Intuitive Physics in GUI Elements: Windows

Pedro Colón-Hernández – pe25171@mit.edu

# Abstract

There has been recent work on attributing predictive looks of people in 2D simulations, to the possibility that humans have an innate intuitive knowledge of physics [1]. However, how do humans apply this intuitive physics to non-physical, manipulable entities such as windows in a graphical user interface (GUI). I believe that this is through a stripped-down version of the innate physics that people have. To test this, I propose a simple Hidden Markov Model (HMM) [2] similar to the one in [1], in which people’s gaze positions are classified as predictive, looking, reinforcing, or other. In this writing, I give the details of the model, and propose some experiments to try the model out, along with the results of the experiments and the comparison of the model.

# Introduction

It has been proposed that people have an innate knowledge of physics which they utilize to simulate behaviors in their environment [3]. There has also been research on how this intuitive model of physics is applied to situations such as trajectory of objects and collisions to help explain causal and counterfactual behavior [4] [5]. However, this model of innate/intuitive physics has not been applied to explain behaviors or understanding for non-physical entities such as GUIs. GUIs have elements that appear out of thin air, elements that defy gravity, and elements that can be resized without side effects, yet we are able to drag things from place to place and manipulate some of the elements in a semi-physical way. This type of GUI is what is called a direct manipulation interface: an interface in which a system provides representations of objects that behave as if they are the objects themselves [6]. The manipulation in some cases is done using actions that correspond at least loosely to manipulation of physical objects. In this work we try to model the usage of a very reduced version of the innate physics (simulating trajectories with velocity and without gravity) on a task of moving GUI windows to a certain place. We say that the interaction happens in a similar manner to the forward and inverse models in [7] such that people predict the future actions by simulating what will happen and looking retrospectively at the causes.

# Hypothesis

The hypothesis that I am looking into will be that when controlling an interface (in this case a window) users simulate (with a somewhat limited intuitive physics simulation) what is going to happen, based on the current position of the mouse and window, and possible future positions. I hypothesize also that users look back to make sure that their action (dragging) in the mouse performs an appropriate movement on the window they are targeting. The way to give some evidence for this is that the eye movements of people will be ahead of the window that they are controlling and may occasionally shift to behind or on the window to verify the behavior. If this behavior were not present, I would expect that a user would immediately jump between target and destination without looking at the trajectory, or that the users vision stays with the mouse or with the window (i.e. there is no trajectory calculation).

# Experiment

The experiment that we propose has 5 parts. The following subsections give the details of the parts. Overall, the user’s goal is to drag the windows to a red zone. The experiment was developed as a web application with a landing page that links to each of the experiments. The code (and the experiments) is available at: <https://github.com/pedrocolon93/cocosci_phys_gui.git> . When the parts begin, an alert is displayed with the instructions for the experiment. The user is then instructed to train the web gazing system [8] by clicking on the mouse on different parts of the screen, while following the mouse with their gaze during the training. This is done only once. Once all of the parts are done, the user is sent to a feedback form and asked some follow-up questions.

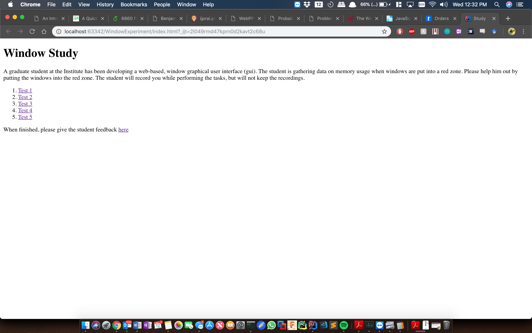


Figure 1 - Experiment Landing Page

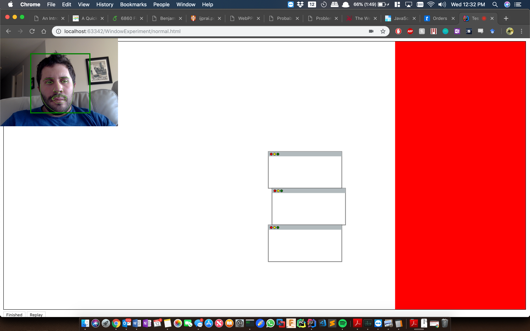


Figure 2 - Normal Window Activity

## Normal Window Behavior

### Description and Expected Results

In this part, we tell users to move the windows to the red zone. There is nothing special about this activity, and we use this as the base case. We expect users to alternate between looking at the mouse, looking at the window they are dragging, and looking at a simulated future position of where they are dragging.

## Inverted Window Behavior

### Description and Expected Results

The twist in this experiment is that the windows now go the opposite direction of where the mouse is going. In other words, the dragging is inverted. We expect that initially, since the user does not know the behavior, that the user will start to perform as in the Normal Window Behavior part. However as soon as the user realizes that the window is inverted, the gaze should begin to alternate from the mouse to the window, and (primarily) to where the window will go, and where the mouse will go (i.e. the user will predict where the mouse will be and where the window will be).

## Gravity-infused Window Behavior

### Description and Expected Results

The twist in this experiment is that windows now obey gravity whenever they are released or not being dragged. What this means is that the windows are constantly falling unless they are grabbed. We expect that the initial user gaze will be on the windows falling, however that will be for a small period of time, and then when the users begin dragging the window, there may be some predictive looks to the bottom of the window just to verify if it is falling or not.

## Disappearing Window Behavior

### Description and Expected Results

The twist in this experiment is that windows are no longer continuous when they are being dragged. What this means is that a user will start dragging a window and at constant intervals, the window will appear and disappear. This will give the test the feeling that there is some lag. We expect in this test that the user’s gaze to keep “following” the window that disappears and reappears. In other words, gaze should follow the trajectory of the window even if it is not visible in the scene.

## Self-Guiding Window Behavior

### Description and Expected Results

The twist in this last experiment is that the user will have no control of the window. The window will go by itself in the direction of the red zone until it is within it. We expect that users should follow the window and should predict its trajectory. If there is a chance that the window collides with another window, then it is possible that the user may perform some counterfactual looks expecting the windows to bounce, in addition to predictive looks of where the window is going.

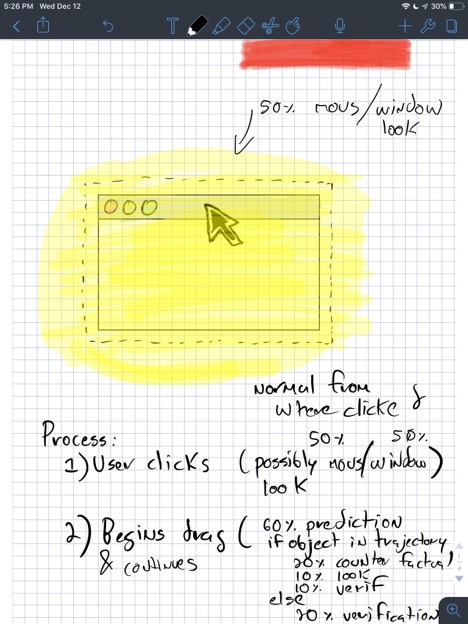
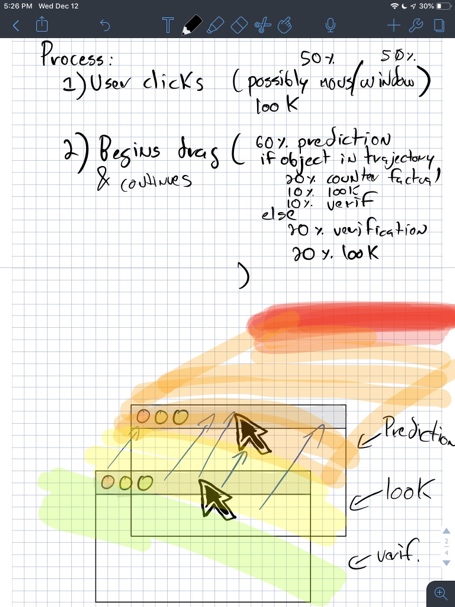
## Feedback Form

The questions that we asked in this section and the reasoning for them were the following

|  |  |
| --- | --- |
| **Question** | **Reason** |
| What did you like about the system? | See if there is any inclination that people may have in common |
| What did you dislike about the system? | See if there is any expectation/simulation that people had that was not achieved |
| Was there anything out of the ordinary? | Try and get a feel for what users were expecting/simulating and how these expectations were not met |
| Did anything impede your completion of the tasks? | See if there was any limitation in the mental modelling |
| Were you able to finish your tasks successfully? | Same as above |
| Which of these did you perform on the windows? (Select all that apply):Drag, Guide, Push, Throw, Other | Try and see how users perceive their actions. |

# Model

The model that we propose to analyze the data that we got from our experiments is similar to the one described in [1]. It is a Hidden Markov Model whose hidden states are the type of look that is being given, and whose visible outputs are the X,Y positions of a user’s gaze. However, the types of looks are different, and we utilize a simplified simulation that simply calculates position based on velocity. We started off with the following two sketches:

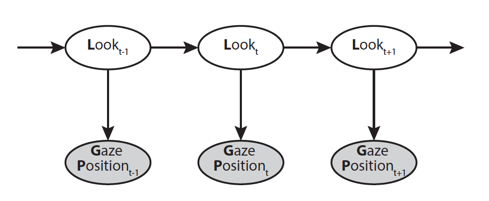
On the left sketch we have an initial condition which is a user is hovering over a window or is about to click it. Since the user has not started performing, we expect that the looks that the user should give are near both the mouse and the window since the user is controlling the mouse, and about to control the window (the yellow area is where most of the looks should be). Additionally, we add an “other” look for the looks that do not fall into these. Once the user takes control of the window by clicking, we switch over to the next part of the model which is pictured on the right sketch. As the user moves the window, he or she will do 6 types of looks: a prediction look (which should be ahead of where they are going), two “looking” look which is simply looking upon the window or the mouse (similar to the left sketch), and two reinforcement looks which is that the user may look behind to make sure that the action that they are performing is as expected. Additionally, we add an “other” look to handle any other possible looks. Putting this into a Hidden Markov Model we have a model whose hidden states have an emission likelihood that is switched depending on the current action that the user has. The visible states of the model correspond to the gaze position. Now for this model we need to add in the likelihood that an observation corresponds to specific look. We note that as mentioned before, these likelihoods vary based on whether the user is dragging an object or not. We based our likelihoods once more on [1]. In the case that there is nothing being dragged, we have that a user could be giving a “looking look” (to a window or a mouse), or an “other” look.

Figure 3 Visualization of Hidden Markov Model taken from [1]

In the case that a user is dragging a window *W*, the user could be giving a “looking” look, a predictive look, or a verification/reinforcement look. The likelihood of an “other” look is the following: where max(x) and max(y) refer to the maximum dimensions of the test canvas. We say that the likelihood of a looking look (for window W or mouse M) is at a point t in time, d is the distance between the center of a window or mouse, and a gaze point . Additionally, for a predicting look we have the following . Lastly, for a reinforcing look, we have the following likelihood where and for reinforcement looks and for predicting looks. The future positions are given by taking the actual future positions and adding some noise to them to represent uncertainty about the predictions. A better approach in future work may be through the velocity where the object will be in the next frame with some noise for uncertainty. If the user is not dragging the window, we have “other”, “looking” looks.

# Results and Discussion

We ran the experiment with 3 people. We acknowledge that further testing may be needed. After running the experiments and replaying the data along with the model predictions for the looks we found the following. The most likely set of chains for the experiments do not include a reinforcement look. What this means is that possibly there is no actual reinforcement look, and that the gaze tracking library may not be too precise. It may also be the model that we are using, which we try to take into account the previous 5 moves, may be too limited, and we may need to modify it to have a larger window of previous moves. Additionally, a surprising find is that in the activity that involves gravity, the user seems to be trying to predict where the window would be going if gravity were affecting it as he or she is dragging it. We also see in the case where the window disappears, that the user’s gaze follows the mouse, seemingly under the assumption that the window is linked to it. Some of this behavior seems to indicate that the simulations in people’s minds may have some tuneability (they can turn on and off gravity), and that people at least do some kind of continuity modeling (when they follow the mouse when the window disappears), however more thorough modeling and a greater number of subjects are needed to verify this.

In the feedback questionnaire, some interesting responses were in the question of whether there was anything ordinary or not in the test, and people seemed to be aware that the system did not behave as they expected with the self-moving windows, the inverted windows, and the windows with gravity. Another interesting thing was that people felt they were dragging and guiding the windows (which are some causal terms). Some interesting remarks are that this area (intuitive physics in GUIs) may be a more natural way to investigate the intuitive theories domain. Although it may be an unnatural (not in the real world) environment, a lot of people have become used to these direct manipulation interfaces and within them we have the physical like interactions of dragging, dropping, and moving.

# Bibliography

|  |  |
| --- | --- |
| [1] | T. . Gerstenberg, M. . Peterson, N. D. Goodman, D. A. Lagnado and J. B. Tenenbaum, "Eye-Tracking Causality:," *Psychological Science,* vol. 28, no. 12, pp. 1731-1744, 2017. |
| [2] | Z. Ghahramani, An introduction to hidden markov models and bayesian networks, 2001. |
| [3] | T. Gerstenberg and J. Tenenbaum, Intuitive theories, vol. Oxford handbook of causal reasoning, 2017, pp. 515-548. |
| [4] | T. . Gerstenberg, N. D. Goodman, D. A. Lagnado and J. B. Tenenbaum, "Noisy Newtons: Unifying process and dependency accounts of causal attribution," *Cognitive Science,* vol. 34, no. 34, p. , 2012. |
| [5] | T. . Gerstenberg, N. D. Goodman, D. A. Lagnado and J. B. Tenenbaum, "From counterfactual simulation to causal judgment," *Cognitive Science,* vol. 36, no. 36, p. , 2014. |
| [6] | E. . Hutchins, J. D. Hollan and D. A. Norman, "Direct manipulation interfaces," *Human-Computer Interaction,* vol. 1, no. 4, pp. 311-338, 1985. |
| [7] | P. . Agrawal, A. . Nair, P. . Abbeel, J. . Malik and S. . Levine, "Learning to Poke by Poking: Experiential Learning of Intuitive Physics," *arXiv: Computer Vision and Pattern Recognition,* vol. , no. , pp. 5074-5082, 2016. |
| [8] | A. . Papoutsaki, P. . Sangkloy, J. . Laskey, N. . Daskalova, J. . Huang and J. . Hays, "Webgazer: scalable webcam eye tracking using user interactions," , 2016. [Online]. Available: https://par.nsf.gov/biblio/10024076-webgazer-scalable-webcam-eye-tracking-using-user-interactions. [Accessed 12 12 2018]. |
| [9] | P. . Agrawal, A. . Nair, P. . Abbeel, J. . Malik and S. . Levine, "Learning to Poke by Poking: Experiential Learning of Intuitive Physics," *arXiv: Computer Vision and Pattern Recognition,* vol. , no. , pp. 5074-5082, 2016. |