Marcus Panozzo, Sharon Lin
Interactive Visualization- Final Project Report

Introduction

The motivation for this visualization is to create a user interface to assist people in moving large pieces of furniture. It can take a lot of time and physical effort to move large furniture through tight spaces like doorways. To reduce the time and effort, this visualization allows users to select dimensions for a doorway and a piece of furniture from a list. Then the software computes if the furniture can fit by calculating the largest cross sectional area of the furniture and comparing it to the area of the doorway. Thus, users will know if the furniture can actually fit without exerting as much time or energy trying to move the actual furniture.

Since most people would have to move furniture at some point, whether moving to a different residence, getting rid of old furniture, or buying new ones, this visualization is meant for the general public. It would help users figure out if it is possible to move a piece of furniture as is or if it must be taken apart to fit. This would be especially helpful if someone was moving residences and has to move many large pieces of furniture.

The goal is to create a user interface that is expansive and easy enough for users to use. Although it is not possible to compile a list of every existing piece of furniture, we gathered models of 24 items from Ikea varying in type and size. These items range from tables, beds, sofas, desks, cabinets, etc. This list is limited but more pieces can be added.

Background

One of the interesting papers we found was “Interactive 3D modeling of indoor environments with a consumer depth camera” by Hao Du et al. In this paper, the authors develop a 3D modelling system to allow users to build a 3D model of a room simply by scanning it with a camera. This system is meant for non-professional users who want to redesign or simply explore their space virtually. It also allows users to move objects around or add new ones, such as furniture, to the room. This is a much larger project that is beyond our scope but scanning furniture pieces to allow users to input their own furniture, instead of having them search for it in a list, would make for a great extension of our project.

We also found “The Information Cube: Using Transparency in 3D Information Visualization” by J. Rekimoto and M. Green, in which the authors introduce a 3D visualization technique for hierarchical structures, such as software structure, traffic control information, or network management. They used nested boxes to represent the structure and use semi-transparent rendering so users can see inside the boxes, to a certain extent, based on how transparent they are. The users can interact with these boxes to explore the information. For our project, rendering furniture semi-transparently can give users an x-ray view which could give them hints on how to move or disassemble the furniture.

Finally, we also looked into "Differential equations and exact solutions in the moving sofa problem" by Dan Romik, which could be helpful for backend computations. This paper offers a solution to an open mathematical problem posed by L. Moser in 1966. The question asks for a 2D shape of the largest area that can be moved through a 90° L-shaped corner in a hallway of unit width. This problem differs from ours in several ways. First of all, the sofa moving problem is an optimization problem since it asks for the shape of the largest area. However, we just want to figure out if a shape with some known area can be moved through a doorway, also with some known area. Another difference is that the problem is introduced in 2D but ours is in 3D. So we would need to consider how rotations can change how the furniture fits in the third dimension. However, these computations can still be helpful to look at if we develop more sophisticated calculations for fit and if we also add hallways and corners to our visualization.

Initial Design

Our initial design is a user interface that has an area for users to input door dimensions and select a furniture piece from a list. Then the door and furniture are displayed on the screen. When the user clicks on the Fit button, the menu disappears and the user is able to try to move the furniture. In addition, the user can adjust the camera angle, and rotate or translate the furniture. The user can click on the left arrow button to return to the menu screen.

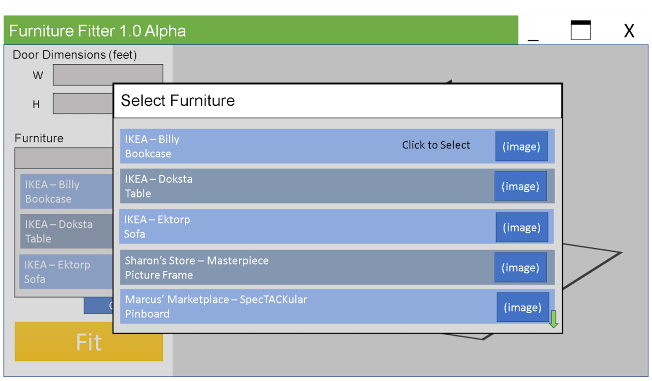

Fig 1- Input door dimensions and select furniture,

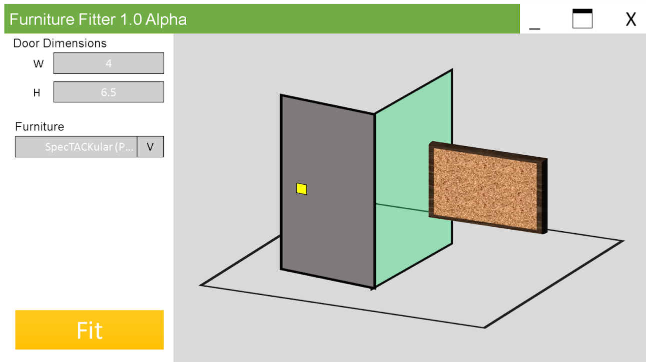

Fig 2- Door and furniture are displayed.

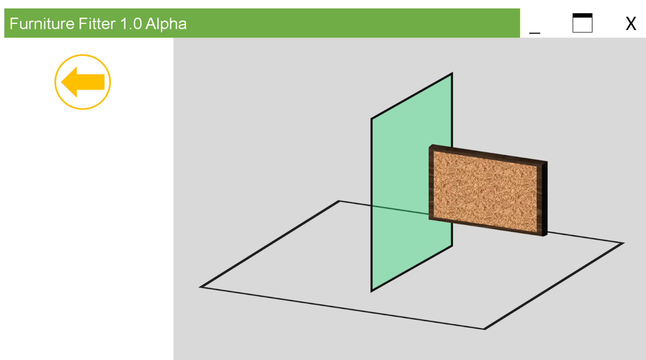

Fig 3- Users can try to move the furniture or click the back button to return to the menu.

Design Changes

To get this initial list of furniture, we were planning on web scraping some dimensions from Ikea and finding models of the web scraped furniture or creating them ourselves. However, finding 3D models that matched what we web-scraped was more difficult than we thought due to two reasons. First, there simply are not that many 3D models available for download online. Second, there were also some models of furniture that have been discontinued and therefore were not available on the Ikea website. So we decided to just use what we could find and edit the models if needed. We ended up getting the models from 3DWarehouse, where users can create an account to build and share 3D models. Several of the collada files we downloaded needed some minor adjustments due the scale being inaccurate or some parts being displaced. We used Blender to edit these models. The web scraper could still be useful for future work if we want to add new pieces.

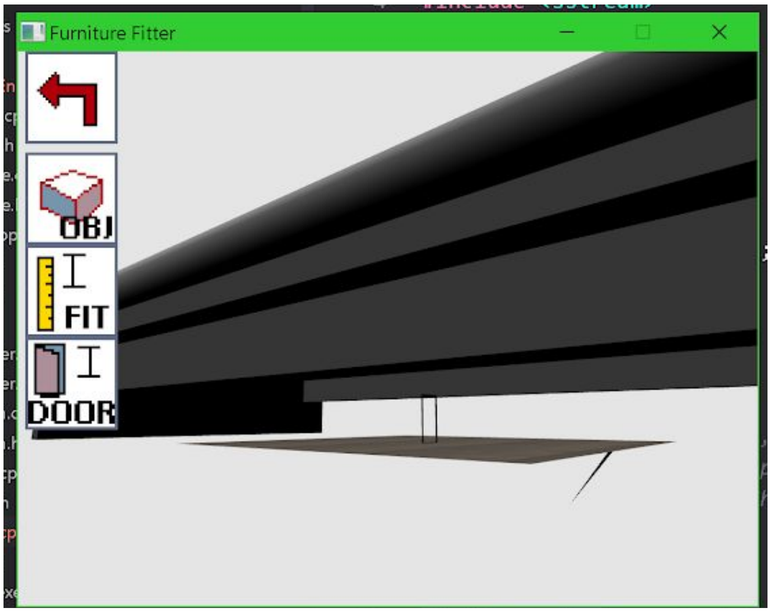

Fig 4 - What happens when models are not adjusted to be scaled to the correct units.

We also changed the appearance of the user interface. The layout of the window is largely the same, however the controls are simply on top of a fullscreen version of the 3D visualization. Also the fidelity of the control textures is low and the text labels might be slightly misaligned. This is due to initial attempts at using an existing gui library such as nano not compiling with the project properly; as a result, we created our own barebones gui on top of OpenGL directly with simple click and keyboard interaction. Because of issues with dynamically rendering text a user may type out on the screen, the user is given three door dimension presets to select from. Finally, instead of the door lighting up green if an object fits, a green check mark is displayed on the gui. Likewise for a furniture piece that does not fit, a red 'X' is displayed, indicating the furniture piece will not fit even with rotation.

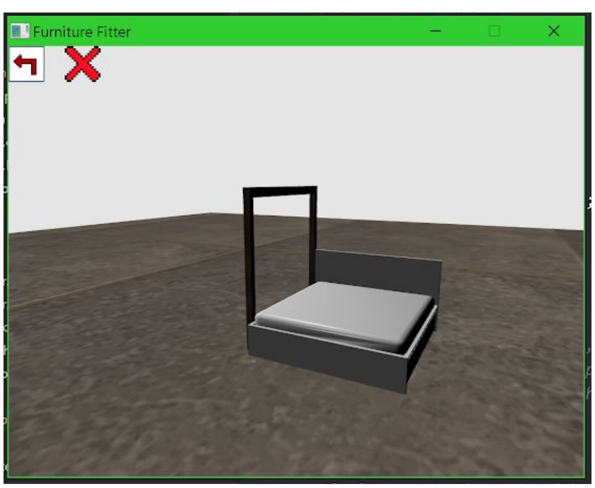

Fig 5 - Result of trying to fit a king sized ikea bed through one of the doorways

Final Design

The final implementation of our design allows for three main things: furniture selection, door dimension selection, and fitting. The three main functions are displayed in large icons on the gui as well as a back button to either undo the last action or to minimize the gui to give full visibility.

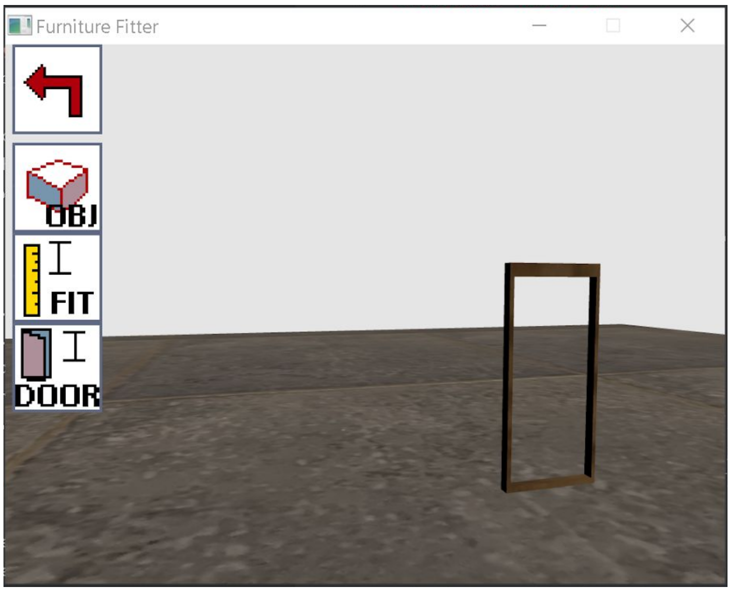

Fig 6 - Screenshot of the FurnitureFitter application window.

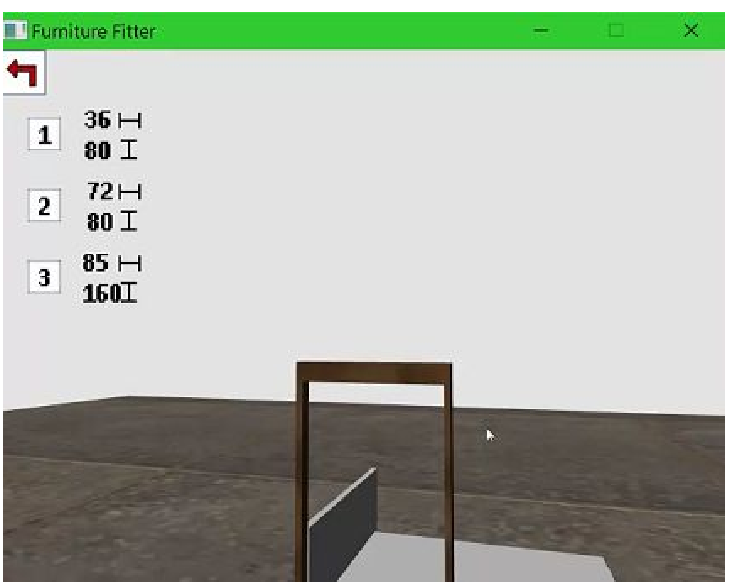

Fig 7- Door dimension options in inches.

The application is written in C++ and was compiled with the MinGW toolkit. This was done in order to facilitate easier cross compatibility in the future as well as with other libraries. The libraries used include GLEW (OpenGL extension wrangler), GLFW (Graphics Library Framework), GLM (Opengl Mathematics), and stb_image. Some code for importing object files was referenced from opengl-tutorial.org; assimp was originally going to be used however the compilation ran into some issues with assimp repeatedly. In order to move the project along we switched from importing any model file type to converting files into “.obj” format to work with the custom importer. Other code came from old projects. The user interface textures as well as textures for the floor and door were created by hand in paint.NET or photographed, rotated and then cropped.

The program is a simple OpenGL rendering loop with callback functions for GLFW to allow for responses to user mouse and keyboard input. The user can use the mouse to move the screen around and the up, down, left, and right arrow buttons on the keyboard to pan the camera in those directions. When the program first starts, the models and textures are loaded into memory and stored as vertex array objects, storing the vertex positions, the texture coordinates, and normal coordinates. The textures themselves are stored and linked with a texture unit when rendering each specific geometry. Basic shaders are also compiled from text files and loaded into memory.

To compute if the furniture would fit, we rotated the model’s bounding box to find the largest cross sectional area of the furniture that is less than the area of the door way. If there is no such area, then the furniture will not fit no matter how it is rotated. Otherwise, the furniture will fit.

One challenge while developing the application was learning the intricacies of making the rendering work. For example, when trying to develop a system for rendering text, the results were questionable. Originally a tile based system was used where each tile was two triangles with texture coordinates corresponding to a letter in a texture file. However, sometimes tiles would not render, other times they would render one half correctly and then distort the texture on the other half.

Another challenge during development was the user interface. Due to time and compatibility issues between the different libraries, a user interface had to be formed on top of OpenGL by hand. If more time was given or with more knowledge, a more effective user interface would be built.

Finally, development was impacted by some bugs on the machine used to build it on (a detachable surface laptop). The NVidia control panel settings have to be changed not to use the high performance GPU in order to render many of the models (it works fine without it), but for transparency effects this needs to be turned on. The reason for this is as of now unknown, in any case, it proves to be extremely frustrating.

Feedback and Future Work

We got several suggestions that would be interesting for future work. For example, implementing an x-ray view could give users a helpful perspective especially for oddly shaped furniture. In the future, we could explore letting the users change the opacity via a slider or implementing a fully transparent view to give users insight on how the furniture is structured.

People also suggested including hallways and corners, etc., which could be more bothersome than doorways. We really liked this suggestion but this would require more complex computations and rendering so we focused on doorways for this project. Someone suggested that we look into the sofa moving problem, which is somewhat related to our work. While the sofa moving problem is quite different, some of the calculations used could be useful. However, we would need to consider this in 3D as well.

One of the limitations of our current visualization is that we only have 24 models for users to choose from. Expanding this list would be a good idea for future work. However, not everyone has Ikea furniture so even if we included every item in their catalogue, most people would probably find it a hassle to search for something with similar dimensions as what they actually have. To make it more flexible, we could allow users to input their own furniture dimensions. Someone on Submitty also suggested taking photos to get the exact furniture details. The paper by Hao Du would be interesting to consider here and would give users much more flexibility to input custom furniture and door dimensions.

We got this feedback mostly from students and instructors, who are also part of our target audience. In addition, since college students have to do a lot of moving around during their time in school, they are an excellent group to get feedback from.

Who did what

For this project, Marcus created the application implementation, user interface, associated textures and rendering using OpenGL. Sharon wrote the web scraper for scraping Ikea furniture dimensions and found/edited 3D models.

References

Hao Du, Peter Henry, Xiaofeng Ren, Marvin Cheng, Dan B. Goldman, Steven M. Seitz, and Dieter Fox. (2011). “Interactive 3D modeling of indoor environments with a consumer depth camera”. *Proceedings of the 13th international conference on Ubiquitous computing (UbiComp ’11)*. Association for Computing Machinery, New York, NY, USA, p.75–84.
https://dl.acm.org/doi/abs/10.1145/2030112.2030123

J. Rekimoto, M. Green. (1993). “The Information Cube: Using Transparency in 3D Information Visualization”. *Proceedings of the Third Annual Workshop on Information Technologies and Systems*, p. 125-132.
http://www.vrarchitect.net/anu/cg/Import/zb/rekimoto/cube.pdf

Romik, Dan. (2017). “Differential equations and exact solutions in the moving sofa problem”. *Experimental Mathematics*. Vol. 26 p. 316–330.
https://www.math.ucdavis.edu/~romik/data/uploads/papers/sofa.pdf

Ikea. (2017). https://www.ikea.com/us/en/

almodoniar. (2014). IKEA HEMNES Glass-door cabinet with 4 drawers. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/bc08a79dd9f1fb862964041053b36a3/IKEA-HEMNES-Glass-door-cabinet-with-4-drawers

angelpe86. (2014). Bissa shoe cabinet. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/979392141fe9ea8c27b59d3ec270045/Bissa-shoe-cabinet

AlexC_HAL. (2014). LIATORP glass door cabinet. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/90166aba9d8bdc28a9f015dc2ef2c415/LIATORP-glass-door-cabinet

Nina S. (2014). IKEA PS 2012 Dining table 2. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/1893e7831f8852a2f51fa0238791f5dc/IKEA-PS-2012-Dining-table-2

redave. (2015). Ikea LERHAMN table. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/u237a5af7-b2b9-40d5-844a-d3f65a1af75e/Ikea-LERHAMN-table

Nina S. (2014). IKEA INGATORP Dining table. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/f886e51b9a0ef04ef51fa0238791f5dc/IKEA-INGATORP-Dining-table

Jan D. (2014). IKEA Malm desk - white working table. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/5d819ba4e21208ac2b5fb024f65f6da/IKEA-Malm-desk-white-working-table

Nina S. (2014). IKEA BESTA BURS Desk white. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/6fe9c27aa9fe6643c1a25f36f85c3fd6/IKEA-BESTA-BURS-Desk-white

Brian M. (2018). Ikea Alex Desk. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/d9d22f26-81ec-4733-9380-427a2cfa1c05/Ikea-Alex-Desk

mado. (2014). IKEA_Alex. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/2437cdb254fc7aaada49e29c548bb1eb/IKEAAlex

alons. (2014). IKEA HOPEN Dresser. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/d81d3191fcbcb111cab5175e3d716816/IKEA-HOPEN-Dresser

hm?. (2014). Ikea Hemnes Dresser. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/9bf9e8d47f7b996263d977b06bbd429d/Ikea-Hemnes-Dresser

Maurice D. (2014). IKEA Expedit bookcase 184 x 184 cm. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/78e03b090b5b8689b382d24930b3a04/IKEA-Expedit-bookcase-184x184-cm

Brian K. (2014). Ikea Lack TV Stand. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/8309ee3f60983735c9fb11be9be46c4a/Ikea-Lack-TV-Stand

Jared N. (2014). Ikea Besta Burs TV Stand. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/f4e007189ec3a071b59df26243593b17/Ikea-Besta-Burs-TV-Stand

hm?. (2014). Ikea Pax Wardrobe. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/92c490a802d0fe463d977b06bbd429d/Ikea-Pax-Wardrobe

Emanue-l. (2014). Ikea Brimnes Wardrobe - 2 Doors. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/bf4ebdda4a95dce3e62e3259d8a92730/Ikea-Brimnes-Wardrobe-2-Doors

Ogolav .., (2014). IKEA Brimnes Wardrobe with 3 doors. 3D Warehouse.
https://3dwarehouse.sketchup.com/model/u761161a0-278e-42e3-8ae3-34d0b347a7c6/IKEA-Brimnes-Wardrobe-with-3-doors

Michael C. (2015). MALM King Size Bed Frame (Black Brown). 3D Warehouse. https://3dwarehouse.sketchup.com/model/uff3462ae-1312-47e1-af2c-35f1fe94b760/MALM-King-Size-Bed-Frame-Black-Brown

dillbilly. (2014). IKEA Malm Bed- Queen. 3D Warehouse. https://3dwarehouse.sketchup.com/model/bd930eb22e73ff7d609662108230548e/IKEA-Malm-Bed-Queen

Tails. (2014). Ikea Malm Bed Single. 3D Warehouse. https://3dwarehouse.sketchup.com/model/3768c89ac578b7bd290dcdea57874bf/Ikea-MALM-Bed-Single

nicolo. (2014). IKEA FAGELBO: Canape d'Angle convertible. 3D Warehouse. https://3dwarehouse.sketchup.com/model/d3db0f5bf70b9e32f7823f022ac104a2/IKEA-FAGELBO-Canap%C3%A9-dAngle-convertible

alons. (2014). IKEA EKTROP Sofa. 3D Warehouse. https://3dwarehouse.sketchup.com/model/fc553c244c85909c8ed898bae5915f59/IKEA-EKTROP-Sofa

mado. (2014). IKEA_Kramfors_Three-seat sofa. 3D Warehouse. https://3dwarehouse.sketchup.com/model/c3e248c5f88e72e9ea65c47b660136e7/IKEAKramforsThree-seat-sofa