Project report for the course Multisensory Interactive Systems Academic Year: 2022-2023

Training Reaction Tool

Noemi Canovi, Mattia Carolo, Laura Corso, Roberto Mazzaro
noemi.canovi@studenti.unitn.it [MAT 230487],
mattia.carolo@studenti.unitn.it, [MAT 232126]
laura.corso@studenti.unitn.it, [MAT 230485]
roberto.mazzaro@studenti.unitn.it, [MAT 229301]

Abstract:

As part of Multisensory Interactive Systems course, run by prof. Turchet in University of Trento, an interactive system involving three of the main senses (vision, touch and hearing), has been developed and tested.

It represents an opportunity for sportsmen and others to sharpen their reaction time and improve their hand-eye coordination. Indeed, it is an automatic training tool in which tennis balls fall from a platform and the user has to catch them before they touch the ground. To aid the reaction response, numerous cues are presented before the fall, such as vibrations, sounds and lights.

In addition, the timing between the start of the fall and the catch of the user is stored and sent to a website in order to provide a tool for the monitoring of the training.

This report presents the main ideas and assumptions that lead the creation of this tool, as well as the main contributions of the students that have participated in the project.

The code is stored in a GitHub (link: https://github.com/MattiaCarolo/MIS22-23)

1 Introduction

By definition, multisensory interactive systems are systems that are able to communicate with the user through multiple sensory channels (also called modes), thus acquiring and conveying information with different kinds of input and outputs. For their usefulness and adaptability this type of systems is part of our daily life. Consider, for instance, phones that communicate with the user through three modes: haptic, audition and vision. Upon typing, the user sends a haptic signal to the device, which as response modifies the text on the screen, giving a visual signal back. In addition, when a notification has to be shown to the user, this can happen both in auditory (ringtone) or haptic (vibration) form.

In our case, we have decided to build a multisensory interactive system to effectively aid the training of reaction time (RT) and hand-eye coordination. In this way, sportsman can enhance their abilities in a scenario that involves more senses, as in real competitions, but in a controlled environment. Even if not for training, other users can enjoy our system for their own fun and entertainment.

The systems consists of a fair number of integrated components which are explained in depth in the following sections. The basic idea is that through a number of sensors and actuators, two balls fall from a wooden platform and the user's purpose is trying to catch them before they are able to touch the ground.

This was inspired by a practice done by sportsmen, especially F1 pilots. In this scenario, a trainer has one tennis ball in each hand and after some time the grip is relaxed. Consequently, one or both balls start to fall and the sportsman has to catch it with its good reaction time and hand-eye coordination. Our objectives are to automatized the process, so that no other person outside the sportsman is needed, and, to give the user a tool to monitor his/her performance recording all timings.

To further refine our system, we also added some cues attached to the fall, such as lights, sounds, and vibrations. These are provided as single stimulus or combined. In both cases, they are meant to aid the user in the action of catching the balls. In fact, since both auditory and haptic system have an higher temporal resolution than visual one, they allow to a more precise and granted performance.

In addition, these cues offer the possibility to test theoretical hypothesis and perform a statistical analysis. For instance, our research question is: *"Is reaction time influenced by different modal cues?"*. In order to answer to the aforementioned query, the following hypothesis are considered:

1. The *reaction time* is **smaller** when an *auditory cue* is presented in comparison to all the other single mode stimulus. In addition, the users are able to *catch* the balls **more frequently** compared to all the other single mode stimuli. These assumptions are derived from the theoretical evidence that auditory system has the highest temporal resolution.
2. The *reaction time* is **smaller** when the fall is supported by a *haptic cue* in comparison to a visual cue, and the *number of catches*, according, is **higher** in the *haptic* case. These assumptions are derived from the theoretical evidence that haptic system has an higher temporal resolution than the visual one.
3. The *reaction time* is **smaller** and the users *catch* the balls the **highest** number of time when the fall is supported by *multimodal cues* in comparison to single modal cue. These assumptions are derived from the theoretical evidence of the superadditive effect which states that *"responses to multimodal stimuli are greater than the combined response to either modality independently."*
4. The *reaction time* is **smaller** and the users *catch* the balls the **highest** number of time when the attack (A) of an auditory cue is shorter. This assumption follow [15], where attack time was seen to be influential on RT.

To answer the theoretical hypothesis, we have asked some participants (22 in total: 12 female and 10 males) to try our system and we have collected their performances. In addition, to evaluate the usability and the experience, user have been provided with a questionnaire.

2 Related works

From the sport-related literature, it emerges that improvements in visual abilities can translate into improvements in on-field performance [7]. Indeed, many skills shape a top athlete, and among them visual skills take an important place. For this, many sportsmen attend vision programs on top of traditional training: the so-called *Sport Vision Training (SVT)*. For it, many digital tools and techniques have been implemented [6], which can be divided in *component* and *naturalistic* sports training.

Component training concentrates on the training on foundational visual skills or visual-motor reactions. In fact, these are single components that may limit the athlete on-field performance if not properly developed. Among them, popular is the use of setups in which sportsman have to press, as quickly as possible, buttons that randomly lit up, such as FitLight [8].

In naturalistic training, athletes practice their sports with an additional element that alter or augment the training experience, for instance, stroboscopic visual training uses eyewear to interrupt normal visual input. These alterations are proved to enhance the sensory systems and will result in a better athletic performance.

In addition, numerous studies have demonstrated that learning is improved when the information is presented in multiple sensory modalities, leading to better encoding and retrieval of perceptual information [13] and to better generalization [14]. To this end, multisensory systems have been developed for aiding efficient learning: many are tools for helping students' education in schools [10] [11], common are dental training systems with multisensory feedback for surgical operations [12], others are methods for sport training. In particular, Ultimeyes [9] is a digital application that uses a variety of stimuli and multimodal feedback to aid vision training.

Nonetheless, from the best of our knowledge, our application is the first to incorporate multisensory stimuli and feedback into a complex system that is not solely digital, for the purpose of SVT.

On another note, in the scientific community numerous studies have been trying to understand how reaction time works with different stimuli and tasks. Most, however, focus in simple visual search [1] [2], or address continuous conditions, such as a music constantly playing during the task [3]. Here, we address a more complex scenario with a compound task involving both reaction time and hand-eye coordination and thus should be leading to more interesting results.

3 Architecture design

Our system is well summarized in Figure 2. The core of the whole system is the *Raspberry Pi*, a Single-Board Computer (SBC) with Linux as Operative System, running a web service which will have four main tasks:

- Communicate with the *Teensy* (for more details see Section 4.1);
- Control *Pure Data* (for more details see Section 4.2);
- Run an user interface made with *HTML*, *CSS* and *Processing* (for more details see Sections 4.3, 4.4);
- Save the results of the tests on a *CSV* dataset.

The backend of the web service exploits *Flask* as a web framework, *Celery* to run asynchronous tasks in parallel, and *Redis* as a broker to manage information between the original application code and the evoked task. A web server, able to tackle asynchronous tasks, enables us to follow calls directed at our electronic device during their entire lifespan (for more details see Section 4.5).

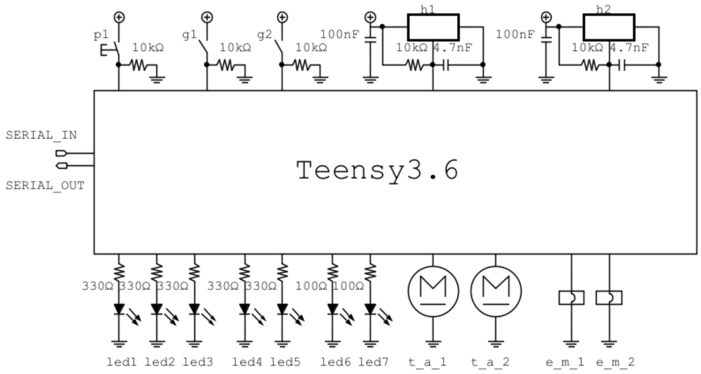

Figure 1: Architecture Physical Linkage

The *Raspberry Pi* is connected with Wi-Fi to a LAN made available by a router. Any device connected to this network can access the User Interface by linking to the IP of the *Raspberry Pi*.

Connected to the *Raspberry Pi* with USB link there is the microcontroller *Teensy*. It is a crucial component of the system and it handles low-level control of the electronic board where all the sensors and actuators are connected. These are required to execute a trial, which is defined as follows:

1. The participant and the balls get in position;
2. After a random delay, a cue is presented to the participant;
3. After 400 msec one of the two balls gets released;
4. The participant tries to catch the balls before they reach the ground;
5. Consistently with the success or failure of the participant, a visual-audio feedback is presented;
6. Performance results are saved in a *CSV* database.

In practice, the Teensy is responsible for controlling sensors and actuators to execute a flawless trial. To accomplish this, the following components are connected to the electronic board:

- Two **electromagnets** are utilized to secure the tennis balls in place. The balls have a small cupola-shaped piece of metal attached to them, enabling the electromagnets to hold them.
- Two **US5881 hall sensors** are employed to detect if the user's hands are in the correct positions. Special gloves are worn by the user, featuring a magnet with the south pole oriented upward, allowing the hall sensors to detect its proximity. The hall sensors are connected in their recommended three-wire conditioning circuit, as suggested in the datasheet.

- Two **signaling LEDs** illuminate when the hands are correctly positioned.
- Three **signaling LEDs** serve as output indicators, indicating the system's operational state during the test. These LEDs are turned off when the user is waiting for stimuli to prevent distractions.
- Two **high-brightness LEDs** produce visible visual stimuli. High-brightness LEDs are utilized to ensure clear visibility for the user.
- Two **vibration motors** are embedded in the gloves to generate haptic stimuli. These are located specifically on the top of the index finger because this area is one of the most sensible to haptic stimulation of our body.
- The system includes two **gloves**. The outside of the glove palm is outfitted with a sewn metal mesh, divided into two parts: one covering the right half and the other covering the left half. This arrangement allows the circuit to be completed when the user grasps the aluminum-coated tennis ball. Alternatively, the circuit can also be closed by having two fingers covered by different sections of the metal mesh touch each other. Thus, it is assumed that the user follows the instructions and properly uses these gloves. These special gloves, which allow the user to detect balls covered with aluminum foil, enable them to be actively involved in the experiment through tangible interaction.

The communication between the *Teensy* and the *Raspberry Pi* occurs through the serial port. The key exchanged messages are: the initialization one and the one containing the trial's results (for more details see Section 4.6).

Auditory stimuli are produced by *Pure Data*, which efficiently controls the **Behringer UMC202HD soundcard** connected with USB link to the *Raspberry Pi*. Two **JBL loudspeakers** are connected to the soundcard, one for each output channel, providing stereo audio by placing one speaker on the left side and the other on the right side of the user. They are placed laterally to provide a spatial congruency across all the unimodal stimuli that are presented to the user and to provide an auditory cue that can be easily localized. In addition, loudspeaker are chosen instead of headphones to not hinder the user's movements with additional cables. The server controls *Pure Data* using the *OSC* (Open Sound Control) protocol, which is optimized for real-time communication between computers and sound synthesizers. This protocol ensures the production of sound stimuli without any noticeable delay.

The entire system is mounted on a **wooden plank** supported by two iron stands, which are securely bolted to the ends. This setup ensures stability and a fixed position for the system during operation. The iron stands are adjustable in height, allowing for the height of the wooden support to be customized according to the user's requirements. In the middle of the board, at the back, there is a wooden shelf on which all the hardware components of the system are placed, except for the electromagnets, which are mounted on the wooden plank at 20 centimeters from the center, and the hall sensors, which are positioned about ten centimeters away from the electromagnets towards the outer edges.

3.1 Usage model

Welcome to the *Training Reaction Tool* user manual!

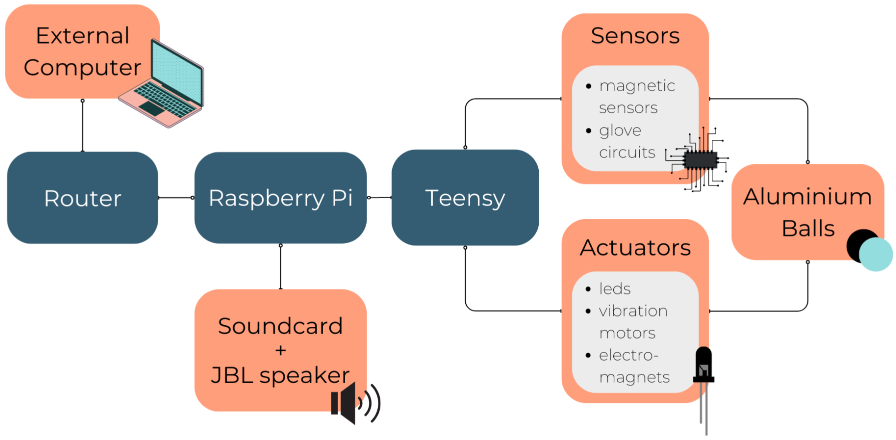

Figure 2: Architecture Physical Linkage

3.1.1 Before using the system

1. Plugin the system to a power source. If correctly linked you can see the *red* and the *yellow* LED switched on.
2. Login in the website or create an account if you are a new user.
3. You can try the stimulus (audio, visual, haptic) presented by the appliance from the web interface.

3.1.2 Use the system

You can try the different trials by using the dropdown menu in the *Single Trial Session*. To start a trial:

1. Press the *Start* button.
2. Wear gloves carefully in order to not damage the circuits sewed on the palm.
3. Attach the two *aluminium balls* under the magnets.
4. Align the yellow pockets sewed on the top of the gloves to the black tabs.
5. Wait for a stimulus.
6. After a random time, at the presentation of the stimulus, try to catch the ball at the side to which the cue is presented. For instance, if the right white LED turns on, the right ball will fall and you have to catch it as fast as possible with your right hand.
7. A visual-audio feedback is played according to your performance: if you were able to catch the ball you will be presented with an happy feedback, otherwise a sad feedback is produced.

You can also use the system to monitor your reaction to every stimulus with the *Test Session* page of the website. In this modality, follow the points aforementioned repeating all of them until a banner appears on the screen warning you that the test is over (in total there will be eight trials).

3.1.3 Statistics

In the *Charts* web page, your progress are displayed as graphs to allow you a continuous monitoring of your performance.

4 Implementation

Our system is backed by a software composed of different parts. Each part is described in detail in the following subsections.

4.1 Teensy

As already introduced in Section 3, the *Teensy* is responsible for controlling sensors and actuators in the proper manner required to execute a trial of the test. Since a trial can be divided into multiple sequential steps, the code structure that best fits this idea is called a *State Machine*. It consists of a set of states and transitions between those states. The current state represents the system's current condition, and transitions define how the system moves from one state to another based on certain events or conditions. The *Teensy* can be programmed with *Arduino* language which is C++ based. The structure of an Arduino sketch involves an initial phase of constants and variables initialization. Then, in the *void setup* function, the roles of the utilized PINs and interrupts are initialized; contrary to *Arduino* here is not necessary to initialize the serial communication since it is always set to maximum USB speed for the *Teensy*.

These initialization steps are executed only once when the *Teensy* is turned on, after which the *void loop* begins. Inside the *void loop* the serial port is read to search for communications coming from the Server running on the *Raspberry Pi*, and then there is the *State Machine* which presents the following states:

0. **Case(0):** Wait for test initialization. This means that the *Teensy* is waiting to receive instructions on which stimuli to activate in this trial. During this phase, the electromagnets are activated to keep the balls in position, and the yellow and red LEDs are turned on to indicate that the *Teensy* is operational and waiting to be initialized. When a valid stimuli initialization is received, the green LED is turned on, and a transition from state 0 to state 1 is occurs.
1. **Case(1):** Wait to read a low state from both hall sensors. This indicates that the user is correctly positioned to start the test. If the condition is verified, all the LEDs are turned off to prevent user distraction, a random timer is initialized, a random number between 0 and 1 is generated to determine whether to release the right or left ball and finally state advance to state 2. The user can see if his hands are correctly positioned thanks to two LEDs which are turned on when the hand is correctly positioned.
2. **Case(2):** The user has to keep the hands in position for all the duration on the randomly initialized timer. Once this condition is verified cues are reproduced, current timestamp is

saved and considered as the start time of the test. Then state 3 begins. However, if the hands are misplaced, state 8 is called. State 8 serves as a small state to reset the timer, and then we return to state 1.

3. **Case(3):** This state waits a fixed time, 400msec, before releasing one of the balls turning off one electromagnet.
4. **Case(4):** The stimuli are turned off when the user catches the ball or when the maximum stimuli duration is elapsed. In the first case, the current timestamp is saved. Then, state 5 begins.
5. **Case(5):** A message is sent to the Server, asking to reproduce a feedback coherent with the user performance (success or unsuccess). Indeed, the user has a maximum time available to catch the ball before the trial is considered unsuccessful. Furthermore in case of success the program advance to state 6, otherwise it ends to the reset state.
6. **Case(6):** Here we communicate to the Server how much the user took to catch the ball. Then we move to reset state.
7. **Case(7):** This state can only be invoked from the Server when the *Teensy* receives the proper message. It is used to test the visual and tactile stimuli. Note: there is no need to control auditory stimuli since it can be done directly from the Server by sending a communication to *Pure Data*.
8. **Reset State:** This is a simple state used to reset the timers and re-initialize all the variables to their initial values.

Since we have to acquire reaction times, the system must be very efficient. For this reason, we cannot use the *delay()* function inside the code. Instead, everything has been done with the *millis()* function, which exploits internal *Teensy* timers to work asynchronously. Additionally, the catching of the balls is implemented with *interrupts* since we wanted to stop the timer as soon as possible. It is important to remember that all the variables used inside an ISR (Interrupt Service Routine) must be declared as volatile.

4.2 Pure Data

Pure Data [4], a visual programming language for creating music, was exploited to create auditory cues and feedback.

In particular, the two cues consists in two pure tones at 440 Hz and total duration of 200 msec. The main difference between the two is in the envelope, which is one of the main characteristics of sounds. In practice, it can be considered as a curve outlining the amplitude of the auditory signal. ADSR model is often used to describe the envelope and stands for attack (A), decay (D), sustain (S) and release (R) phases. In [15], attack time was seen to be influential on RT. Following this, we tested two attack times: 10 msec and 50 msec.

In addition, we wanted to provide positive and negative feedback to the user both in visual and auditory form. In case of success, the system plays a triumphant melody consisting of six periodic complex high tones and a bell sound in rapid succession. Otherwise, a slower four periodic complex low tones melody was played to match the user's performance.

For all these, a single Pure Data patch was created with 6 different sub-patches. These are are selected when needed, and reproduce: unsuccessful melody, successful melody, cue with attack time equal to 10 msec only on the left loudspeaker, cue with attack time equal to 50 msec only on the left loudspeaker, cue with attack time equal to 10 msec only on the right loudspeaker and cue with attack time equal to 50 msec only on the right loudspeaker.

4.3 User Interface

In our system an User Interface (UI) has been developed. This allows the user to start a trial or a test, which is composed by many trials, and have an instant feedback of her/his performance. In addition, the user interface permits the user to record and monitor her/his progress with the aim of training use through the creation of a personal account.

The User Interface is developed with the flow showed in **Fig. 3**. It starts with a *Login* page that asks to an already registered user his/her credentials. If a new person wants to use the tool a link to the *Sign Up* page is provided.

After the access, the user lands on the *Cue Test* page in which three buttons allow to experience what are the stimuli later presented (audio, visual, haptic). In addition, in the bottom right corner, an arrow allows the participant to proceed in the flow toward the *Single Trial Session*. In this page, a dropdown menu can be used to select an example trial, i.e. the haptic/audio/visual/combined stimulus followed by the release of the balls. Again, an arrow on the bottom left corner allows the user to proceed in the *Test Session*, where eight trials will be presented in succession to complete a test.

After the test termination, the user is redirected to the *Questionnaire* page, through the bottom-left arrow, where impressions on the utilized system are collected.

In every aforementioned page, except for *Sign Up* and *Login*, a sandwich menu allows to navigate through all the interfaces without following the depicted flow. In addition, two more items can be found in the menu: the *Log Out* command that redirects the user to the *Login* page and the *Charts* page. The latter contains the average reaction time, the average percentage of catch, the line chart illustrating the reaction time per trial and the bar chart for the catchings.

The composition and the colors of the UI are chosen referring to the *Gestalt principles* to ensure clear legibility and natural interaction.

4.4 Processing

Processing [15], a flexible software sketchbook that allows to draw two and three-dimensional graphics, was exploited in this project to develop an animation in the aforementioned *Single Trial Session* and *Test Session* pages of the UI.

The graphic is composed of: a timer, which starts when the balls are released by the electromagnets, a "Good Job!" animation and a "Try Again" animation. These two are played along the auditory positive and negative feedbacks to provide a visual confirmation of the trial's outcome. In each, the colors and the movements of the elements were chosen in order to reflect the meaning of the information to communicate. Therefore, the "Good Job!" message is printed in green and is anticipated by fast, bright colored circles appearing on the white background along

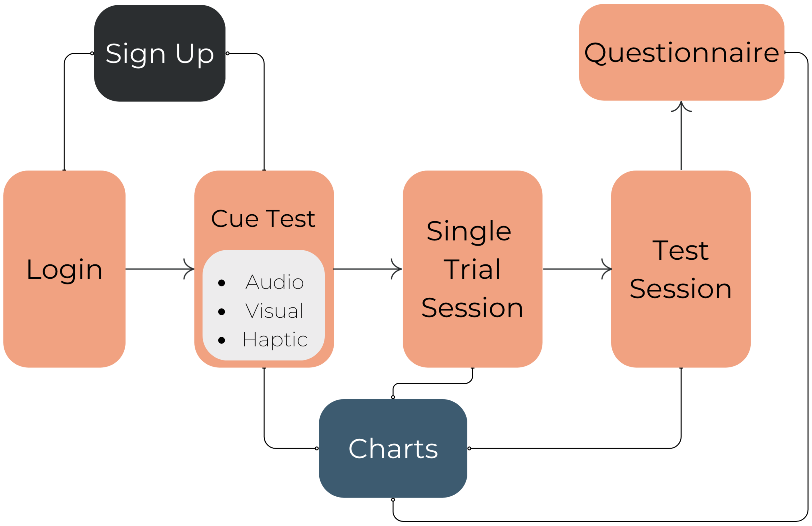

Figure 3: User Interface Flow

with the reaction time. Instead, the "Try Again" message is printed in red and it is anticipated by circles colored in blue hues that fall top to bottom to mimicking the rain.

4.5 Server

4.5.1 Frontend

The frontend of our project consists in the User Interface (see Section 4.3) paired with some functions coded in *JavaScript* and *AJAX*. As already mentioned, enabling asynchronous tasks is vital to our project so even for the frontend part we needed to develop methods that could work with that. For every trial, an *AJAX* call is made to the Server where it waits for the initial response. Depending on the response, the hosting page will return a feedback animation depending on the result received. Further explanation on the overall process can be seen in Section 4.6.

4.5.2 Backend

The core component of the backend is *Flask*, which is a micro web framework optimal for our project since it does not bring any unnecessary feature to the codebase, which usually tend to slow our communication process. Paired with it, we added the *Celery* framework because we needed a tool able to generate separate tasks that work in an asynchronous way with respect to the Server. Still, tasks are capable of being tracked and serve data to the client during their lifetime. Moreover, the possibility of tracking a process during its overall lifespan, separately from the main application code, is exploited to follow every state of a trial, so that the response to the client can always be synced with the *Teensy* counterpart. Those two frameworks are the backbone of our network. The *Flask* framework enables us the routing of all data across the system, working as a kind of central hub. All results and requests pass through here and than

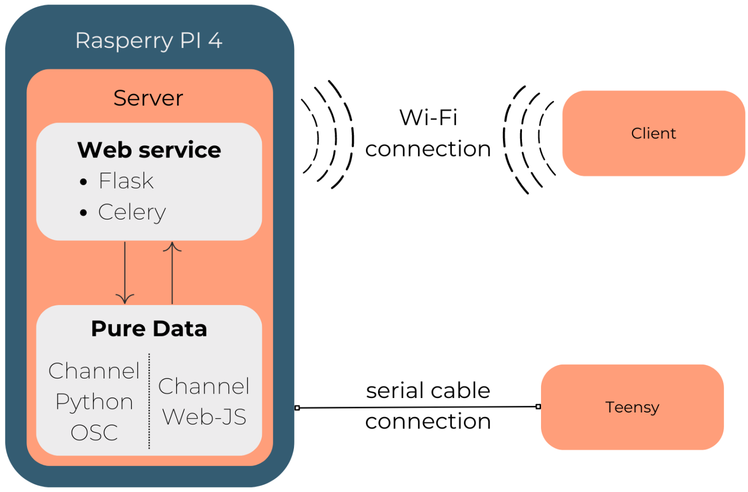

Figure 4: Networking

will be digested and redirected to the appropriate *Celery* worker on the necessary cases.

4.6 Networking

As can be seen in **Fig. 4**, the network inside the system can be separated in three macro areas that corresponds to the different communication channels.

The first channel is the *Teensy-Server*, which is responsible of updating the Server on the state of the trial and communicating the results when a trial has ended. Here, is the Server that establishes a connection by issuing a command. This command indicates the execution of a stimuli or the start of a trial. In the first case, the *Teensy* controls the actuators to produce the stimuli, and no additional feedback is required. In practice, there is no need to track the task and the response will be given immediately. The second case is a bit more complex as the Server needs to be in synchronization with the *Teensy* during all the different states of a trial. To achieve so, the network handles jitter episodes, i.e. errors coming from the board and missing messages, while ensuring the key messages are prioritized. At first, the Server parses the command received from the client in order to be redirected to the *Teensy*. After the parse, the host will try to establish a connection to the board by sending the command desired. When the command is received by the *Teensy* it will be parsed and will send back to the Server the different status of the trial. Basing on the different state of the *Teensy*, the Server updates locally an internal state that will be show by the status code of the task. The main status codes are :

1. **PENDING** which signifies the two balls are attached to the structure
2. **PROGRESS** which signs the balls have been released, so the animation timer shown on the frontend starts
3. **SUCCESS/FAILURE** which tells the result of the trial

The second one is the *Client-Server* channel, which takes into care the tracking of the overall status of a trial through the aforementioned status codes. At the beginning of the communication the client starts a new trial which corresponds to an *AJAX* call. Due to the nature of the HTTP protocol as soon as a request is sent a response must be given by the Server. However in a normal situation to send a response the Server must wait for a result from the *Teensy* but the client will be timing out long before it. To face this problem, a dynamic URL address is given to the client. With it, the client can send request to the newly created URL on the status of the trial which will send as a response a *JSON* with all the information that can be retrieved on the trial process.

The last channel is *Pure Data-System*. This part is exclusively made for the audio interactions within the Server. Here, *Pure Data* will open two ports for routing messages inside the patch. One is directed towards the client and one towards the Server but both share the the same communication protocol.

5 Evaluation

The research question of our project is: *"Is reaction time influenced by different modal cues?"*. To answer this and the hypothesis (See section 1), an experiment procedure was designed consisting on a defined number of trials.

The definition of trial is given in Section 3.

The experiment procedure is as follows:

1. **Login session**, in which the user creates an account and access the website with her/his credentials.
2. **Practice session**, in which the user get accustomed with the system:
   (a) Three unimodal cues are presented to the user;
   (b) User tries the system with the trimodal cue.
3. **Trials session**, in which we collect the user performance:
   (a) The user executes eight customized trials, which are summarize in Table I, in random order and without repetitions.

The eight customized trials differ only on the cues presented. In this sense, our control and independent variables are the stimuli. From the experiments we wanted to collect and analyze two dependant variables: catch of the ball and reaction time.

The former is a discrete variable with three values:

- 1: the user was able to properly catch the ball during the fall;
- 0.5: the user was not able to catch the ball during the fall, but her/him was close (for instance, the user touched the ball but was not able to grasp it);
- 0: the user did not catch nor touch the ball during the fall.

CUES			
	VISUAL	AUDITORY	HAPTIC
UNIMODAL	X	-	-
	-	X	-
	-	X*	-
	-	-	X
MULTIMODAL	X	X	-
	-	X	X
	X	-	X
	X	X	X

Table I: Different cues attached to the fall of the balls. "X" means the cue is present, "-" means it is not. Regarding auditory cues, "X" refers to cue with attack time equal to 10 msec, "X*" refers to cue with attack time equal to 50 msec

Reaction time is a continuous variable indicating the time between the start of the fall and the catch of the user. This is set to 0 if the user was not able to catch or touch the ball during the fall.

In addition, our experiment falls in within subjects design: all participants performed all conditions, i.e. each user executed the eight customized trials. Furthermore these trials are performed in random order to avoid order effect.

Participants were recruited by asking friends and acquaintances, organizing sessions with multiple people. Nonetheless, each person did the experiment alone, without envisioning other participants' trials. This was done to ensure every participants to have equal knowledge and experience with the system. In total 22 participants were recruited: 10 males and 12 females, of medium age of 25.19.

Regarding reaction time we performed Analysis of Variance (ANOVA), which is a statistical method to estimate if two or more factors influence a dependant continuous variable in a statistically different way from each other. In practice, we compare our 8 forms of stimuli as single factors to see whatever reaction time changes in a meaningful way. Considering both the successful trials alone (with catch equal to 1), and the successful plus the almost successful ones (with catch equal to 0.5 and 1), no statistically difference can be seen between the different

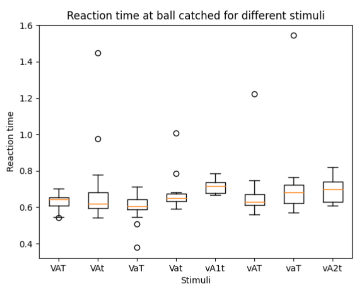

Figure 5: Boxplot of reaction time with catched ball

STIMULUS	NUMBER OF CATCHES	PERCENTAGE (%)
Vat	14	63.64
vAt	6	27.27
vA*t	12	54.55
vaT	13	59.09
VAt	15	68.18
vAT	18	81.82
VaT	16	72.73
VAT	20	90.91

Table II: Percentage of catch. V = visual, A = audio, T = tactile. When uppercase the modality is present, if lowercase not. A = audio with attack time equal to 10 msec, A* = audio with attack time equal to 50 msec

stimuli (p = 0.248 and p = 0.289 respectively). This means in our experiment reaction time was not influenced by different modal cues. This can also be seen in the corresponding boxplots (**Fig. 5**).

In contrast to this, catch percentage is visibly affected by the stimuli. In particular, from Table II we can notice how the trimodal cue performed the best (90.91%), confirming our hypothesis on superadditive effect. However, the other stimuli did not performed as expected: regarding unimodal cues, auditory and visual ones gives the worst (27.27-54.55%) and best (63.64%) performance respectively, opposite to what anticipated. In addition, the auditory cue with shorter attack time did not acquire better results compared to its counterpart. Out of four, we rejects three of our hypothesis.

In order to collect subjective responses from the users, a questionnaire was compiled after the experiments. The questionnaire is composed of six sections covering personal information, opinion on the stimuli, Self-assessment manikin, SUS questionnaire and opne questions.

Overall, the cue analysis shows a deep contrast between participants impressions and actual performance. In particular, participant thought the unimodal stimulus that aided the most was the haptic one (42.9%), followed by auditory (33.3%) and visual ones (23.8%). In reality, the one that achieve best catch perfomance is the visual cue. Among multimodal cues, the ones participants thought more helpful were the cues involving haptic stimuli (47.6 and 42.9%), and auditory plus visual receive few votes (9.5%). However, no participant chose the trimodal cue, which achieved the best catch result.

The Self-assessment manikin shows the majority of people experienced a pleasant interaction with the system (7.23/9 valence), a bit of excitement (5.90/9 arousal) and a medium level of control (5.10/9 dominance). This corresponds with the open answers, where participants claimed to have had fun and a positive experience, but at the same time they indicated also difficulty when aligning the hands to the wooden plank, lacking a sense of control.

SUS analysis gave the system an average usability scale score of 80.23 out of 100, which is a score above average.

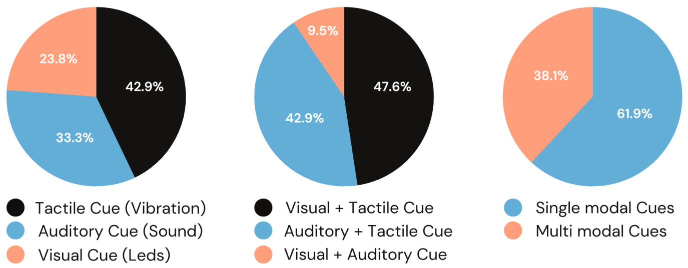

Figure 6: Questionnaire answers to the question: "which cue do you think aided the most?". The answer was divided in three: only unimodal cues, only multimodal cues and unimodal vs multimodal

6 Discussion and conclusions

Three out of four of our initial hypothesis have been rejected. Below, some considerations about them can be found. **HP 1**: the auditory cue has not resulted as the best single modal stimulus, maybe because the loudspeakers' stereo audio creates ambiguity. Headphones could supply better localization results. **HP 2**: the visual stimulus results to be the most useful in catching the balls because one can focus at the white LEDs, while looking at the balls through peripheral vision. Peripheral vision is able to perceive small movements detail, aiding the catch. **HP 3**: multimodal cues achieves better performance in comparison to unimodal ones thanks to superadditive effect, as expected. **HP 4**: the auditory cue with shorter attack time did not perform as expected. This can be caused by the order effect: most of the time, this cue was presented as first or second trial (mean position: 3.5 over 8), where the user still had to learn how to efficiently react to the cues. In contrast, the auditory cue with longer attack was more spread (mean position: 4.9 over 8) and was presented only 6 times out of 22 before the other. Therefore, it's possible that the users were more accustomed to react to the cue with longer attack than the shorter one.

Some limitations emerged while developing and evaluating the system.

Overall, our system is a prototype built with good but not professional hardware components. This results in some occasional failure. One of the main issues while implementing the system was how to connect the *Teensy* to the *Raspberry Pi* in an efficient way. For this, a specific protocol was developed. In addition, hardware components are sensible so both their capabilities and common problems have to be taken into account. For instance, sometimes an error occurs where the board timer returns a not realistic number, such as *0.00* or milliards. These are not predictable, but fortunately happen rarely. Regarding the hardware failures, the other critical part are the gloves, which are homemade thus very fragile.

During the experiments, our users reported by voice and in the questionnaire that the most difficult part of interacting with the system was placing the hands in the initial position. Therefore, an obvious solution can be placing bigger magnets in the gloves to help the hall sensors with the detection of the magnetic field.

Sometimes, between the catch of the ball and the audio-visual feedback there is a bit of delay. Nevertheless, this was not perceived as an issue by the participants: they were vocal about the satisfaction to hear and see a successful feedback, and a bit disappointed when the unsuccessful feedback was played.

In the future, it could be interesting to test our system with a bigger group of people, to have more reliable and statistically significant results. Furthermore, trying other stimuli could offer other perspectives and change the results of the different modalities.

7 Group members contributions

The main contributions of each member are:

- Noemi: Literature Review, PureData, Frontend, Questionnaire, Video
- Mattia: Backend, Networking
- Laura: Processing, Frontend, Video, Statistics
- Roberto: Arduino code and hardware (plank, glove, balls, board and schematics)

Note that, even if each of the members concentrated on a different part of the project, we were rather dynamic and collaborative in searching for possible improvements and issues in the whole system. Additionally, we met often in order to ensure every member was up to date with the project implementation process.

References

[1] M. K. Ngo, C. Spence. "Auditory, tactile, and multisensory cues facilitate search for dynamic visual stimuli". *Attention, Perception, & Psychophysics 72.6: 1654-1665*. 2010.

[2] E. R. Hoeg, L. J. Gerry, L. T. Niels, C. Nilsson, S. Serafin, "Binaural sound reduces reaction time in a virtual reality search task". *IEEE 3rd VR workshop on sonic interactions for virtual environments (SIVE)*. 2017.

[3] M. L. Turner, J. E. Fernandez, K. Nelson. "The effect of music amplitude on the reaction to unexpected visual events". *The Journal of General Psychology 123.1: 51-62*. 1996.

[4] PureData *puredata.info* [Accessed Apr. 14 2023]

[5] Arduino *www.arduino.cc* [Accessed Apr. 14 2023]

[6] L. G. Appelbaum, G. Erickson. "Sports vision training: A review of the state-of-the-art in digital training techniques". *International Review of Sport and Exercise Psychology*. 2016.

[7] L. G. Erickson. "Sports vision: Vision care for the enhancement of sports performance". *Butterworth-Heinemann*. 2007.

[8] Fitlight Trainer *fitlighttraining.com* [Accessed Apr. 14 2023]

[9] Ultimeyes *ultimeyesvision.com* [Accessed Apr. 14 2023]

[10] M. Kast, M. Meyer, C. Vögeli, M. Gross, L. Jäncke. "Computer-based multisensory learning in children with developmental dyslexia". *Restorative Neurology and Neuroscience 25.3-4: 355-369*. 2007

[11] N. S. B. M. Maliki. "Application of Multisensory in Learning Alphabets Identification Skills for Special Education Students" *Journal of ICSAR 1.2: 150-154*. 2017

[12] D. Wang, T. Li, Y. Zhang, J. Hou. "Survey on multisensory feedback virtual reality dental training systems". *European Journal of Dental Education 20.4: 248-260.*2016.

[13] L. Shams, A. R. Seitz. "Benefits of multisensory learning". *Trends in cognitive sciences 12.11: 411-417*. 2008.

[14] L.-Q. Xiao, J.-Y. Zhang, R. Wang, S. A. Klein, D. M. Levi, C. Yu. "Complete transfer of perceptual learning across retinal locations enabled by double training". *Current biology 18.24: 1922-1926*. 2008.

[15] J. Schlittenlacher, W. Ellermeier. "Simple reaction time to the onset of time-varying sounds". *Attention, Perception, & Psychophysics, 77, 2424-2437*. 2015.

[15] Processing *processing.org* [Accessed May. 22 2023]

[16] Flask *https://flask.palletsprojects.com/en/1.1.x/* [Accessed May. 22 2023]

[17] Celery *https://docs.celeryq.dev/en/stable/* [Accessed May. 22 2023]