for various upper body parts (using the Borg scale [9]), usability (performance, fun, and ease of learning), and pragmatic and hedonic qualities (using 21 semantic differential items on a 7-point response scale).

In [38], the usability and user experience of a hand gesture–based in-vehicle system as perceived by elderly people is investigated. The subjective expectations and the subjective experience were evaluated respectively before the system usage through the $SUXES_i$ questionnaire and after the system usage through the $SUXES_f$ questionnaire [39], whereas the **System Usability Scale (SUS)** questionnaire was used to evaluate the post-test perceived usability of the system [10].

The study in [32] compared the utilization of the Leap Motion controller with standard keyboard and mouse controls for playing video games. A total of 15 participants were recruited to play two games and assess the experience in terms of usability, engagement, and personal motion control sensitivity. The SUS was used to rate the subjective experience of the system. Also, the participants had to select the device they considered to be most fun to use, the device they would choose for longer gaming sessions, the overall preferred mode, and the easiest gestures.

The gesture-based user experience for controlling the gameplay of three simple video games was evaluated in [31] using the **User Experience Questionnaire (UEQ)** [20], which consists of 26 total items classified into 6 different scales. Participants were also asked during the post-experiment interview about how natural they thought each gesture was and to rate the reliability of each game's gestures. Furthermore, a performance score was defined to compute the percentage of correctly performed gestures (intended actions) compared with misses (no action) or errors (unintended actions).

Finally, a general model to evaluate the usability of gesture-based interfaces is proposed in [7], which considers the main factors that determine the quality of a gesture (i.e., fatigue, naturalness, accuracy, and duration) and how to measure them.

Concerning the HGR applications for controlling media players discussed in Section 2.1, most of the studies [18, 29, 33] only evaluated objective performance related to the HGR task. A subjective quality assessment was only conducted by the authors of [2] and [3].

In [2], 15 persons were asked to rate their experience with the proposed HGR application using a 4-point scale: poor, satisfactory, good, and excellent. A total of 70% of persons rated their experience as good.

In [3], 17 individuals were asked to perform simple tasks on the computer, such as watching videos, listening to songs, and viewing images. They had to use a mouse, a keyboard, and the proposed hand gesture–based interface. They were asked which option they found more convenient for controlling the applications: 47.06% preferred using the hand gesture interface, 29.41% preferred using the keyboard, and 23.53% preferred using the mouse.

To gain a better understanding of the user-perceived quality when using this HCI and to identify current limitations, we designed and conducted a quality assessment including specific tasks to motivate participants to utilize the hand gestures, and we considered both objective and subjective quality metrics for the evaluation. The details of the assessment are described in Section 4; the QoE results are discussed in Section 5.

3 PROPOSED HGR SOLUTION FOR CONTROLLING MEDIA PLAYERS

We introduce the considered scenario in Section 3.1, describe the proposed DT-HGR solution in Section 3.2, and discuss the achieved performance results in Section 3.3.

3.1 Considered Scenario

We considered a scenario in which a person is sitting in front of a display equipped with a camera and wants to use a media player to watch videos or listen to music. The traditional HCIs used for interacting with applications are the mouse, keyboard, and touch screen, which are well known by most people. We want to investigate what would be the user experience when interacting with and controlling the applications with the hands instead of using an end device, such as a mouse or a keyboard. To this end, we designed an HGR solution for controlling a media player application that captures video sequences from the camera placed in front of the user and it is capable of recognizing 6 different hand gestures.

For our HGR application, we selected 6 hand gestures (which are shown in Figure 1) from the dataset in [13], which includes 27 dynamic hand gestures intended for HCIs and captured at high

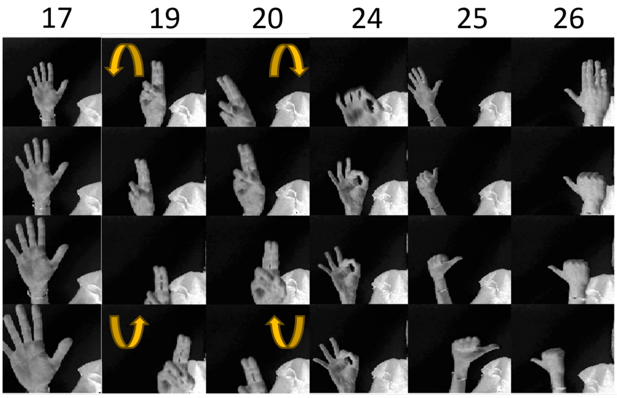

Fig. 1. The six hand gestures selected from the dataset in [13] considered for the proposed DT-HGR system. Gesture 17: stop the playback. Gesture 19: increase the volume (clockwise rotation). Gesture 20: decrease the volume (anticlockwise rotation). Gesture 24: play/pause the playback. Gesture 25: go to the previous content. Gesture 26: go to the next content.

image quality. Most of the gestures in [13] (i.e., the first 25 gestures) were adopted from the NVIDIA popular dataset [25], while the 2 additional hand gestures have been designed to command the playback of previous/next content when controlling media players. There is a difference in the numbering of the gestures between the dataset in [13] and the NVIDIA dataset: the dataset in [13] numbers the gestures starting from 1, whereas the NVIDIA dataset starts from 0. In this article, we refer to the numbering used in the NVIDIA dataset. Thus, the 2 additional gestures introduced by the dataset in [13] are Gesture 25 and Gesture 26.

We selected 6 gestures able to control the media player with some of the most common actions:

- Stop the playback: Gesture 17 (open hand for 3 s).
- Increase the volume: Gesture 19.
- Decrease the volume: Gesture 20.
- Play/Pause the playback: Gesture 24 (“okay” sign).
- Go to the previous content: Gesture 25.
- Go to the next content: Gesture 26.

We have chosen these 6 gestures by considering related literature studies for controlling media players with hand gestures [3, 18, 26, 29, 33]. Although these studies use different gestures for the same player control commands we consider (e.g., play/pause, stop, increase/decrease volume), all considered gestures include using the hand flat and closed fist either statically or dynamically (e.g., moving hand flat and closed fist right-left or up-down). The selected gestures are all included in the NVIDIA dataset [25], which includes 25 gestures adopted from existing commercial systems and popular datasets. Thus, we considered the hand flat and “ok” gestures to stop and start/play the video, respectively; moving the closed fist (right to left and left to right) to go to the previous/next video; and clockwise and counterclockwise movement with two fingers to increase/decrease the volume. The selection of these gestures has been driven by a preliminary survey involving 12 students who were asked to pick the most intuitive gesture among the 25 in [25] for each of the 6 control commands. We then selected the most popular gestures for each one.

3.2 The Proposed Solution

We present the proposed novel Deep Transformers neural network for HGR (DT-HGR). A typical HGR system relies on three main components: scene acquisition, hand detection, and gesture recognition. Once the scene has been acquired, we perform hand detection, which processes each frame to identify the hand skeleton. To this end, we have deeply investigated the state-of-the-art and exploited the most appropriate approach, which has been adapted to the HGR problem at hand. The subsequent gesture recognition algorithm relies on a completely novel approach that exploits separate transformers for each hand's finger feature set to better identify the finger characteristics to feed the following classification.

In the following, we provide the details of the proposed procedure.

3.2.1 Scene Acquisition. The scene acquisition provides a continuous video of the hand gesture at full HD resolution. In our design and testing phases, we used the Logitech Brio Stream 4K HDR video camera to acquire the video scenes at full HD resolution (1920 × 1080) at 30 **frames per second (fps)**.

3.2.2 Hand Detection. As discussed in Section 2.1, the hand detection task is a well-studied problem, and state-of-the-art methods for hand detection extract hand skeleton information [5]. None of the HGR systems for controlling media players found in the literature extracts hand skeleton data; rather, those methods rely on traditional image-based algorithms (i.e., image color analysis [2, 29, 33], the Viola-Jones algorithm [3]) to extract image-based hand gesture features, such as shape and color. However, these solutions have shown some major limitations because (i) appropriate lighting conditions are needed (lack of robustness); (ii) they are computationally slow; and (iii) the nature of hand gestures is linked to the spatial motion relationships of the hand joints [14]. The study in [18] is the only one using a CNN-based solution, which is commonly used to achieve fast and effective recognition of static gestures. However, no dynamic gestures and no hand skeleton data were considered in this study.

For these reasons, for the hand detection task of the proposed HGR system, we relied on the pipeline described in [46], which has been proposed for hand tracking. It is composed of the CNN proposed in [8] trained for palm detection and the CNN Keypoint Detection via Confidence Maps proposed in [35] for hand landmarks detection. These neural networks extract hand-skeleton data and have proven to be lightweight and fast even with low-power devices. For instance, the CNN presented in [8] can execute the inference process in 0.3 s running on an iPhone XS. First, the pipeline identifies the hand's palm, i.e., it selects a boundary box within the video frame including the hand image. Then, this boundary box is processed by the second stage of the pipeline, the CNN Keypoint Detection, which detects 21 hand landmarks as shown in the hand in Figure 2. Each landmark is described by a tuple (x, y), which identifies the landmark's position in the x- and y-axes of the image. As the same gesture may be performed with different "sizes" (e.g., Gestures 19 and 20 of Figure 1 can be performed both doing a "thin" clockwise movement or a "large" clockwise movement of the fingers), the coordinate values have been normalized as in Equation (1):

$$(x^j_{norm}, y^j_{norm}) = \frac{(x^j, y^j) - (x^0, y^0)}{(max(x), max(y))}, \tag{1}$$

where x^j and y^j are the x and y coordinates, respectively, of landmark j, which ranges from 0 to 20. (x^0, y^0) are the coordinates of the reference landmark (shared with all fingers) and $(max(x), max(y))$ are the maximum values of the coordinates of the 21 landmarks. From each video frame, the hand detection component computes 21 normalized landmark coordinate tuples, which are used as the input features for the gesture recognition component.

3.2.3 Gesture Recognition. Figure 2 illustrates how the proposed DT-HGR solution implements the hand gesture recognition and classification tasks. The 21 hand landmarks provided by the hand detector are the input of the Temporal landmark analysis module, which applies a temporal

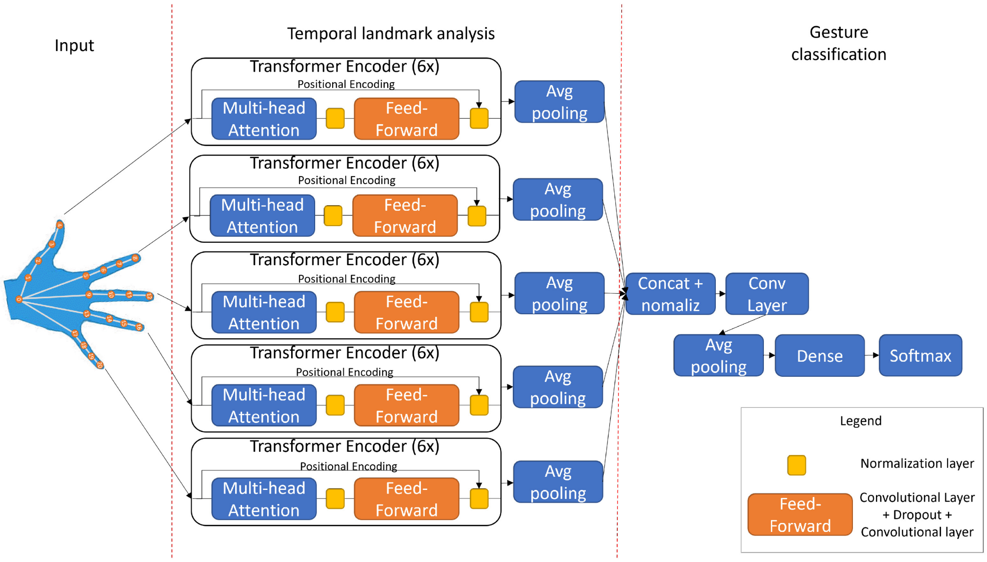

Fig. 2. The proposed DT-HGR solution for gesture recognition and classification.

combination to each set of finger landmarks. In particular, each set of finger landmarks is processed by a single **Transformer Encoder (TE)**, which is composed of the **Multi-head attention (MHA)** layer proposed in [41], two normalization layers, and a Feed-Forward layer (composed of two **one-dimensional [1D]** Convolutional layers with kernel size of 1 and a Dropout layer). Since the hand gestures are the result of subsequent movements of the 5 fingers, the MHA layer aims to detect the most relevant finger landmarks that identify the gesture. Each MHA block is a self-attention layer that repeats its computation 8 times to draw connections between any part of the input sequence. Moreover, each TE repeats its computation 6 times. These numbers of repeating cycles provided the best trade-off between accuracy and computational cost and were determined empirically by employing a greedy algorithm that looked for the optimal parameters' setting in a given search area. The output of the TE is the input of the Avg pooling module, which decreases the feature dimension. The outputs of the 5 Avg pooling modules undergo a concatenation and normalization process followed by a 1D Convolutional layer with a kernel size of 5, an Avg pooling layer for feature reduction, and, finally, a Dense layer and a Softmax activation function that classifies the gesture label.

In the following, we provide more details about the data processing. The input **I** is the concatenation of 5 matrices $\mathbf{F_i}$, with $i = \{1, 2, \ldots, F\}$, where $F = 5$ is the number of fingers. Each matrix $\mathbf{F_i}$ has shape $\{N \times L_i\}$, where N is the total number of video frames of the recorded gesture and L_i is the number of landmarks for finger i. Note that landmark 0 is shared between the 5 fingers and $L_i = 5$ for each finger. Each row of the matrix $\mathbf{F_i}$, defined as f_i^n, is the concatenation of the coordinates of landmarks $l_{i,k}^n$ of the finger i for the video frame n, with $n = \{1, 2, \ldots, N\}$. The index k identifies the finger landmark depending on the finger i, as defined in Equation (2):

$$f_i^n = l_0^n \oplus l_{i,k}^n \oplus l_{i,k+1}^n \oplus l_{i,k+2}^n \oplus l_{i,k+3}^n \ : \left\{ \begin{pmatrix} i = 1 \Rightarrow k = 1 \\ i = 2 \Rightarrow k = 5 \\ i = 3 \Rightarrow k = 9 \\ i = 4 \Rightarrow k = 13 \\ i = 5 \Rightarrow k = 17 \end{pmatrix} \right\}. \tag{2}$$

As an example, for the video frame 1, the arrays f_i^1 are defined as $f_1^1 = \{l_0^1, l_{1,1}^1, l_{1,2}^1, l_{1,3}^1, l_{1,4}^1\}$, $f_2^1 = \{l_0^1, l_{2,5}^1, l_{2,6}^1, l_{2,7}^1, l_{2,8}^1\}, \ldots, f_5^1 = \{l_0^1, l_{5,17}^1, l_{5,18}^1, l_{5,19}^1, l_{5,20}^1\}$.

Then, the $\mathbf{F_i}$ matrix describes the movements of finger i through time during the gesture execution. It is defined as the concatenation of the f_i^n arrays:

$$\mathbf{F_i} = {f_i}^0 \oplus {f_i}^1 \oplus {f_i}^2 \oplus \cdots \oplus {f_i}^N. \tag{3}$$

For instance, for finger 1, the matrix $\mathbf{F_1}$ is defined as $\mathbf{F_1} = \{l_0^1, l_{1,1}^1, l_{1,2}^1, l_{1,3}^1, l_{1,4}^1\} \oplus \ldots \{l_0^2, l_{1,1}^2, l_{1,2}^2, l_{1,3}^2, l_{1,4}^2\} \oplus \{l_0^N, l_{1,1}^N, l_{1,2}^N, l_{1,3}^N, l_{1,4}^N\}$.

Finally, the input $\mathbf{I}$ is defined as the concatenation of the 5 matrices $\mathbf{F_i}$:

$$\mathbf{I} = F_1 \oplus F_2 \oplus F_3 \oplus F_4 \oplus F_5. \tag{4}$$

The $\mathbf{F_i}$ matrices undergo the **temporal combination (TempComb)** process applied by the TE modules, whose outputs are the input of the **Avg pooling (AvgPool)** for feature reduction:

$$\mathbf{TC_i} = TempComb(\mathbf{F_i}) = AvgPool(TE(\mathbf{F_i})). \tag{5}$$

The result of the 5 TempComb processes are then normalized and merged in a single matrix $\mathbf{H}$ by means of a **concatenation (Conc)** process:

$$\mathbf{H} = \mathbf{TC_1} \oplus \mathbf{TC_2} \oplus \mathbf{TC_3} \oplus \mathbf{TC_4} \oplus \mathbf{TC_5}. \tag{6}$$

In order to identify the essential features from the landmarks data collected from the 5 fingers, the matrix $\mathbf{H}$ undergoes a **convolutional (Conv)** process, whose result is reduced in size by means of the Avg pooling module:

$$\mathbf{H_c} = AvgPool(Conv(\mathbf{H})). \tag{7}$$

Finally, the Dense layer and the Softmax activation function output the predicted hand gesture label hg_p:

$$hg_p = SoftMax(ReLu(\mathbf{H_c})) \tag{8}$$

3.3 Performance Results

3.3.1 Comparison of HGR Accuracy on the NVIDIA Dataset. We computed the performance of the proposed DT-HGR approach on the NVIDIA dataset, which is widely used by state-of-the-art studies to prove the performance of their HGR methods. We computed the HGR performance training the DT-HGR with 5-fold cross-validation and 70%/30% training/validation rate, resulting in 1,050 training videos and 482 validation videos. Table 1 compares the performance between state-of-the-art methods and the proposed DT-HGR in terms of HGR accuracy when training and testing the algorithms on the NVIDIA dataset [25]. Note that for a fair comparison of state-of-the-art methods with our approach, only the unimodal column should be considered, which reports the accuracy achieved with the unimodal HGR approach (i.e., RGB only). However, for completeness, the multimodal column reports the accuracy achieved by multimodal HGR approaches, which consider a combination of the RGB data with additional sources of image information, such as depth, flow, or **infrared (IR)**. It can be seen that our proposed method achieved a mean HGR accuracy of 0.853, outperforming the HGR accuracy achieved by state-of-the-art unimodal methods. In particular, our solution achieved an 8.8% higher accuracy than that achieved by the transformer-based solution in [12], which used one transformer for all fingers. Thus, considering one transformer for each finger rather than a single transformer for all fingers allowed our approach to achieve a greater accuracy score. Furthermore, our proposed method outperformed 5 of the 9 considered multimodal methods. This is a promising result, which suggests that feeding our system with multimodal image data would likely enhance the HGR performance.

Table 1. Comparison of the HGR Accuracy Achieved by State-of-the-Art HGR Systems and the Proposed DT-HGR When Both Training and Testing are Performed on the NVIDIA Dataset [25]

Method	**Accuracy**	
	Unimodal (Color)	**Multimodal**
HOG + *HOG*^{2} [28]	0.245	0.369 (color + depth)
Simonyan and Zisserman[36]	0.546	0.656 (color + flow)
Wang *et al.* [42]	0.591	0.734 (color + flow)
C3D [37]	0.693	–
R3DCNN [25]	0.741	0.838 (color + depth + flow + IR)
GPM [15]	0.759	0.878 (color + depth + flow + IR)
PreRNN [44]	0.765	0.850 (color + depth)
Transformer [12]	0.765	0.876 (color + depth + normals + IR)
ResNeXt-101 [19]	0.786	–
MTUT [1]	0.813	0.869 (color + depth + flow)
Yu *et al.* [45]	0.836	**0.884** (color + depth)
DT-HGR (Ours)	**0.853**	–

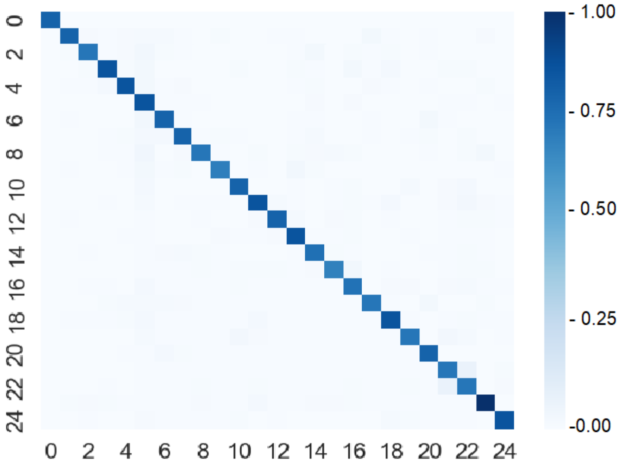

Fig. 3. Confusion matrix achieved by the proposed DT-HGR for the 25 classes of the NVIDIA dataset.

Figure 3 shows the confusion matrix computed for the 25 classes of the NVIDIA dataset with the proposed solution. It can be seen that all gestures are predicted with good accuracy. However, lower accuracy values are achieved for the classification of Gestures 21 and 22. The reason may be that to execute Gesture 21, some users have made some hand movements that are similar to those needed to execute Gesture 22.

3.3.2 HGR Performance on the Dataset in [13]. Firstly, we computed the performance of the proposed DT-HGR approach trained on the NVIDIA dataset (as described in the previous section) and validated on the dataset in [13] (excluding Gestures 25 and 26, which are not present in the NVIDIA dataset). A mean HGR accuracy of 0.808 was achieved; this dataset cross-validation result further demonstrates the validity of the proposed DT-HGR method.

In Figure 4, we show the confusion matrix achieved by the DT-HGR for the first 25 classes (i.e., those adopted by the NVIDIA dataset) of the dataset in [13]. It can be seen that the proposed approach found some difficulties in the classification of Gesture 16, which was confused with

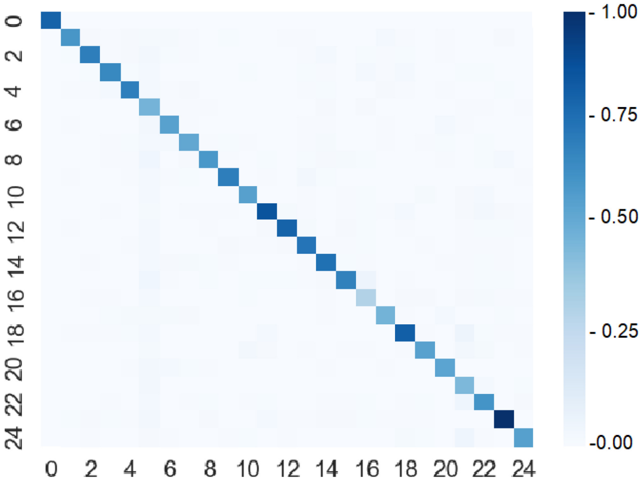

Fig. 4. Confusion matrix achieved by the proposed DT-HGR for the first 25 classes of the dataset in [13] (i.e., those adopted by the NVIDIA dataset).

Gesture 17. The reason is that the “push hand down” and “push hand up” gestures are very similar to each other. Moreover, some executions of these two gestures are very similar to each other because the subjects executed them in the wrong way. Furthermore, Gesture 5 has not achieved classification performance comparable to that of the other gestures because the DT-HGR solution has confused this gesture with other gestures. In addition, lower accuracy values are achieved for the classification of Gestures 21 and 22, as for the NVIDIA dataset. However, on average, our solution achieved great classification results both on the NVIDIA dataset and the dataset in [13] (cross-validation).

Secondly, we computed the performance of the proposed DT-HGR approach training with 5-fold cross-validation and 70%/30% training/validation rate on the dataset in [13]. A mean HGR accuracy of 0.885 is achieved over the 27 gestures of this dataset, which is a greater accuracy than that achieved on the NVIDIA dataset (0.853) although th dataset in [13] has 2 additional gestures. This can be motivated by the higher image quality of video frames collected in this dataset and the more precise gesture execution (participants were monitored during gesture execution and they had to repeat the hand gesture in case the movement was not correct) compared with the NVIDIA dataset. Figure 5 shows the confusion matrix for the 27 classes. Even in this case, Gestures 21 and 22 achieve the lowest accuracy scores. Note that a state-of-the-art comparison with the dataset in [13] (similarly to Table 1) could not be provided because none of the literature studies used this dataset to test their models and no software was publicly provided to allow for computing the models’ performance on the dataset in [13].

Finally, the proposed DT-HGR approach achieves a mean HGR accuracy of 0.983 if only six hand gestures in Figure 1 are considered from the dataset in [13]. Table 2 reports the performance of the proposed HGR solution for these six hand gestures in terms of accuracy, precision, and recall, computed for the single classes. The HGR accuracy achieved for single gestures is greater than 0.96. In particular, except for Gestures 25 and 26, achieving an accuracy of 0.97, the accuracy achieved for single gestures is 0.99. Indeed, Gestures 25 and 26 (“Next/Previous content”) also achieved lower precision and recall values than those obtained by the other gestures, which could derive from the different motion speeds needed to recognize these kinds of gestures. For this reason, the camera could have lost some frames when trying to capture the high-speed movements. However, all

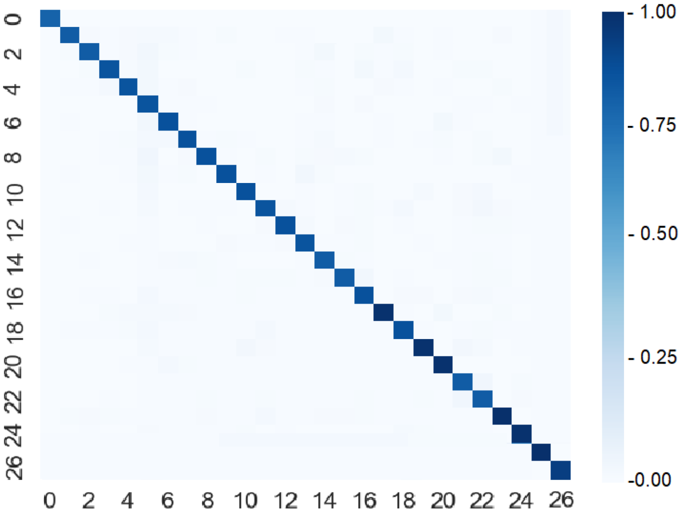

Fig. 5. Confusion matrix achieved by the proposed DT-HGR for the 27 classes of dataset [13].

Table 2. Performance of the Proposed DT-HGR Achieved for the Six Considered Hand Gestures in Terms of Accuracy, Precision, and Recall

Gesture	Accuracy	Precision	Recall
17	0.99	0.98	0.97
19	0.99	0.98	0.98
20	0.99	0.98	0.98
24	0.99	0.98	0.97
25	0.97	0.94	0.92
26	0.97	0.95	0.93

gestures achieved precision and recall results greater than 0.93 and 0.91, respectively, which motivates their utilization for the real-time experiment. By only considering the 6 selected gestures, the DT-HGR approach achieves greater performance than considering the entire dataset. The reason is that the lower the number of gestures to be recognized, the lower the chance for error. Also, the selected gesture movements are well distinguished from each other.

4 QOE ASSESSMENT

In Section 2.2, we discussed the QoE assessments conducted by the state-of-the-art studies. Although there is no standardized methodology to follow, each study focused on specific functionalities of the application under study that involved the utilization of the hand gestures to be evaluated. Concerning the QoE evaluation, there are objective and subjective quality metrics that are common to most of the literature studies. From the objective side, we considered and measured the performance score (PScore), which is defined in [31] as the ratio between the number of correct gestures and the total number of gestures. From the subjective side, four perceptual dimensions were considered: the physical fatigue perceived when performing the hand gesture; the usability of the HCI in controlling the video player; the pragmatic quality in terms of accuracy and responsiveness; and, finally, the hedonic quality in terms of satisfaction, comfort, interactivity,

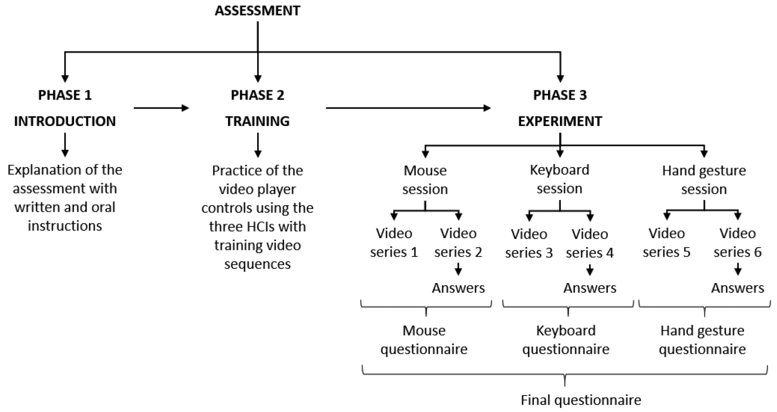

Fig. 6. Phases of the designed quality assessment.

Table 3. Video Player Controls with the 3 Different HCIs for the VLC Video Player

Video player control	Hand gesture	Mouse	Keyboard
Stop	17	Left click on the “Stop” button	S key
Increase volume	19	Mouse wheel up	Arrow up key
Decrease volume	20	Mouse wheel down	Arrow down key
Play/Pause	24	Left click on the “Play/Pause” button	Spacebar key
Previous video	25	Left click on the “Previous video” button	P key
Next video	26	Left click on the “Next video” button	N key

naturalness, and fun. In Section 4.1, we describe the designed assessment methodology. Section 4.2 discusses the evaluation process.

4.1 Methodology

The considered scenario is described in Section 3.1. The designed QoE assessment consists of three phases: Introduction, Training, and Experiment, as illustrated in Figure 6.

During the Introduction phase, which lasted about 10 minutes, the participants were given explanations of the objective and the steps of the assessment. The hand gestures, mouse controls, and keyboard controls needed to control the video player were introduced and explained to the participants. They were given written and oral instructions to understand how to utilize each of the HCIs under test to operate the considered video player controls, which are summarized in Table 3. They were also given explanations regarding the tasks needed to be performed to complete a part of the experiment (answers to questions concerning the content of the watched video). The post-experiment surveys were introduced and explained so that participants understood the quality perceptual dimensions to be evaluated and the rating scales. During this phase, the participants had the opportunity to ask further questions to eliminate any doubts regarding the experiment. Finally, this phase ended with participants providing written consent regarding the publication of the obtained results.

The training phase required test participants to practice the three HCIs under test by operating the video controls they would have to use during the Experiment phase. For the practice session, training videos were created using videos different from those used for the Experiment phase. The objective of this phase was to ensure that the participants acquired a base level of practice before

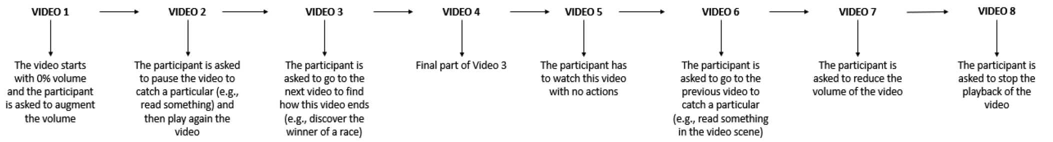
VIDEO 1
VIDEO 2
VIDEO 3
VIDEO 4
VIDEO 5
VIDEO 6
VIDEO 7
VIDEO 8
The video starts with 0% volume and the participant is asked to augment the volume
The participant is asked to pause the video to catch a particular (e.g., read something) and then play again the video
The participant is asked to go to the next video to find how this video ends (e.g., discover the winner of a race)
Final part of Video 3
The participant has to watch this video with no actions
The participant is asked to go to the previous video to catch a particular (e.g., read something in the video scene)
The participant is asked to reduce the volume of the video
The participant is asked to stop the playback of the video

Fig. 7. Example of video series where the participant is asked to perform specific tasks and answer video content-related questions.

judging the HCIs and had already used the controls in similar situations. The participants had to familiarize themselves with each of the 6 video player controls in Table 3 using the three HCIs. They had to perform the controls several times until they were sure they understood how to use them. In particular, most of the time was spent practicing with the hand gesture–based HCI, since each participant already had some experience using a mouse and keyboard to control video players. This phase lasted about 15 minutes.

Finally, during the Experiment phase, the test participants performed the experiment and evaluated the perceived quality. The Experiment phase consisted of three sessions, each performed with a different HCI. The utilization order of the 3 HCIs was chosen randomly for each participant so that it would not affect the data analysis. Each session consisted of 2 video series. During the first video series, the participant had to watch the videos and follow the written instructions appearing on the screen asking to perform the 6 video player controls at specific times. The second video series was similar to the first. However, the participant was also required to pay careful attention to the video content since specific questions were asked to test the participant's attention and ability to use the HCIs to control the video player. Figure 7 shows an example of the tasks required during the second video series. The participant had to write the answers on a sheet of paper. A pause of 5 minutes was taken between each session, which lasted an average of 10 minutes. At the end of each session, the participant had to fill out the subjective questionnaire to evaluate the perceived physical fatigue, usability, pragmatic quality, and hedonic quality using that HCI. At the end of the experiment (all three sessions completed and rated), the final questionnaire had to be completed where the preferences regarding the utilization of the three HCIs were asked.

In Figure 8, we show the experiment environment. The participant was seated in front of the screen behind a desk. The distance from the screen was the same when using each of the three HCIs. The main limitation concerning the workable distance using hand gestures regards the video camera field of view, which determines the maximum distance of the hand gesture recognition from the camera. Similarly, the diagonal field of view determines the maximum workable angles. Even without extensive experimentation, we have determined that the gesture area should be at least 200 × 200 pixels in size when the gesture is in front of the video camera. When considering the typical commercial webcam (full HD, diagonal field of view of 78 degrees), this corresponds to a distance of around 1 meter. The distance can be extended by using zoom and reducing the diagonal field of view. As to the workable angle, we have empirically found that it should not be greater than 45 degrees.

4.2 Evaluation

We evaluated the QoE using both objective and subjective quality metrics. From the objective side, we considered and measured the **performance score (PScore)**, which is defined in [31] as the ratio between the number of correct gestures and the total number of gestures. A correct gesture results in the intended control by the video player. When a gesture is not performed correctly, it can be a *miss* or an *error*. A miss is a gesture that resulted in no action, i.e., a participant's delayed reaction or a wrong movement not recognized by the HGR application that did not result

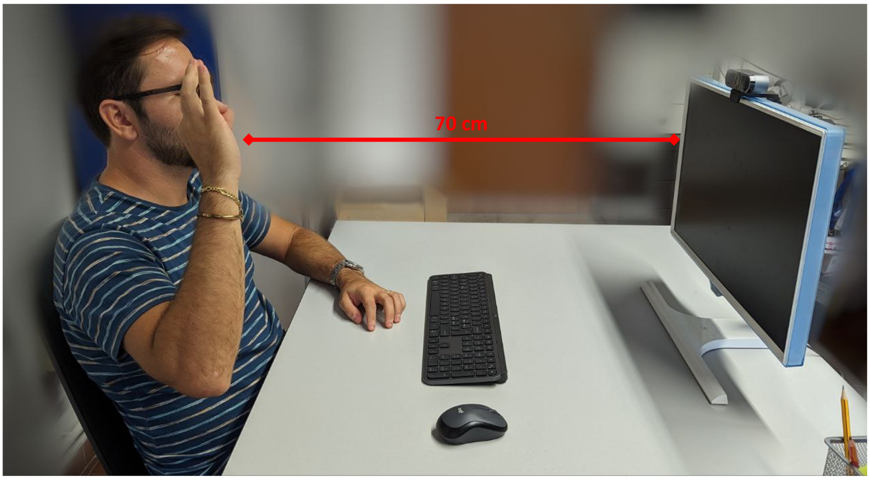

Fig. 8. The experiment environment.

in a control recognized by the video player. An error is a wrong movement that results in an unintended action, such as a different control executed by the video player. We also applied the PScore to mouse and keyboard controls. A miss, in this case, is a participant's delayed reaction whereas an error is the click/press of a wrong button/key. Test participants were observed during the Experiment phase and the number of correct gestures, misses, and errors was noted to compute the PScore for each video player control and each HCI.

The considered subjective metrics are physical fatigue, usability, pragmatic quality, and hedonic quality. At the end of each Experiment session, the participants had to fill out the questionnaire, including the questions concerning these four subjective metrics. The questionnaire was divided into four sections, one for each metric. Physical fatigue regards the level of exertion perceived when performing hand gestures, clicking the mouse buttons, and pressing the keyboard keys, which had to be evaluated using the Borg scale, from 0 to 10 [9]. The SUS questionnaire was used to evaluate the perceived usability of the three HCIs [10]. The SUS is a ten-statement questionnaire the participants had to use to rate the HCIs. It contains a 5-point Likert scale to indicate whether they strongly disagree (0) or strongly agree (4) with each statement. From the ratings of the 10 statements, the SUS score was computed, which ranges from 0 to 100. Usability aspects taken into account by the SUS statements included complexity, confidence, and ease of learning.

The pragmatic quality is defined as "*the extent to which a system allows for effective and efficient goal-achievement, and is closely related to the notion of usability*" [40]. We measured the pragmatic quality in terms of accuracy and responsiveness of the used HCIs. Accuracy concerns the system's ability to perform the action/control intended by the participant whereas the responsiveness includes the system quality and speed of the reaction to the participant's input. Note that other pragmatic quality aspects, such as ease of use and ease of learning, were not considered as already included in the SUS questionnaire. On the other hand, the hedonic quality is defined as "*the extent to which a system allows for stimulation by its challenging and novel character or identification by communicating important personal values*" [40]. Therefore, with the hedonic quality, we aimed to investigate non-task-oriented aspects regarding the perceived quality: satisfaction, comfort, naturalness, interactivity, and fun. Satisfaction regards the perceived level of enjoyment when using the HCI, whereas comfort takes into account how comfortable the HCI is to control the video player. With *interactivity*, we refer to the correlation between the performed gestures and the reactions re-

Table 4. Number of Miss Actions and Errors Observed, and PScore Results Computed for each Video Player Control and Each HCI

Video	Hand gestures			Mouse			Keyboard		
control	# Miss	# Err.	PScore	# Miss	# Err.	PScore	# Miss	# Err.	PScore
Play/Pause	1	1	93.33%	2	0	93.33%	2	0	93.33%
Stop	2	0	93.33%	0	1	96.67%	1	2	90%
Next/Prev.	3	3	90%	0	1	98.33%	3	2	91.67%
Incr./decr.	1	4	91.67%	0	0	100%	0	2	96.67%
Overall	7	8	91.67%	2	2	97.78%	6	6	93.33%

turned by the video player. *Naturalness* indicates how intuitive the HCI is for controlling the video player, whereas *fun* concerns the pleasure and entertainment provided by the HCI. A 5-point Likert scale had to be used by participants to rate the hedonic qualities, from 1 to 5. Participants were also asked whether the used HCIs met their expectations.

At the end of the experiment, i.e., after all three sessions were completed and evaluated by the participants, they were asked to fill out the final questionnaire, in which they had to provide their preferences regarding the utilization of the three HCIs. In particular, they were asked to indicate their absolute favorite HCIs and what HCI they preferred for each of the four kinds of controls: Play/Pause, Stop, Previous/Next video, and Increase/Decrease volume.

5 QOE RESULTS

The quality assessment involved 30 Italian people, 18 males and 12 females, ages 21 to 42 (mean: 25.87, standard deviation: 5.04). Participants were students (20), PhD students (6), and post-doc researchers (4).

5.1 Objective Quality

Table 4 summarizes the number of miss actions and errors observed as well as the PScore results computed for each video player control and each HCI. The mouse achieved the greatest overall PScore (97.78%). Two participants missed the action to pause the video playback at a specific instant to catch a particular scene in the video. They had a delayed reaction. One participant clicked the wrong button to stop the video and another participant clicked the wrong button to go to the next video. The keyboard achieved second place, with an overall 93.33%. In this case, the miss actions were a delayed reaction to pause the video (as for the mouse case), to stop the video, and to play the next and previous video. The 6 errors, instead, were due to pressing the wrong keys, which resulted in unintended actions (2 errors for Stop, 2 for Increase/Decrease volume, and 2 for Next/Previous video). Finally, the hand gestures achieved an overall PScore of 91.67%. The miss actions (1 for Play/Pause, 2 for Stop, 3 for Next/Previous video, and 1 for Increase/Decrease volume) in this case concern the gestures not recognized by the HGR application because they were not performed properly by the participants. On the other hand, all the errors (1 for Play/Pause, 3 for Next/Previous video, and 4 for Increase/Decrease volume) involved the unintended Pause command given by the participants when raising the hand to perform a gesture. Indeed, some of them tended to keep the thumb and index finger close to each other, which was recognized as the “ok” command to pause the video.

5.2 Subjective Quality

Figure 9 shows the mean perceived physical fatigue with the 95% **confidence interval (CI)** for the three HCIs. The physical fatigue for the hand gestures is slightly greater than that perceived

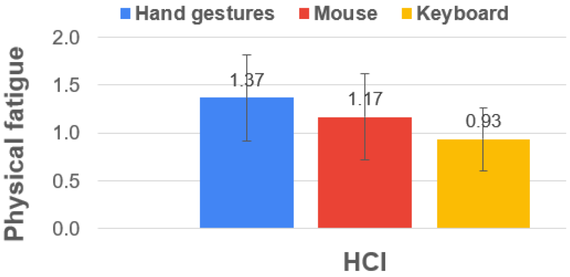

Fig. 9. Mean perceived physical fatigue with the 95% CI for the three HCIs.

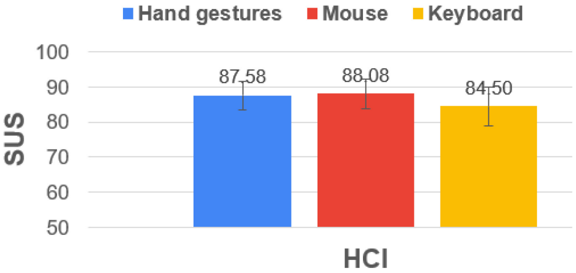

Fig. 10. Perceived usability in terms of SUS with the 95% CI for the three HCIs.

for the mouse, which, in turn, is slightly greater than that perceived for the keyboard. However, these values are very similar and indicate very little fatigue perceived for the three HCIs (around 1 on a scale ranging from 0 to 10). Figure 10 shows the perceived usability in terms of SUS with the 95% CI. The SUS perceived for the hand gestures and the mouse is comparable whereas that perceived for the keyboard is slightly lower. However, all of the perceived SUS are very high, which indicates the usability of the three HCIs is more than acceptable for the participants (over 80 on a scale ranging from 0 to 100). The one-way **Analysis of Variance (ANOVA)** results between the groups of scores provided for the three HCIs have not shown significant differences in terms of physical fatigue and SUS.

Figure 11 shows the **mean opinion score (MOS)** computed for the perceived pragmatic quality (accuracy and responsiveness) and hedonic quality (satisfaction, comfort, interactivity, naturalness, and fun) with the 95% CI. Concerning the pragmatic quality, the results obtained by the three HCIs are quite good and comparable. The hand gestures achieved the lowest perceived accuracy, which, however, corresponds to a good accuracy (3.90). The mouse achieved the greatest perceived accuracy (4.20), slightly greater than that obtained by the keyboard (4.13). In contrast, the keyboard achieved the greatest perceived responsiveness (4.40) whereas those achieved by hand gestures and mouse are comparable (around 4.2). The ANOVA results between the groups of scores provided for the three HCIs have not shown significant differences in terms of accuracy and responsiveness.

Concerning the hedonic quality, the mouse achieved the lowest score for all quality aspects except for naturalness. On the other hand, the hand gestures always achieved the greatest quality. The participants were very satisfied with the experience with the hand gestures (4.40), which resulted in the most enjoyable interface. Instead, sufficient satisfaction was perceived for the keyboard (3.40) and the mouse (3.13). The hand gestures are also perceived as the most comfortable HCI for controlling the video player (4.23), followed by the keyboard (3.73) and the mouse (3.57), which achieved sufficient to good comfort. Concerning the interactivity provided by the HCIs to

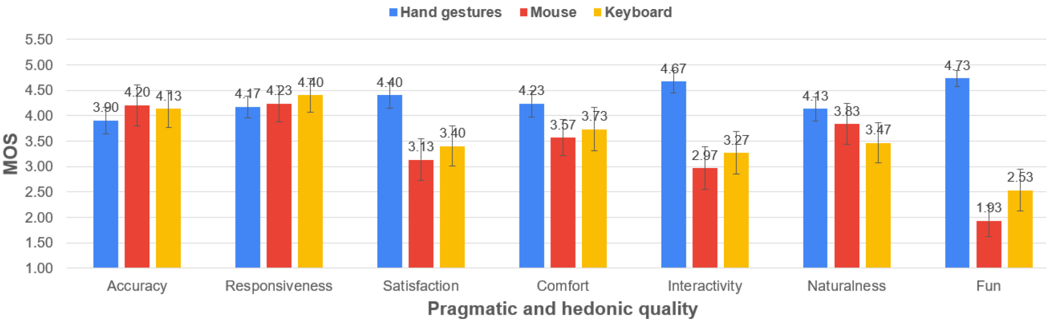

Fig. 11. Mean opinion score (MOS) computed for the perceived pragmatic quality (accuracy and responsiveness) and hedonic quality (satisfaction, comfort, interactivity, naturalness, and fun) with the 95% CI.

control the video player, the hand gestures achieved a very high result (4.67) whereas, again, the keyboard (3.27) and mouse (2.97) achieved just sufficient rates. The naturalness results are, instead, quite comparable among the three HCIs. The hand gestures (4.13) are perceived as the most natural HCI to control the video player, followed this time by the mouse (3.83) and the keyboard (3.47). Finally, hand gestures are the most fun HCI to control the video player (4.73). The keyboard achieved poor to sufficient fun (2.53) whereas the mouse is the least fun HCI, achieving a poor rating (1.93).

We also computed the ANOVA results between the groups of scores provided for the three HCIs, which have shown significant differences in terms of the 5 hedonic qualities. In particular:

- Satisfaction ($p < 0.001$): Significant difference between hand gestures–mouse and hand gestures–keyboard. No significant difference between mouse–keyboard.
- Comfort ($p < 0.05$): Significant difference between hand gestures–mouse. No significant difference between hand gestures–keyboard and mouse–keyboard.
- Interactivity ($p < 0.001$): Significant difference between hand gestures–mouse and hand gestures–keyboard. No significant difference between mouse–keyboard.
- Naturalness ($p < 0.05$): Significant difference between hand gestures–keyboard. No significant difference between hand gestures–mouse and mouse–keyboard.
- Fun ($p < 0.001$): Significant difference between hand gestures–mouse, hand gestures–keyboard, and mouse–keyboard.

The hedonic quality results are confirmed by Figure 12, in which we summarize the preferred HCI for each video player control as well as the overall favorite HCI. The mouse is the favorite HCI only for controlling the volume whereas the keyboard is the preferred HCI only for playing the previous/next videos from the playlist. These results may be motivated by the easiness and user-friendliness of scrolling the mouse wheel to increase/decrease the volume and pressing a single key to play the previous/next video. However, the hand gestures are the overall favorite HCI, the favorite HCI for playing/pausing and stopping the video, and the second favorite HCI for controlling the volume and selecting the next/previous video. Thus, the hand gestures were really appreciated by the test participants to command the video player. The keyboard was also well appreciated for most of the player controls whereas the mouse resulted in the least beloved HCI (except for controlling the volume).

Finally, Figure 13 shows the response to the question "*Has the used HCI met your expectations?*". This result further confirms that the hand gestures HCI were not only well appreciated but exceeded the expectations of almost half of the test participants. On the other hand, the mouse and keyboard mostly met participants' expectations with low surprise.

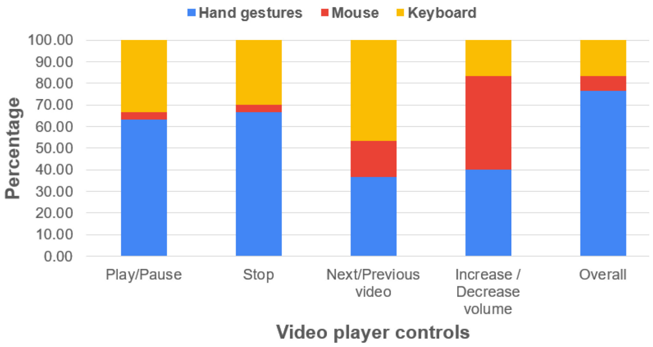

Fig. 12. Preferred HCI for each video player control and overall favorite HCI.

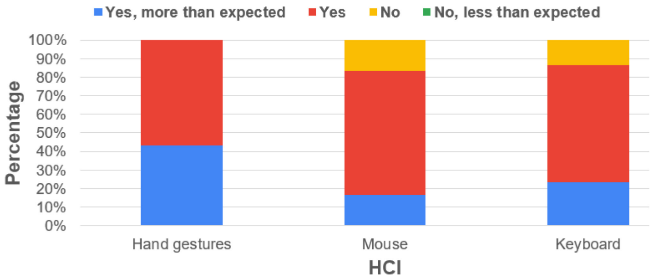

Fig. 13. Response to the question: “Has the used HCI met your expectations?”.

6 CONCLUSION

Hand gesture–based HCIs are a promising means of interaction for current and future multimedia-based applications. In the current study, we focused on the QoE assessment of device-free hand gesture–based controls when compared with traditional HCIs, i.e., the mouse and the keyboard. The application under testing was a video player, and the test participants had to perform common video controls (i.e., play, pause, stop, increase/decrease volume, next/previous video) using the three considered HCIs. We proposed our DT-HGR system to control the video player using hand gestures, whose HGR algorithm relies on a completely novel approach that exploits separate transformers for each hand–finger feature set. The DT-HGR aims to benefit from the transformer architecture to better identify the finger characteristics to drive the following classification. The recognition accuracy of the DT-HGR (0.853) outperforms the accuracy achieved by 11 state-of-the-art HGR approaches tested on the same NVIDIA dataset. Moreover, the DT-HGR achieved an accuracy of 0.808 when cross-validated on the dataset in [13].

The comprehensive evaluation (objective and subjective) conducted in this study highlights the importance of assessing the subjective QoE perceived by the users. Indeed, although the objective performance of the proposed HGR application has shown some minor flaws (e.g., some gestures were not immediately recognized and participants had to repeat the gesture), the subjective results have emphasized the level of satisfaction and appreciation of the hand gestures against the two traditional HCIs. In particular, hand gestures achieved the greatest scores for all pragmatic and hedonic quality dimensions, including satisfaction, comfort, interactivity, naturalness, and fun. Moreover, the hand gestures were chosen as the overall preferred HCI and the favorite HCI to perform half of the considered controls, i.e., play/pause/stop. Furthermore, subjective feedback can be the input for identifying the weak points of the HGR application and understanding how to solve them to enhance the objective performance.

Overall, the obtained results also confirm the current (and likely future) trend that sees the end user willing and able to command high-tech systems in a multimodal device-free way. **Intelligent personal assistants (IPAs)**, such as Google Home and Amazon's Alexa, are examples of devices controlled through vocal commands. Hand gestures can be an alternative or support for vocal commands (e.g., for deaf-mute people) for future IPAs (equipped with a camera) and can be extremely useful for several future applications, such as interacting with smart walls in public spaces to get information; controlling the radio/air conditioning/navigation systems in cars (autonomous driving, in particular); and manipulating objects or controlling virtual/augmented reality environments.

However, widespread use of hand gesture–based systems would require facing some challenges, for example, gesture standardization, system scalability, and technology integration. Gesture standardization is required to avoid the system being retrained for each new use case. However, a dataset such as that collected by NVIDIA [25] already includes a complete set of gestures for controlling multimedia systems (adopted from existing commercial systems and popular datasets), which may be considered as a standard set for implementing hand gesture–based systems for controlling media players. Indeed, most state-of-the-art HGR approaches tested the performance of their methods on this dataset. Concerning scalability, the 25 gestures included in the NVIDIA dataset can be more than enough to control multimedia applications. The reason is that the user should be able to remember the gestures needed to control a multimedia application. Generally, only a selected subset of these 25 gestures are selected to control a specific multimedia application. Moreover, increasing the number of gestures raises the chances of having similar gestures, which may be confused by the HGR system, decreasing the performance. Finally, technology integration requires HGR systems to be supported with simply a camera and software that links the recognized gesture to a specific action. Thus, they are easily extendable to many applications.

In future works, we aim to test and validate the proposed system in different environments for further application use cases. From the experience achieved in the development of the proposed use case, we have obtained and reported some guidelines concerning the determination of the maximum workable distances and angles that would allow the system to work correctly. The proposed HGR system would regularly perform well if workable conditions are respected, i.e., if the user executes the correct gesture in front of a camera despite the type of application to be controlled. These guidelines can be followed to precisely define the workable conditions for each considered environment and application, which requires specific experimental tests. We also aim to investigate the impact of utilizing multimodal image information (e.g., depth, flow) to feed the proposed HGR system and verify whether the performance can be further enhanced.

REFERENCES

[1] Mahdi Abavisani, Hamid Reza Vaezi Joze, and Vishal M. Patel. 2019. Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training. In *2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1165–1174.

[2] Swapna Agarwal and Saiyed Umer. 2015. MP-FEG: Media player controlled by facial expressions and gestures. In *2015 5th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*. 1–4. https://doi.org/10.1109/NCVPRIPG.2015.7490026

[3] Anupam Agrawal, Rohit Raj, and Shubha Porwal. 2013. Vision-based multimodal human-computer interaction using hand and head gestures. In *2013 IEEE Conference on Information Communication Technologies*. 1288–1292. https://doi.org/10.1109/CICT.2013.6558300

[4] Danilo Avola, Marco Bernardi, Luigi Cinque, Gian Luca Foresti, and Cristiano Massaroni. 2019. Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures. *IEEE Transactions on Multimedia* 21, 1 (2019), 234–245. https://doi.org/10.1109/TMM.2018.2856094

[5] Danilo Avola, Luigi Cinque, Alessio Fagioli, Gian Luca Foresti, Adriano Fragomeni, and Daniele Pannone. 2022. 3D hand pose and shape estimation from RGB images for keypoint-based hand gesture recognition. *Pattern Recognition* 129 (2022), 108762.

[6] Jasmina Baraković Husić, Sabina Baraković, Enida Cero, Nina Slamnik, Merima Oćuz, Azer Dedović, and Osman Zupčić. 2019. Quality of experience for unified communications: A survey. *International Journal of Network Management* 30, 3 (2019), 1–25.

[7] Kayne Barclay, Danny Wei, Christof Lutteroth, and Robert Sheehan. 2011. A quantitative quality model for gesture based user interfaces. In *Proc. of the 23rd Australian Computer-Human Interaction Conference* (Canberra, Australia) *(OzCHI'11)*. Association for Computing Machinery, 31–39. https://doi.org/10.1145/2071536.2071540

[8] Valentin Bazarevsky, Yury Kartynnik, Andrey Vakunov, Karthik Raveendran, and Matthias Grundmann. 2019. BlazeFace: Sub-millisecond neural face detection on mobile GPUs. In *CVPR Workshop on Computer Vision for Augmented and Virtual Reality*.

[9] G. A. Borg. 1982. Psychophysical bases of perceived exertion. *Medicine and Science in Sports and Exercise* 14, 5 (1982), 377–381.

[10] J. Brooke. 1996. *Usability Evaluation In Industry* (1st. ed.). CRC Press, Chapter SUS: A 'Quick and Dirty' Usability Scale, 1–6.

[11] Shelley Buchinger, Ewald Hotop, Helmut Hlavacs, Francesca De Simone, and Touradj Ebrahimi. 2010. Gesture and touch controlled video player interface for mobile devices. In *Proceedings of the 18th ACM International Conference on Multimedia* (Firenze, Italy) *(MM'10)*. 699–702. https://doi.org/10.1145/1873951.1874055

[12] Andrea D'Eusanio, Alessandro Simoni, Stefano Pini, Guido Borghi, Roberto Vezzani, and Rita Cucchiara. 2020. A transformer-based network for dynamic hand gesture recognition. In *2020 International Conference on 3D Vision (3DV)*. 623–632.

[13] Graziano Fronteddu, Simone Porcu, Alessandro Floris, and Luigi Atzori. 2022. A dynamic hand gesture recognition dataset for human-computer interfaces. *Computer Networks* 205 (2022), 108781. https://doi.org/10.1016/j.comnet.2022.108781

[14] Qing Gao, Yongquan Chen, Zhaojie Ju, and Yi Liang. 2022. Dynamic hand gesture recognition based on 3D hand pose estimation for human–robot interaction. *IEEE Sensors Journal* 22, 18 (2022), 17421–17430.

[15] Vikram Gupta, Sai Kumar Dwivedi, Rishabh Dabral, and Arjun Jain. 2019. Progression modelling for online and early gesture detection. In *2019 International Conference on 3D Vision (3DV)*. 289–297.

[16] Cristian Iorga and Victor-Emil Neagoe. 2019. A deep CNN approach with transfer learning for image recognition. In *2019 11th Int. Conf. on Electronics, Computers and Artificial Intelligence (ECAI)*. 1–6.

[17] ITU. 2008. Subjective video quality assessment methods for multimedia applications. Recommendation ITU-T P.910.

[18] Eldhose Joy, Sruthy Chandran, Chikku George, Abhijith A. Sabu, and Divya Madhu. 2018. Gesture controlled video player a non-tangible approach to develop a video player based on human hand gestures using convolutional neural networks. In *2018 2nd International Conference on Intelligent Computing and Control Systems (ICICCS)*. 56–60. https://doi.org/10.1109/ICCONS.2018.8662901

[19] Okan Köpüklü, Ahmet Gunduz, Neslihan Kose, and Gerhard Rigoll. 2019. Real-time hand gesture detection and classification using convolutional neural networks. In *2019 14th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2019)* (Lille, France). IEEE Press, 1–8.

[20] Bettina Laugwitz, Theo Held, and Martin Schrepp. 2008. Construction and evaluation of a user experience questionnaire. In *HCI and Usability for Education and Work*, Andreas Holzinger (Ed.). Springer, Berlin, 63–76.

[21] Joseph J. LaViola Jr., Sarah Buchanan, and Corey Pittman. 2014. *Multimodal Input for Perceptual User Interfaces*. John Wiley & Sons, Ltd, Chapter 9, 285–312. https://doi.org/10.1002/9781118706237.ch9

[22] Patrick Le Callet, Sebastian Möller, and Andrew Perkis. 2012. *Qualinet White Paper on Definitions of Quality of Experience (2012)*. European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003), Lausanne, Switzerland, Version 1.2, March 2013.

[23] Yang Li, Jin Huang, Feng Tian, Hong-An Wang, and Guo-Zhong Dai. 2019. Gesture interaction in virtual reality. *Virtual Reality and Intelligent Hardware* 1, 1 (2019), 84–112. https://doi.org/10.3724/SP.J.2096-5796.2018.0006

[24] Shuying Liu, Wenbin Wu, Jiaxian Wu, and Yue Lin. 2022. Spatial-temporal parallel transformer for arm-hand dynamic estimation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1–6.

[25] Pavlo Molchanov, Xiaodong Yang, Shalini Gupta, Kihwan Kim, Stephen Tyree, and Jan Kautz. 2016. Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural networks. In *2016 IEEE Conf. on Comp. Vision and Pattern Recognition (CVPR)*. 4207–4215. https://doi.org/10.1109/CVPR.2016.456

[26] Gayathri Devi Nagalapuram, S. Roopashree, D. Varshashree, D. Dheeraj, and Donal Jovian Nazareth. 2021. Controlling media player with hand gestures using convolutional neural network. In *2021 IEEE Mysore Sub Section International Conference (MysuruCon)*. 79–86.

[27] Juan C. Núñez, Raúl Cabido, Juan J. Pantrigo, Antonio S. Montemayor, and José F. Vélez. 2018. Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition. *Pattern Recognition* 76 (2018), 80–94.

[28] Eshed Ohn-Bar and Mohan Manubhai Trivedi. 2014. Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations. *IEEE Transactions on Intelligent Transportation Systems* 15, 6 (2014), 2368–2377.

[29] Manuj Paliwal, Gaurav Sharma, Dina Nath, Astitwa Rathore, Himanshu Mishra, and Soumik Mondal. 2013. A dynamic hand gesture recognition system for controlling VLC media player. In *2013 International Conference on Advances in Technology and Engineering (ICATE).* 1–4. https://doi.org/10.1109/ICAdTE.2013.6524715

[30] Sang-Min Park and Young-Gab Kim. 2022. A Metaverse: Taxonomy, components, applications, and open challenges. *IEEE Access* 10 (2022), 4209–4251. https://doi.org/10.1109/ACCESS.2021.3140175

[31] Chao Peng, Jeffrey Hansberger, Vaidyanath Areyur Shanthakumar, Sarah Meacham, Victoria Blakley, and Lizhou Cao. 2018. A case study of user experience on hand-gesture video games. In *2018 IEEE Games, Entertainment, Media Conference (GEM).* 453–457. https://doi.org/10.1109/GEM.2018.8516520

[32] Johanna Pirker, Mathias Pojer, Andreas Holzinger, and Christian Gütl. 2017. Gesture-based interactions in video games with the leap motion controller. In *Human-Computer Interaction. User Interface Design, Development and Multimodality*, Masaaki Kurosu (Ed.). Springer International Publishing, Cham, 620–633.

[33] S. Rautaray and A. Agrawal. 2010. A vision based hand gesture interface for controlling VLC media player. *International Journal of Computer Applications* 10 (2010), 11–16.

[34] S. Rautaray and A. Agrawal. 2015. Vision based hand gesture recognition for human computer interaction: A survey. *Artificial Intelligence Review* 43, 1 (2015), 1–54. https://doi.org/10.1007/s10462-012-9356-9

[35] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. 2017. Hand keypoint detection in single images using multiview bootstrapping. In *Proc. of the 30th IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).* 4645–4653.

[36] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *Proc. of the 27th International Conference on Neural Information Processing Systems —Volume 1* (Montreal, Canada) *(NIPS'14).* MIT Press, Cambridge, MA, USA, 568–576.

[37] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3D convolutional networks. In *Proc. of the 2015 IEEE Int. Conf. on Computer Vision (ICCV'15).* IEEE Computer Society, USA, 4489–4497.

[38] Diana Trojaniello, Alessia Cristiano, Stela Musteata, and Alberto Sanna. 2018. Evaluating real-time hand gesture recognition for automotive applications in elderly population: Cognitive load, user experience and usability degree. In *HEALTHINFO 2018, The Third Int. Conf. on Informatics and Assistive Technologies for Health-Care, Medical Support and Wellbeing.* 36–41.

[39] Markku Turunen, Jaakko Hakulinen, Aleksi Melto, Tomi Heimonen, Tuuli Laivo, and Juho Hella. 2009. SUXES —User experience evaluation method for spoken and multimodal interaction. In *INTERSPEECH 2009, 10th Annual Conf. of the Int. Speech Communication Association.* 2567–2570.

[40] Maurice H. P. H. van Beurden, Wijnand A. Ijsselsteijn, and Yvonne A. W. de Kort. 2012. User experience of gesture based interfaces: A comparison with traditional interaction methods on pragmatic and hedonic qualities. In *Gesture and Sign Language in Human-Computer Interaction and Embodied Communication*, Eleni Efthimiou, Georgios Kouroupetroglou, and Stavroula-Evita Fotinea (Eds.). Springer, Berlin, 36–47.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* 30 (2017).

[42] Heng Wang, Dan Oneata, Jakob Verbeek, and Cordelia Schmid. 2016. A robust and efficient video representation for action recognition. In *International Journal of Computer Vision 119.* MIT Press, 219–238.

[43] Ina Wechsung, Klaus-Peter Engelbrecht, Christine Kühnel, Sebastian Möller, and Benjamin Weiss. 2012. Measuring the quality of service and quality of experience of multimodal human–machine interaction. *Journal on Multimodal User Interfaces* 6, 1 (2012), 73–85. https://doi.org/10.1007/s12193-011-0088-y

[44] Xiaodong Yang, Pavlo Molchanov, and Jan Kautz. 2018. Making convolutional networks recurrent for visual sequence learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 6469–6478.

[45] Zitong Yu, Benjia Zhou, Jun Wan, Pichao Wang, Haoyu Chen, Xin Liu, Stan Z. Li, and Guoying Zhao. 2021. Searching multi-rate and multi-modal temporal enhanced networks for gesture recognition. *IEEE Transactions on Image Processing* 30 (2021), 5626–5640.

[46] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. 2020. MediaPipe hands: On-device real-time hand tracking. In *CVPR Workshop on Computer Vision for Augmented and Virtual Reality.*

Received 5 June 2023; revised 25 November 2023; accepted 20 December 2023