iPad Implementation of “Structured Annotations for 2D-to-3D Modeling”

Yotam Gingold, William Brown

Missive: The goal of this project is to create an iPad implementation of “Structured Annotations for 2D-to-3D Modeling.” The project was originally written in a mix of Python and C/C++ (mostly Python), so the Python parts will have to be rewritten in Objective-C. Moreover, the GUI was designed to behave like a desktop drawing app (e.g. Illustrator); iPad conventions differ. The visual style of the app can remain the same.

Schedule: This document is written Thursday, January 19th; the last day of classes is Monday, April 30th. There are 14 weeks beginning this coming Monday, January 23rd. Biweekly goals:

Monday, Jan 23rd: Will will have thought about 1 and we will discuss it together
Thursday, Feb 9th: 1 and 2
Thursday, Feb 23th: 3 and 4a
Thursday, March 8th: 4b,c,f and 6a
Thursday, March 22th: 6b
Thursday, April 5th: 5
Thursday, April 19th: 4d,e,g
Thursday, May 3rd: 7 and 8

Milestones:

1. Designing the layout (pencil and paper/wireframes): The application was originally designed as a desktop GUI. We should study interactive iPad editing applications like Apple’s Keynote, Autodesk SketchPad, ColorSplash, Autodesk 123D.
2. Getting a basic iPad app running with the desired layout: Using XCode and InterfaceBuilder to design the layout and controller classes for the project. This includes panning and zooming the canvas and displaying an image in the background.
3. Placing shapes: On an iPad, drawing the silhouette of an ellipse and the spine of a generalized remains a good mechanism. If Yotam wants to use a radically different implementation of generalized cylinder fitting from the paper, he should figure that out himself.
    a. Ellipses
    b. Generalized cylinders
4. Manipulating shapes: These may remain unchanged from Yotam’s implementation; Will can adapt or improve them to be more iPad-like as he sees fit.
    a. Translation, rotation, uniform xy scale
    b. Out-of-image-plane tilt
    c. Cross-section scale
    d. Spine manipulation
    e. End-cap handles
    f. Symmetry sheet handle*
    g. Cross-section shape adjustment*
5. Placing annotations: These may remain unchanged from Yotam’s implementation; Will can adapt or improve them to be more iPad-like as he sees fit.
    a. Connection curves
    b. Mirror
    c. Alignment

d. Same-length
e. Same-tilt
f. Same-scale

6. Generating 3d models: A fancy mesh representation may not be necessary; normals for the ellipse and the generalized cylinders can in general be stated simply and analytically. For pinch-free shapes or end-caps, though normals may have to be computed by averaging the one-ring.
    a. Basic
    b. Pinch-free
7. Bonus:
    a. 3D manipulation for connection annotations: Connection annotations can be more intuitively modified by dragging the shapes in the 3D view.
    b. Registering the app with the operating system as a target for sending images: There is way to accept image data (as a local URL?); I believe there is a library for this that many iOS image editing apps use.
    c. Export of 3D models: export to OBJ and optionally to a fancier format or to one of the 3D printer warehouse sites.
8. Documentation: A high level document describing the control flow of the application.

Deliverables: git repository, including documentation, code, and XCode project that can be used to compile a working iPad executable. Every “deadline” on the schedule we will meet and review the progress.

Evaluation: Evaluation will be based on meeting the deadlines. If a milestone proves substantially more difficult than expected, then this document will be amended. Grade will be reduced if non-bonus milestones are never accomplished, unless coordinated with Yotam due to unforeseen difficulties.