NUFFIELD DEPARTMENT OF
CLINICAL NEUROSCIENCES
Medical Sciences Division

The Reaching Task Game

Prepared by

Deniz Kucukahmetler

Supervisor

Hayriye Cagnan

Abstract

In the University of Oxford, the Nuffield Department of Neurosciences department has several groups working with tremor patients. One of the groups is the Cagnan Group, which is led by Hayriye Cagnan, my supervisor. The project I was assigned and did for two months was called "The Reaching Task Game". This game is designed for tremor patients who mostly have Parkinson's disease or essential tremor. It is implemented to investigate if the decision making and/or tremor is influenced by the sensory cues. It is decided to do home testing instead because of the pandemic situation so the implementation plan was done accordingly. Unity and Leap motion are used for development. The main idea behind the game is after seeing a sensory (visual) cue on the screen for a while, the user decides a location and pops the balloon in that location. If the decision was correct, the balloon pops. Task design was done as a 2x2 matrix using the combination of two parameters: precision and uncertainty. Four different completed versions of the Reaching Task Game were developed. In each of them, precision was determined with the balloon size whereas uncertainty was determined by different objects in the versions.

Contents

1 Introduction

During the internship I was assigned one of the Project in the Cagnan Group, the "Reaching Task Game" which aims to investigate sensory cue effect on tremor patients in terms of tremor and the response time in decision making. In the light of this idea, a game was designed called as "Reaching Task Game". It was designed with different conditions assigned to different uncertainty and precision levels. Each condition had a unique sensory cue on the screen that will make the patient decide where the balloon appears in the next scene and point to the decided balloon accordingly. However, this game was made for the lab environment and had several weak points. I was assigned to this project to make improvements and solve the bugs in the game as the first step. As the second step, I was expected to modify the game suitable for at-home testing (instead of lab testing). Finally, I was expected to do a pilot of the data obtained. However, time didn't permit to conduct the final step.

2 Project Background

2.1 Department Information

Department in the institution my project ie. Reaching Task Game Project was assigned to the Nuffield Department of Clinical Neurosciences in the Medical Sciences division. The unit of the project assigned in this department was The Medical Research Council Brain Network Dynamics Unit. I joined to Cagnan Group which was led by Hayriye Cagnan. I worked mostly with Hayriye Cagnan, Tim West and Carolina Reis throughout my internship.

2.2 Motivation and/or problem definition

The main aim of the project was to design a Unity 2D game using the Leap Motion device (hand detection device, shown in Figure 2.1) and software for tremor patients. The purpose of this project is to investigate sensory cue impact on decision making (in terms of the response time passes until the decision is made) while analyzing the tremor change on tremor patients in at-home testing conditions. The target user profile for the game is more than 65 years old on average. So, the game had to be easy to open and play, easy to understand where to put the Leap Motion camera, the instructions had to be simple and easy to follow since no one would be there to give directions to the users (tremor patients). Besides, being suitable for home testing brings along the need for accurate working in different operating systems.

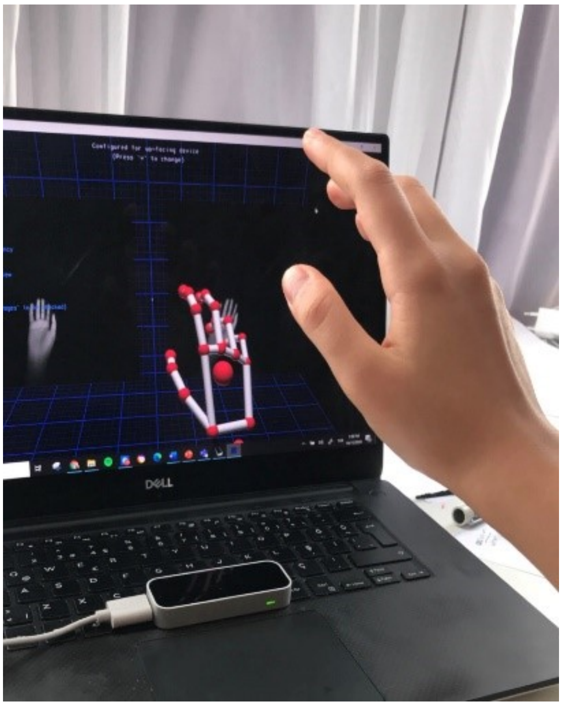

Figure 2.1: Leap Motion is on the space tab. User's hand is detected by the Leap Motion and shown as a virtual hand on the screen.

2.3 Related literature

It is important to understand Parkinson's disease and essential tremor before designing the game since they are brain diseases i.e. not causing only physical tremors but has emotional and mental outcomes. The reason why it is important to understand is, the game should not cause a bad effect mentally. For instance, seeing their hand feedback on the screen. So, some properties of the game may not have an impact on a healthy person while it may interact and cause a bad effect on the patient's brain. Also, it is necessary to know the differentiation of Parkinson's disease and essential tremor to analyze the data better. There have been experiments for both understanding tremor differentiation and their brain differences using several methods as deep brain stimulation, neuroimaging (MEG), response time for different conditions analysis). Detailed source research was done for this purpose. Since the Reaching Task Game Project is interested in the tremor patient's decision making under different sensory cues, sources found related to this subject. As expected, I started reading the papers my supervisor (Hayriye Cagnan) published. Then, as the game evolved withing itself for two months, other papers were read. Cagnan Group team member Tim West made a paper list related to the subject. Main papers from that list which helps to understand the project background better is as follows:

- Loh, M. N., Kirsch, L., Rothwell, J. C., Lemon, R. N., Davare, M. (2010). Hömberg, V., Hefter, H., Reiners, K., Freund, H. J. (1987). Differential effects of changes in mechanical limb properties on physiological and pathological tremor

- Elble, R. J. (1986). Physiologic and essential tremor.

- Lee, R. G., Stein, R. B. (1981). Resetting of tremor by mechanical perturbations: A comparison of essential tremor and parkinsonian tremor.

- Avanzino, L., Bove, M., Tacchino, A., Ruggeri, P., Giannini, A., Trompetto, C., Abbruzzese, G. (2009). Cerebellar involvement in timing accuracy of rhythmic finger movements in essential tremor.

- Bareš, M., Lungu, O. V., Husárová, I., Gescheidt, T. (2010). Predictive Motor Timing Performance Dissociates Between Early Diseases of the Cerebellum and Parkinson's Disease.

- Chen H, Hua SE, Smith MA, Lenz FA, Shadmehr R. Effects of Human Cerebellar Thalamus Disruption on Adaptive Control of Reaching. Cereb Cortex. 2006;16: 1462–1473. doi:10.1093/cercor/bhj087

- Perugini, A., Ditterich, J., Basso, M. A. (2016). Patients with Parkinson's disease show impaired use of priors in conditions of sensory uncertainty. Current Biology, 26(14), 1902-1910.

- Wolpert, D. M., Landy, M. S. (2012). Motor control is decision-making. Current Opinion in Neurobiology, 22(6), 996–1003.

- Gallivan, J. P., . . . ,Flanagan, J. R. (2018). Decision-making in sensorimotor control. Nature Reviews Neuroscience, 19(9), 519–534.

- Krüger, M., Hermsdörfer, J. (2019). Target Uncertainty During Motor Decision-Making: The Time Course of Movement Variability Reveals the Effect of Different Sources of Uncertainty on the Control of Reaching Movements. Frontiers in Psychology, 10, 41.

- Wong AL, Haith AM. Motor planning flexibly optimizes performance under uncertainty about task goals. Nat Commun. 2017;8: 14624. doi:10.1038/ncomms14624

- Gallivan, J. P., Chapman, C. S. (2014). Three-dimensional reach trajectories as a probe of real-time decision-making between multiple competing targets. Frontiers in neuroscience, 8, 215.

- Tzagarakis, C., Ince, N. F., Leuthold, A. C., Pellizzer, G. (2010). Beta-band activity during motor planning reflects response uncertainty. Journal of Neuroscience, 30(34), 11270-11277.

- Gallivan JP, Logan L, Wolpert DM, Flanagan JR. Parallel specification of competing sensorimotor control policies for alternative action options. Nat Neurosci. 2016;19: 320–326. doi:10.1038/nn.4214

- Stevenson, J. K. R., Oishi, M. M. K., Farajian, S., Cretu, E., Ty, E., McKeown, M. J. (2011). Response to sensory uncertainty in Parkinson's disease: a marker of cerebellar dysfunction? European Journal of Neuroscience, 33(2), 298–305.

- Tzagarakis C, Ince NF, Leuthold AC, Pellizzer G. Beta-Band Activity during Motor Planning Reflects Response Uncertainty. J Neurosci. 2010;30: 11270 LP – 11277. doi:10.1523/JNEUROSCI.6026-09.2010

3 Implementation and Details

3.1 Project Objective

There are previous psychology studies that sensory cues in the environment have an impact on decision making and the response time to the cue. For tremor patients, it needs to be investigated considering their neurological and physiological differences. How do the different levels of uncertainty and precision in the sensory cues have an impact on the tremor and decision-making time? For this purpose, Cagnan Group designed a game and implemented it in Unity 2D platform and called it the Reaching Task Game. This game only had one version using the arrows as the uncertainty parameter. Hand detections were meant to be detected by the Leap Motion engine to understand how different conditions in the game affects the tremor and response time for decision making. After the pandemic emerged, it got harder to do neuroscience experiments on patients in university labs. So that the Reaching Task Game thought to be sent to the patients' homes. This brings along the fact that in this way, there is no need to cause physical contact of patient and the researcher. Therefore, especially in those times, it is convenient to do home testing. But with the home testing, there are some problems and concerns listed:

1. Patients are mostly over 65 years old. So, the instructions of the game should be direct and easy to understand

2. There is no one to check if the experiment is done correctly at-home.

3. The game should be compatible with users' computers which has different properties.

4. Data collected in the game should safely acquire by the researchers.

In the design, different types of tremor are considered so both essential tremor and Parkinson's disease patients can play. Different types of operating systems are considered. To prevent implicit racism, the color of a hand in the Hands and Arrows version was created in grey color. Similarly, to make the game color-blind friendly, no transitions from green to red or vice-versa were used on objects. People who don't know English are not considered. To address the problem (1) it is thought to design the game with images showing the actions they would take. Also, adding an intro video was planned. For problem (2), there were several checks planned throughout the game. For (3) and (4), the game planned to be published online. So that the data would be obtained simultaneously while the user is playing, and game would be working on any computer.

3.2 My Responsibilities

My responsibility was developing the Reaching Task game. Of course, I usually got ideas from my group mates and they thought me lots of things especially in the neuroscience field. At the end of the day, I developed the game in the light of Cagnan Group's decisions. Here is a list of the things I have done:

- Making the games scripts run more efficiently
- Differentiation and implementation of four versions
- Making the game adaptable to Leap Motion software

- Collecting data in the end

3.3 Methodology/Tools

The needs and things to avoid were determined in the very first week. But the project evolved for 2 months. After each step I have taken, it has been discussed in the group, and depending on the feedback I went back and modified the step or continued. There have been meetings weekly and we nearly discussed every day about the project. The tool we used to keep contact and have meetings was Microsoft Teams. For the Reaching Task Game, I used Unity 2019.6f.4. For hand detection with the Leap Motion, Orion (4.0.0) Software were used.

3.4 Details

The game is based on reaching the balloon by the index finger on the screen (Figure 3.1). The hand position is detected by the Leap Motion and mapped to the computer screen. But since the screen sizes of different computers are different, a Calibration should be done before the actual game. The only purpose of the Calibration game is to map the index finger position to the different parts of the screen.

Figure 3.1: Balloon turns to red after pointing with the index finger.

In the design of the Reaching Task Game, the flow of the game could be explained as follows:

1. Rest: User looks at the computer screen and waits for the instruction (Figure 3.2).

Figure 3.2: Rest Image in the game flow to make user rest while hands are on chair rests.

2. Posture: With the instruction, user takes the “posture” position. The reason for this step is for detecting the tremor before the hand movement to the target (balloon) (Figure 3.3).

Figure 3.3: Position called ”Posture” image in the game flow to make user mimic the posture.

3. Hold: A cue appears to give an idea of where the balloon is (Figure 3.4).

Figure 3.4: “Hold” image where user waits for the appearance of the balloons in the ”posture” position.

4. Reach: User reaches the balloon and tries to pop it. The balloon pops if the correct balloon is decided and the user can hold his/her index finger on the balloon (Figure 3.5).

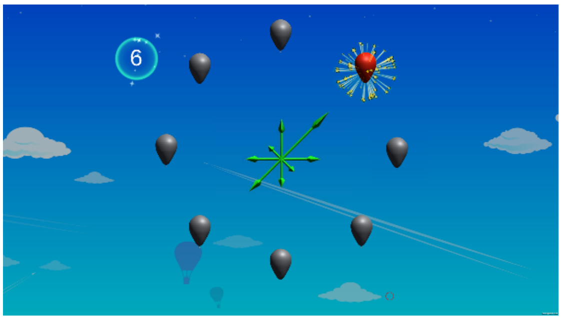

Figure 3.5: "Reach" image. Balloon pops since the correct balloon is chosen and user could hold finger on that balloon.

.

This flow repeats itself with some breaks and for different number of iterations. In the images, Arrows version is used. However, other games have a very similar manner with some changes.

3.5 At-home Testing

For each version of the Reaching Task Game, it is aimed to make the game suitable for at-home testing. For this purpose, those changes were made:

- Images were added for instructions in the game.
- It is made compatible with both 32-bit and 64-bit Windows operating systems.
- An introduction video was recorded and put at the beginning of the Calibration game.
- Feedback according to the player moves are given in the Calibration game.

- The need for Matlab to run the game with sequences of conditions eliminated. Instead, a random sequence is generated at the beginning of the game determining the intermixed conditions.

3.5.1 Considerations

1. **The user has to have a Windows computer.**

   To be able to play the game, the user needs to have a Windows computer. Unfortunately, the latest Leap Motion software available for macOS is 2.3.1 but 4.0.0 was used in the game. Therefore, it wasn't possible to convert the game for macOS without making changes in the properties used for hand tracking. That's why unless we send a tablet or make the game online, the user needs to have a Windows computer to play the game.

2. **If we will be using the Calibration game, there should be someone else (without the tremor) to play with it before the Reaching Task Game.**

   Since the calibration game makes the mapping by tracking data sensitively, if a person with tremor plays it, the mapped coordinated in the game won't be accurate enough. So, a tremor patient can play the Calibration game but with down sides to being seen during the actual game.

3. **Leap Motion should stay at the same place after the calibration.**

   Calibration game does the calibration by taking the hand position data each frame and generating the calibration keys. Hand position is taken respectively to the Leap Motion since it detects the hand by the infrared light it radiates. Leap Motion doesn't know the screen size or position so the calibration keys are all dependent on the hand position according to the Leap Motion. Therefore, if the position of the Leap Motion changes, calibration coordinates won't map the screen anymore.

4. **There are no assumptions about the users' skin color.**

   This is the reason why a grey hand was used to prevent implicit racism.

3.5.2 Challenges and Potential Solutions

1. **Challenge: Calibration Game**

   Calibration game is necessary to do the hand position mapping to the screen with good accuracy. Since the screen size of the computer, the leap motion position, and the relative screen position to the leap motion changes for different users, the only way to ensure the perfect mapping accuracy is by using the Calibration game (as far as we thought). In the "HandsAndArrows" game, calibration is done making the use of Leap Motion Unity packages come with Orion software. However, it didn't work as good as the Calibration game. Since the versions with the Calibration game were much better in terms of mapping, it is decided not to use HandsAndArrows game for at-home testing.

   In lab settings, it is easy to do the calibration in one computer and test the game with patients using that calibrated computer but in at-home testing it is much more harder. There are several reasons for this:

   - There is a need of a person (preferably a person without a tremor) to do the calibration.

   - Calibration game is sensitive and tends to record data continuously between a time interval. If the Leap Motion cannot capture the index finger position long enough (I fixed it to 100 frames per position after the changes). The process of trying to make the leap motion record data during the calibration may be exhausting for the user. This is tried to be solved by giving error messages to the user about the leap motion recording status.

- Some user errors may happen. This is tried to be solved by putting an introduction video at the beginning of the game.

Potential Solution:

The safest solution would be sending a device to patients' homes (ie. tablets) after we do the calibration for that device. Although, this requires a good leap motion position according to the device screen. So, the leap motion and device positions should be clearly specified.

2. **Challenge: Data Acquisition During At-Home Testing**

It is an advantage to be able to do multiple trials at-home testing, perhaps daily trials. But the challenge is, since the game and the leap motion will be sent to the patient, the data collected during the game will be in the user's computer. After some time, the data should be sent back to the Cagnan Group in order to analyze and this process takes time. Also, since the data would be collected after patient completes the trials a problem in the data might not be noticed before.

Potential Solution:

Making the game online. This should be possible even though I wasn't able to do it. The biggest issue I faced was using the websockets to collect the Leap Motion tracking data through web. Although the game implementation as a web app is theoretically possible, I couldn't find any examples of Leap Motion web app game built with Unity. The web apps for Leap Motion were always done using Java but never Unity. This is suspicious but I still think that it is possible to do it. Perhaps with a websocket implementation in the game project.

3. **Challenge: Leap Motion Position**

Leap Motion position is incredibly important since the tremor should be captured during the posture and while reaching to the balloon. User needs to be comfortable while playing so we cannot design the user movements according to the Leap Motion position. Therefore, the Leap Motion should be positioned in a smart way to capture the user's hand throughout the game. For laptops, the best position I discovered is the space bar to detect the hand during posture and reaching to the balloon by index finger.

Potential Solution:

One solution is to explicitly specify the range of the Leap Motion position with respect to the screen. The range could be written in the manuel with its visualization. Second solution needs more setup but solves the issue better. That is to send a 3D printed object to put the leap motion on. It is easy to design and 3D print the object. Also, it doesn't have a high cost with a convenient filament (such as PLA). According to the best position determined for different computers, there could be printings for laptops and desktop computers.

4. **Challenge: Leap Motion Joint Registration Time** While playing the game, it takes a few seconds for the Leap to capture the joints and register the hand in the game. This results as, even though the user points to the correct place, the Leap Motion may not understand it in the beginning. Sometimes, the Leap Motion can only register the hand when it sees all the fingers which is the worst-case since the game is designed to play with pointing finger all the time. This causes a time loss before popping a balloon so that user may not be able to pop or pop it with a delay.

   The time passes until the correct detection of the hand may end up in the wrong analysis in the collected data.

   Potential Solution:

   Potential solution would be putting dummy seconds at the beginning of each trial to make sure that the Leap Motion correctly registers all the joints. The good side is, there is no known impact of tremor while registering the hand to the Leap Motion. So, a tremor patient can safely put her/his hand above the Leap Motion for a few seconds and start the actual trial (posture-¿reach) after the registration of the joints.

3.6 Versions

3.6.1 Hands And Arrows

In this version, an updated version of the Leap Motion Software's Unity package was used. This package enables to have a default calibration done by the Leap Motion which eliminates the need for the Calibration before the actual game starts. Bad side is, sometimes the calibration is not accurate. To see and experience the default mapping behavior of the package, a "Playground" section was added to the game (Figure 3.6). In this option, a person can move hand freely and see the virtual hand on the screen where the Leap Motion maps to. There are 8 arrows in each trial. All have different lengths and are distributed by the Gaussian distribution. After the appearance of the arrows, user predicts the future balloon position. Unfortunately, this version was not working as it was expected in different computers with different screen sizes, so it is not intended to be used in the future for now.

Figure 3.6: Hands and Arrows version, Playground mode

3.6.2 RDK

In this version, so-called "random dot kinematograms" were used. They can be explained as moving dots, coherent/signal dots are moving to a certain location whereas incoherent/noise dots are moving to random locations (Figure 3.7). They are commonly used in psychology experiments. This version is made to get a better understanding of the output data of the game in psychology terms. But turned out to be potentially eye straining in practical usage. The reason is, in psychology experiments, RDK's cover the whole screen but in this version, it could only cover a small part of the screen to give room for the balloons (Figure 3.8).

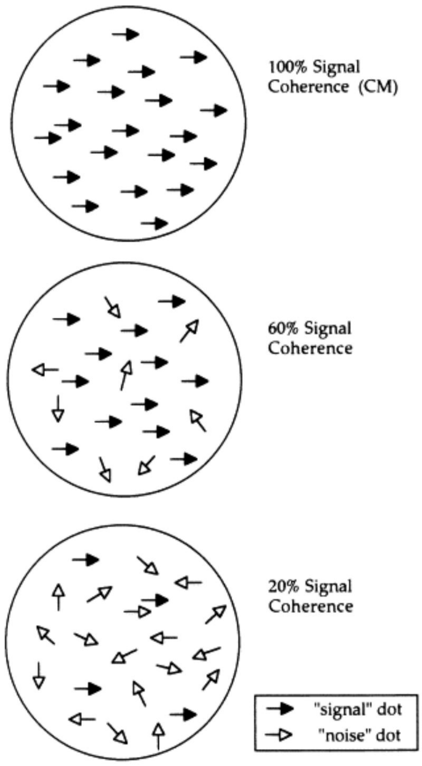

Figure 3.7: Random dot kinematograms given the signal and noise rates

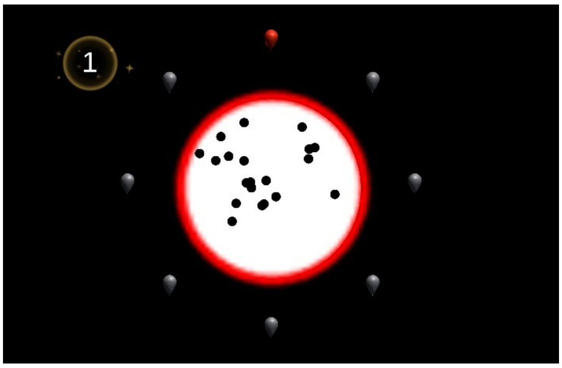

Figure 3.8: RDK version. Dots are moving and giving clues about the correct balloon location. The pointed balloon (by the user) turned to red.

3.6.3 Arrows

This version has the certainty parameter as arrows (same as in the Hands and Arrows game) (Figure 3.9). However, Arrows had the same game flow as the RDK game. The only difference between the RDK game and the Arrows game is the type of cue for predicting the balloon location.

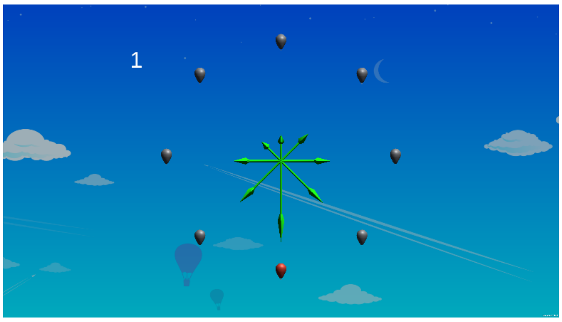

Figure 3.9: Arrows game, the balloon at the bottom is pointed by the user and turned to red from grey.

3.6.4 Fixation Cross

In this version, “Hold” step was simplified to make the game much easier to understand for the user. Instead of using different sized arrows or random dot kinetograms, a fixation cross was used (Figure 3.10). After the fixation cross disappears, the balloon appears and the user points to the balloon (Figure 3.11).

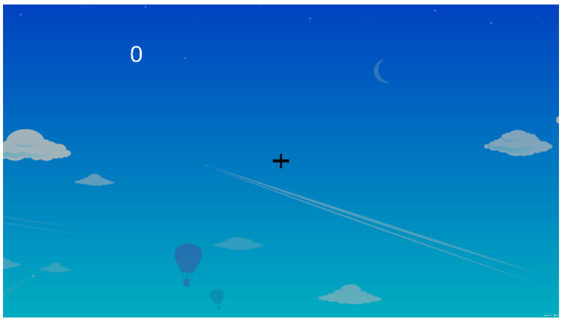

Figure 3.10: Fixation Cross

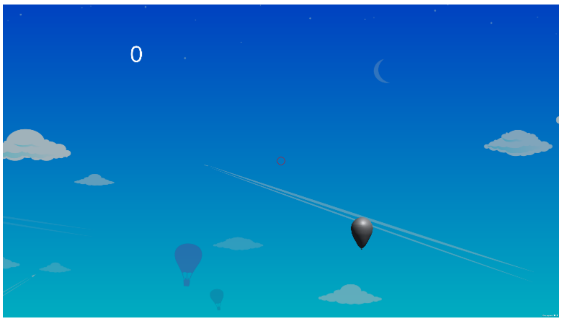

Figure 3.11: Balloon appears after fixation cross disappears

3.7 Parameters

There were two parameters decided as precision and certainty. Different conditions are produced with combinations of low/high uncertainty and low/high precision. This could be thought of as a 2x2 matrix. To have a better understanding, here is the list of the conditions:

1. Low Uncertainty, High Precision
2. Low Uncertainty, Low Precision
3. High Uncertainty, High Precision
4. High Uncertainty, Low Precision

For each game, it is made sure that each condition repeats the same amount of times. This is achieved by creating a random condition array in the beginning and iterate over the array during the game flow.

3.7.1 Precision

In all of the versions, precision was determined by the balloon size. For High Precision, a small balloon was used since it is harder to point a smaller balloon, especially for a tremor patient. For Low Precision, a big balloon was used.

3.7.2 Certainty

Low and High Certainty was assigned to different objects in different games. For the version “Hands and Arrows” and “Arrows” certainty was determined with the different arrow lengths determined by Gaussian distribution. The variance was high for High Uncertainty which made it harder to predict the balloon region. For Low Uncertainty, the variance was low so most of the time it was easy to predict the balloon region. For the version “RDK”, Low/High Certainty was determined with the low/high ratio of coherent dots respectively. For the version “Fixation Cross”, there isn’t a certainty parameter.

Since the only visual cue is the fixation cross image in this game, there is no cue to give the certainty level for predicting the balloon position.

4 Results

The Project has been completed. However, all the points aimed couldn't be implemented. It is designed for home testing, four versions were successfully developed,it made compatible with 32- and 64-bit Windows. The recorded videos of different versions of the game can be found here: https://drive.google.com/drive/folders/1N0t8Muwu-hpKz8H19AbqkBUOrsw1bvAl?usp=sharing The biggest thing I couldn't do was publishing it online. Leap Motion data acquisition through the web had some problems with the web sockets. Not publishing online caused another issue. Since it couldn't be published online, there had to be different builds of the game for different operating systems. I didn't have time to implement the game with a lower software for macOS. In the end, the internship was completed with the game only compatible with Windows.

5 Conclusion

The Reaching Task Game was an interdisciplinary project which was related to clinical neuroscience, computer science, medical sciences, and engineering. The aim was, with at-home testing, investigating the sensory cue impact on decision making for patients with tremor: mostly Parkinson's disease and essential tremor. It was implemented using Unity with the Leap Motion device. The points needed for at-home testing were mostly satisfied with several modifications. In the end, I completed the internship with the implementation of four versions. Earlier version of the Reaching Task Game which uses arrows for the uncertainty parameter was improved as "Hands and Arrows" and "Arrows" versions. RDK version was built using random dot kinematograms to change the quantity of uncertainty. In the Fixation Cross version, uncertainty as a parameter is not used for the sake of simplicity for at-home testing. Future steps would be getting the necessary permissions to use it on patients and publishing the game online. After acquiring the patient's data, analyze outcomes would be useful for the field and would bring broader knowledge.