Interface: an interactive facade project

Robotic Building Fall 2019 workshop research report

Student:
Yongyi Wu (4791614)

Coordinators & Tutors:
Henriette Bier, Arwin Hidding, and Vera Laszlo

November 2019

I INTRODUCTION

Architects of the twenty-first century will shape, arrange, and connect spaces (both real and virtual) to satisfy human needs, according to William J. Mitchell. That means the commodity of architecture will be as much a matter of software functions and interface design as it is of floor plans and construction materials. In the era of informatization, architects have more opportunities while facing more challenges from outside field. Reflecting on the daily life behaviour and original interaction between people and built environment, we rethink the interactivity of basic building elements with new information and digital tools. This paper will try to discuss the concept, computation, materialization of an interactive façade project we did in Robotic Building workshop. And its reflection will be discussed in a wider context with my own graduation project. This workshop project was jointly completed by Yongyi Wu, Filip Romaniuk and Thierry Syriani.

II CONCEPT AND COMPUTATION

As a preparation of graduation, a premise of the design is that it can reflect the similar interests of the group members and the relevance to the direction of our graduation projects. So design started with a general question, what can we do if we have an adaptive system incorporating sensor-actuator mechanisms that enable buildings to interact with their users and surroundings? This question led to a very simple daily scene that a kind of smart curtain that reacts to the environment and users. We found that the window curtains on the 2nd floor of BK building are automated by certain principles, but they work against common sense and actual needs sometimes. Therefore, attention to this problem became a small but worth exploring and scalable architectural practice. Because we found that it actually shares many common principles with more complex adaptive system later.

The main concept is that an interactive shading system which can be controlled by both environmental parameters and user interaction parameters. These parameters work simultaneously but in different priorities. Environmental parameters are 1) Seasonal changes from weather data (fig.1), 2) Adaptation with local weather data (fig.2) and 3) Responds to changes in ambient lighting conditions (fig.3). User interaction parameters are 1) Proximity to the wall intensities the reaction (fig.4), 2) Movement along the wall vs static position (fig.4), 3) Posture: localized intervention with enhanced comfort and accuracy (fig.6). These rules work together to ensure the accuracy and flexibility of shading

In computational design phase, we focused on Environmental parameter 1 and User interaction parameter 2, 3 on account of time and actual effect, which are 1) Responds to changes in ambient lighting conditions, 2) Proximity to the wall intensities the reaction and 3) Posture: localized intervention with enhanced comfort and accuracy.

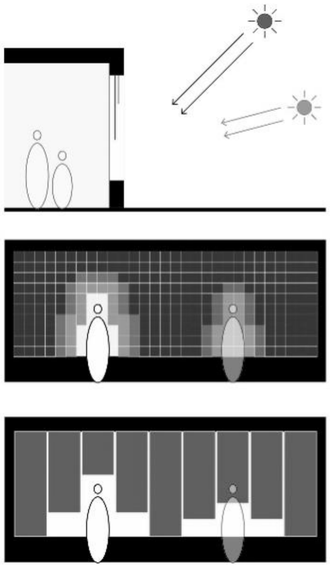

Figure 1

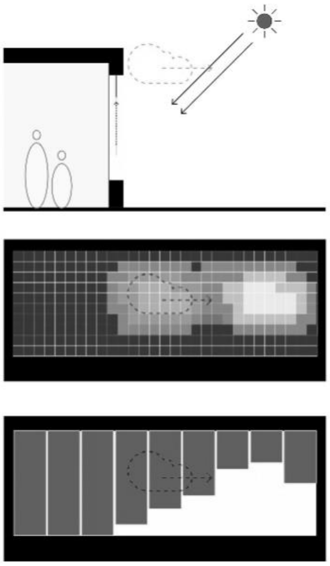

Figure 2

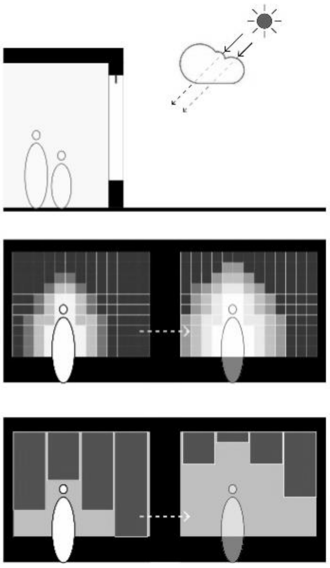

Figure 3

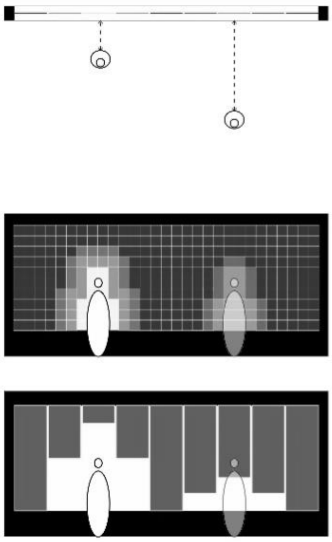

Figure 4

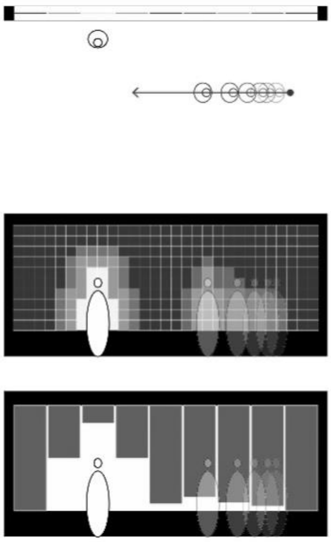

Figure 5

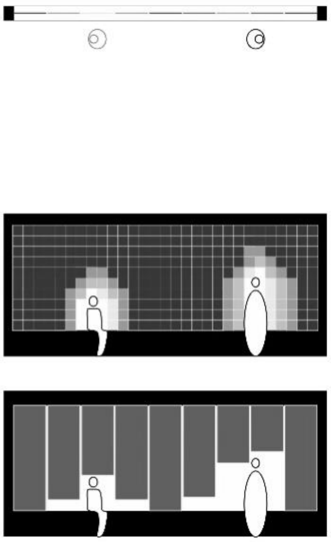

Figure 6

Computation was completed in Rhino/Grasshopper. First, the basic framework was set up digitally to represent the curtain scenario. Second, environmental parameter (light intensity at certain locations) and user parameter (user location and posture) were planned and preset manually to continue the programming. Third, we were finding the qualitative and quantitative relationship between given parameters and geometry deformation which is the curtain movement in this case. Curtain movement was simplified to the movement of point on the lower part of each curtain. Different parameter values represent different point heights, such as higher sunlight intensity values can get smaller point heights to simulate the real situation of curtains closing. The user position and posture parameters follow the similar principle but the mapping relationship is more accurate and faster than the environment parameters, and they have higher priority when they conflicts with the environment parameter. Fourth, some formulas were needed to coordinate the height relationship between different curtains, because in some cases, a certain curtain needs to adjust its height according to the adjacent curtains to obtain a coordinated visual effect on the internal façade. Last, point heights were transformed to other geometries to represent curtains. It should be pointed out that this way of representing specific behavior with simple values is also applicable in the real world, because the control of a curtain requires only one servo motor, and the rotation angle of the servo motor can be converted from the point height mentioned above. Computational design ensured that cyber-physical agents move according to selected parameters correctly (fig.7&8).

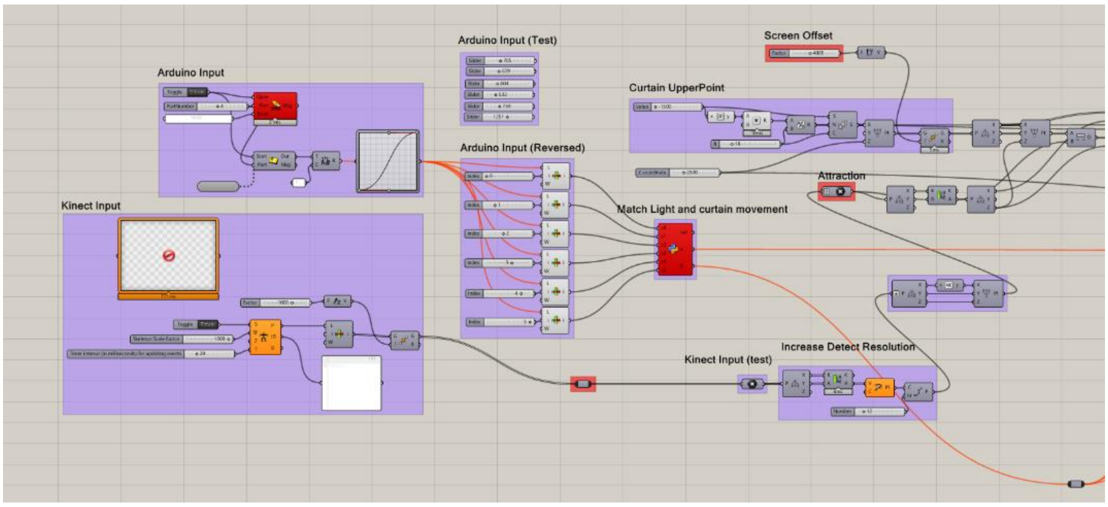

Figure 7. Grasshopper definition overview Part 1

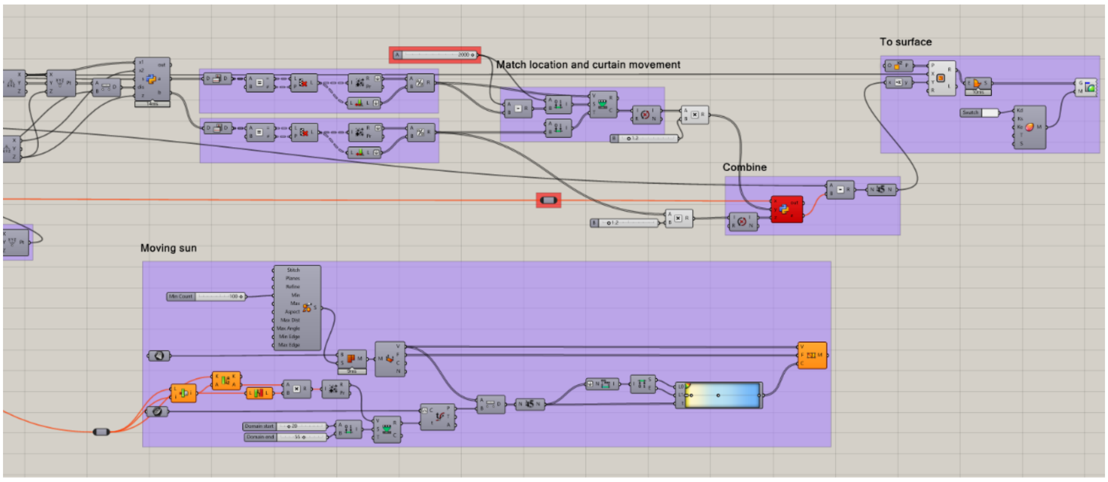

Figure 8. Grasshopper definition overview Part 2

III MATERIALIZATION

In materialization phase, some differences of opinion emerged. We planned to make the physical model since the mechanism was simple enough to build. But we decided to have a virtual presentation considering that it’s easier to apply to other interactive projects later. About the way to capture parameters, we also had some discussion and chose Kinect camera instead of proximity sensor in the end since the former can collect more information.

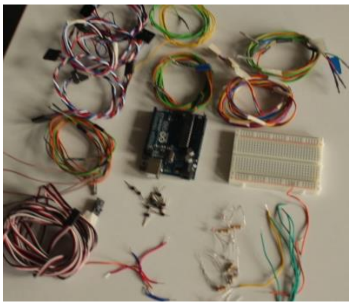

Figure 9. Arduino components

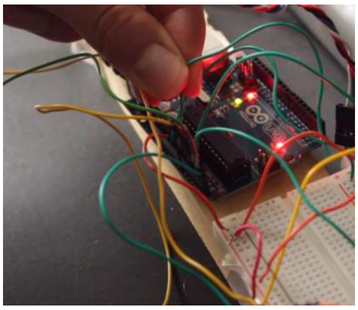

Figure 10

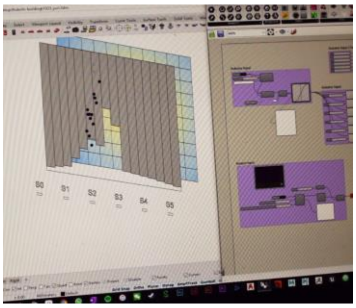

Figure 11. Grasshopper in working

The final physical setup consisted of a rear projection, an Arduino with several photoresistors and a Kinect camera (fig.9&12). Rear projection was used to avoid interruption from users’ shadow on the screen and create a near real experience when scale the image to proper dimensions compared to real people size. 6 photoresistors were equidistantly distributed in front of the screen to simulate light intensity values from 6 location on a window. Photoresistors were wired to an Arduino and Arduino was connected to Grasshopper with Firefly plugin (fig.10). Real time resistances from photoresistors were read into program as numeric values. Kinect camera was set on the other side, it can recognize the “skeleton” of people in front of the camera and read the locations of each point on the “skeleton” in the 3D coordinate System. These information were also transfer into grasshopper by Firefly (fig.11).

The final result worked as our expectation after debugging. Responded animation showed correctly within an acceptable delay corresponding to the digital signal in Rhino/Grasshopper on the screen. First, changes in the overall light intensity cause the curtain to rise and fall overall (fig.13) and local light intensity changes cause local curtain changes (fig.14). Based on this, second, when the user approaches the “window” (screen) to a certain distance (2m), he/she starts to influence the curtain and overpasses the environmental parameters. As the user keeps approaching the window, the curtain at the corresponding position will rise up to the highest height of the user's body part in front of that position to obtain a dynamic opening effect (fig. 15). This system works for two people maximum due to the default setting of Kinect that it can only recognize two people at one time (fig.16).

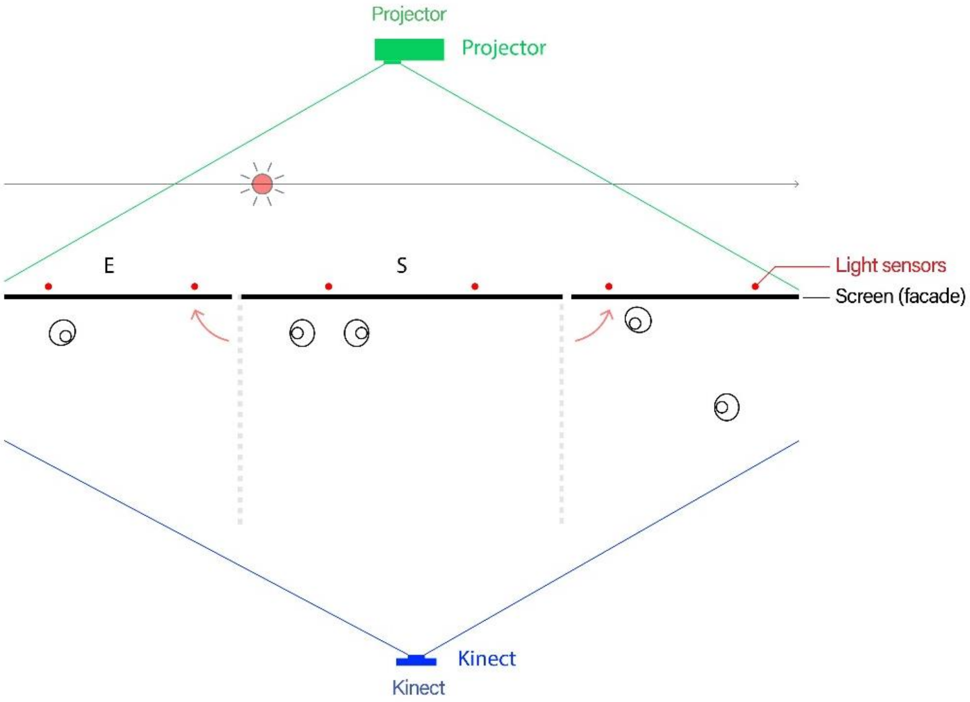

Figure 12. Physical setup

We also tested another design of a more complex interactive façade. This façade is composed of many small hexagons. Every time a user’s hand stays somewhere in front of the screen for a short time, the hexagon at this position will trigger a “twinkle”. Then, the hexagons around it reacts the same way, and so on, forming a visual effect similar to ripples. This experiment demonstrates the scalability of interaction design following the same basic principles and the compatibility of this physical setup with different interactive façade concepts (fig.17&18).

Figure 12. Overall light intensity reaction

Figure 13. Local light intensity reaction

Figure 14. One user

Figure 15. Two users

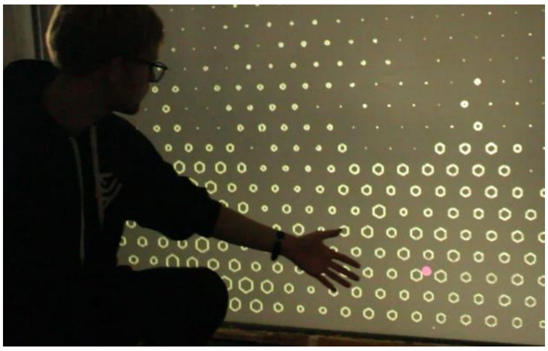

Figure 17. Ripple

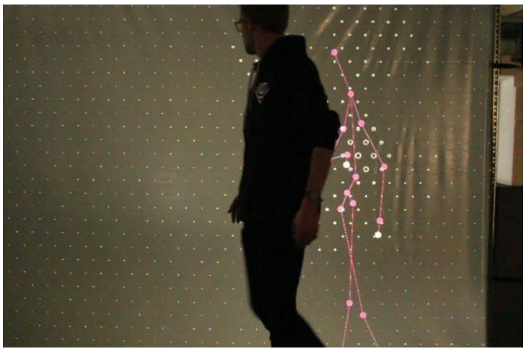

Figure 18. Detected “Skeleton”

IV REFLECTION

In this project, firstly it shows how fast prototyping can help research and design in the D2RP&O method. With certain degree of realization of the design (even a virtual approach), it can be quickly evaluated and redesigned. Every step is able to collect data from feedback would then again be directed to the design or production in the form of parameters and the test moves on in a loop till an optimized model is achieved.

This project is simple, but we believe that it can evolve or even materialize in many aspects of architecture because it shares many basic principles with many other interactive projects. The idea of interactive architecture has a long history. Since the invention of the computer, informational tools provide other possibility besides design and production, which is interactivity. Some architects and theorists imagined the merge between architecture and cyberspace in 1990s. Early practices used other media to strengthen the architectural experience, like Lars Spuybroek and Asymptote. Then the category of Interactive Architecture is gradually established based on the idea that creates a more adaptive environment with the cyber-physical system (CPS) (Rajkumar et al. 2010), a building or some parts of a building can be transformed by certain mechanisms with control from a computer. More and more media are added in this category like screen, light, fog, social media and VR.

These practices provided a good background for my graduation project, and at the same time this workshop provided me with a realistic entrance. In my graduation project, I will further develop interactive ideas and concepts, thinking about the deeper and broader impact on architectural space. A category containing different interaction types, implementation mechanisms, and application scenarios will be established to declare the possible impact of the Internet on society and lifestyle in the future.

Serval steps are planned to carry on. First, spatial narratives in a setting background, which are mentioned above, can be used to define the possible interaction between people and an intelligent environment. Second, the input-process-output loop of every interaction is supposed to be designed with specific action or device. These simple interaction loops are ready to be evaluated with animation or test with certain interfaces in the real world. Third, with the reflection on interactions and simulation of architectural performances, functional spaces for different interactions can be defined within the program. Last but not least, spaces organized themselves with certain computational tools and corresponding structure strategy. Thanks to the liberation of the brain and physical power by computers and robots, not only the design but the design process itself will change. The identity of the designer will also change, because these tools and methods will allow more people, the general public, to participate in the customization of the physical environment or virtual experience through an open platform.