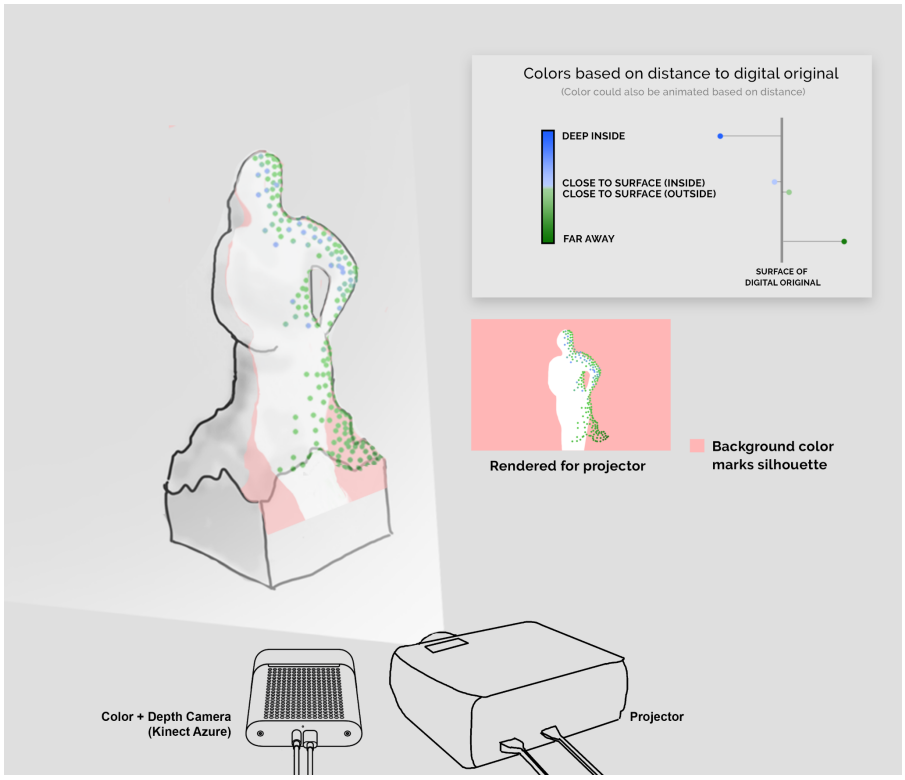
Usage Documentation for sculpting with SARA using Unity

(Spatial Augmented Reality Assistance)

In combination with a Kinect Azure, a video projector, this solution can visually guide a sculpting artist in the work of manual transfer of digital to physical three-dimensional form. It can be used for prototyping or as an alternative to machine-controlled 3D milling and printing.



Usage preparation

1. Connect a projector as second display (use extend, not duplicate)
2. Make sure it is sharp at the distance you intent to sculpt
3. Connect the Kinect via usb (Kinect might also need power cable to function)
4. Make sure the kinect is working, by running Azure Kinect Viewer (part of Kinect SDK)
5. Open the Unity project with the scene: **SARA.unity**
6. Make sure there is only one game-tab, and move it to the projector screen and maximize.
7. Import your 3D model to the Unity project (make sure it has the **correct scale**)
8. Place the model inside the SARA_Main/UserScene/Rotate/User Model

Calibration

Basic settings and calibration in the unity scene is using the GameObject SARA_Control.
In order for SARA to be able to estimate the relative transform between the projector and the Kinect, you need to run a sequence on the projector (twice, with different depth), during the calibration, the main scene (SARA_Main) is turned off automatically.

1. Go into play mode
2. Turn on “SARA_Control/Debug Kinect Stream”
3. Adjust “SARA_RenderKinectCloud” transform so it aligns ok with your model (Use Unity scene view).
4. Place a piece of large cardboard in front of the cameras so that the entire projection falls on it, the entire projection should also be visible from the Kinect camera (RGB and depth). It should also be as close as possible to your sculpting spot
5. Click “SARA_Control/Start Calibration”
6. Wait until you can see a patchy red/yellow/orange gradient on a box in the scene view (the colors represent the coordinates of projector pixels)
7. Move the cardboard so that it is more than 10cm from previous position (in depth) (again, entire projection should fall on it, and it should be visible from the kinect)
8. Click “SARA_Control/Complete Calibration”
9. When SARA_Main is turned on again, calibration is complete, you can view the calculated rays by enabling “SARA_Control/Show Calibration”

10. (Start Sculpting!)

How does the calibration work?

Using coded/structured light, we can get projector-pixel-coordinates associated with kinect-RGB-pixel-coordinates. And using the kinect SDK, we can associate kinect-RGB-pixel-coordinates with 3D coordinates.

By doing 2 passes of coded light with different depth, we get **two** 3D coordinates for each projector-pixel-coordinate. By forming rays from these 3D-coordinate-pairs, we can try to find the single point that is closest to all rays, and this position gives us the position of the projector (in the kinect coordinate space). We can also use the rays (or some of the rays), to estimate the direction of the projector.

We can now create a virtual (Unity) camera using this position and direction, with an approximate field of view.

The last step is creating a 2D displacement map that will “correct” the virtual camera output so that it aligns with projection of the rays. Since we use warping/displacement, we don’t have to care about estimating/modelling the exact field of view, roll, tilt-shift, optic distortion etc.

Known issues

- Calibration can not be repeated within the same session (need to stop and play again)
- Coded Light sometimes fails. (if it happens during the first pass, you can see black bands appear in the patchy red/yellow/orange gradient at step 6)