CS 5630 / 6630 Project Peer Feedback

Basic Info

The project title:
Post-processing Image Statistics for Bidirectional Path Tracing

Git repository:
https://github.com/lediaev/vis2016

Laura Lediaev
lediaev@sci.utah.edu
U0863084

Ouermi Timbwaoga Aime Judicael
touermi@cs.utah.edu
U1078389

Hirad Sabaghian
hirad.sabaghian@gmail.com
U0930231

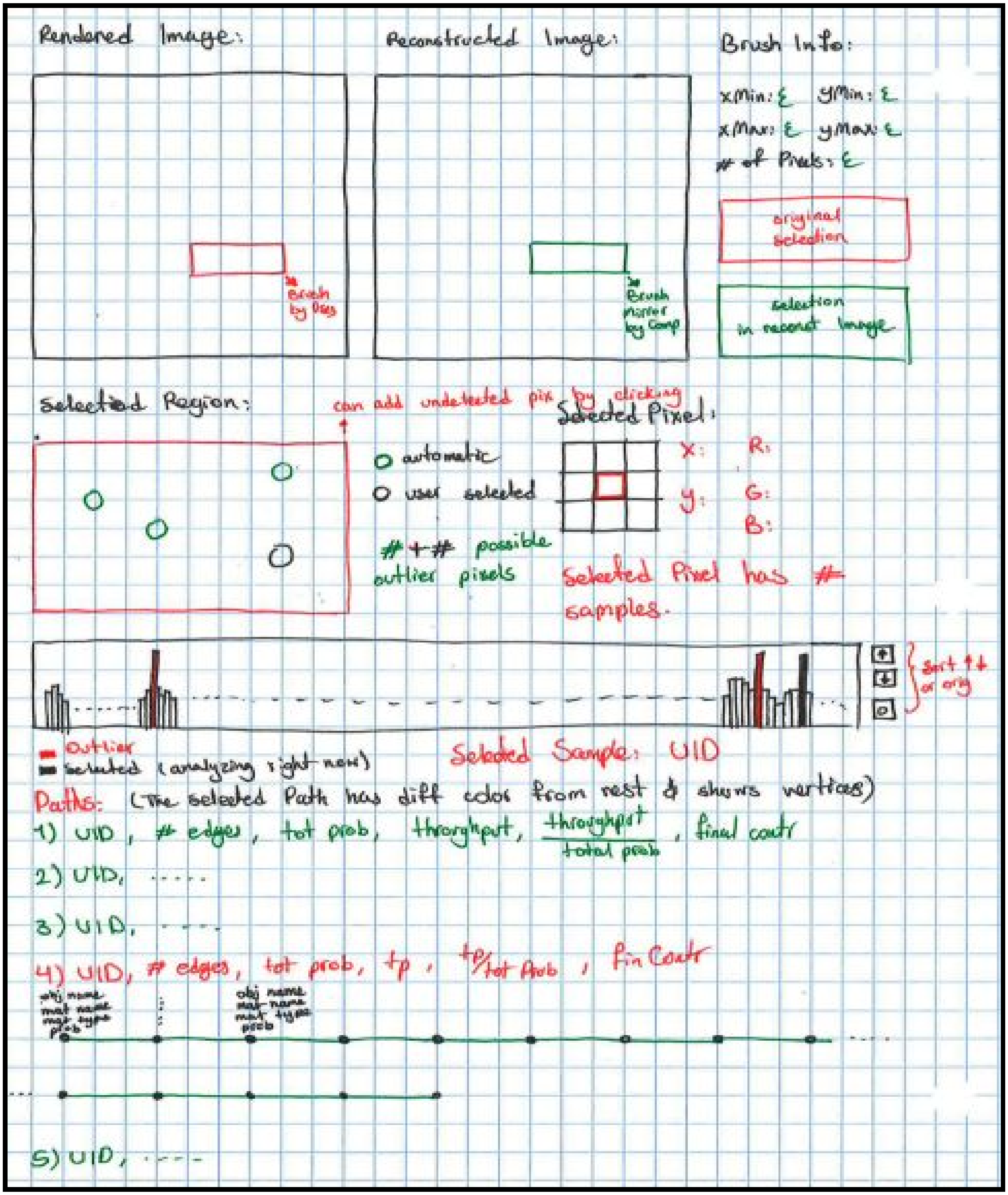

Figure 1: This is the final design we presented at the peer feedback session

Peer Feedback

We found the project peer feedback to be very valuable for our project.The feedback received from x, y (we have to get their names) was very helpful and it help us critically think about our design and how modify it to meet the objectives we defined. They took the time to make sure they understand our project and analyzed the different aspect of our project. Their questions, concerns, and feedback were fair because they were looking to provide us with constructive opinion on our design and project.

The input we got from x and y focused of the different goals we defined and the design of our project. Below is a list of the different feedback we got and how we plan to address them.

1. **Why not combine "Original Image" view with the "Magnified" versions showcasing the brush selection this will reduce redundancy in the design?**

We agreed with this comment. It makes perfect sense because we have same view 3 times in the design and there is not valid reason to have so many. In order to mitigate this issues we have decided to reduce the number of views to 2. The first one being the original image and zoomed in the version of the selected region. Though it seems that there is redundancy, we think it is important for the user to have the zoomed in version. The zoomed in version allows the user to clearly identify the outliers, and the original image give the user a reference at any point of exploration.

2. **Include an indicator to show the association of a bar (sample) and its paths.**

Our design did not clearly indicate the association between a bar and a sample. This can be done by color coding the currently observing sample, and maybe having a small text indicator that "The following paths belong to sample UID"

3. **Is there a limit on how many paths are shown? If so what is it based on? How do we filter and change the criteria of the paths shown?**

These questions indicated the complexity of our project and difficulty to display the information needed. Furthermore, it stresses on the incapability of the browser to handle very large data sets. After the feedback session we discussed about clear targets for the project that are reasonable and not very complex. We decided a user can filter down to the paths of a specific sample but the user can only remove samples. The result image will update base on removed sample. We'll also display the changes made so the user can clearly keep track of the changes they made. In addition we decided to limit the sampling size to a low number in order to be able to make our queries in a timely fashion.

4. **Can we eliminate individual samples?**

It was not clear in our presentation and design what the resulting image was made of. Our peers were confused about how/if eliminate a sample. In order to make clear that user can eliminate a sample, we include and info box next to resulting image to show the samples that have been removed. This will show 0 when no sample has been removed.

5. **Should we change the X,Y,R,G,B indicators for the 9-pixels view to show on hover?**

Our peers recommended that we change the X, Y, R, G, B info to show up on hover. They think would be nicer. We think that if it is not cluttered it is also nice to have the information displayed. Furthermore, the tooltip may hide information we want to see since it will be displayed of neighboring pixels.

6. **How to indicate client-server interactions? E.g. queries sent and logs received? We mentioned maybe having a small console?**

This was more of an implementation concern because our project requires a lot of data transfer. In order to resolve this issue we decided to reduce the amount of sample to a minimum.

7. **How to indicate changes for the reconstructed image? Maybe use the console again or a list of all the modifications so far?**

Our design did not clearly show how a user can keep track of the changes made. We mitigate this problem by providing and box displayed near the resulting image which contains the information about the different changes that has been made to the original image.

8. **How to indicate possible interactions with the views? For example how to inform users that they can reconstruct an image based on a specific path-length? What about other operations?**

For sorting the samples, we use some glyphs that indicate the order. We aim do the same, and maybe even design our own glyphs for other interactions. An other idea is to indicate these relationship through textual elements. E.g. “Click this to do X”