Art and Machine Learning | Prof. Eunsu Kang & Prof. Barnabas Pozcos | TA: Jonathan Dinu | Carnegie Mellon Univ. 2018 Spring

Robot-RNN Doodle by Ardavan Bidgoli

This project explores the possibilities of integrating the RNN-Sketch Demo[1] with a robotic arm to form a collaborative workflow between user, RNN, and robot. The motivation behind this project is to find a robot-human interaction scenario to open the discussion for further development in Robot-Art competition.

Sketch-RNN is trained on a data set of millions of doodles collected through Quick Draw project. It leverages LSTM architecture to generate svg doodles based on a given svg stroke as the initial seed.

Sketch-RNN comes with hundred of trained models, ranging from trivial doodles of animals and object to complex combination of them. For this project I used the Flamingo model to generate flamingo doodles.

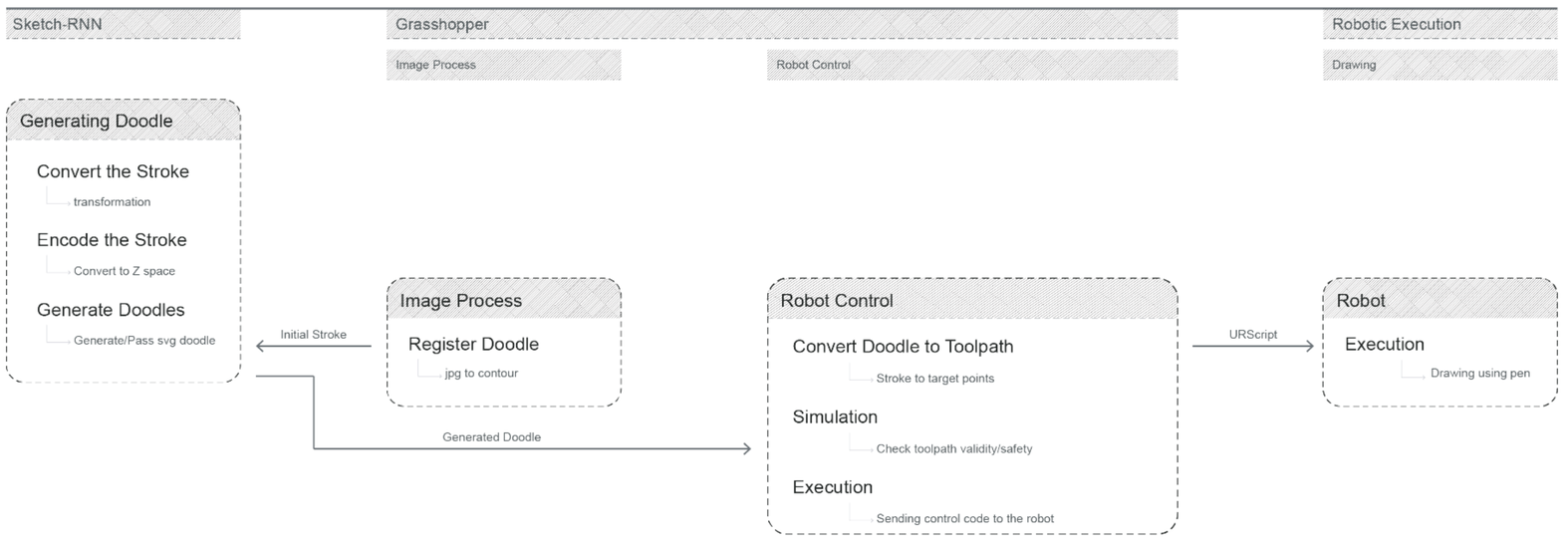
1- Ha, David, and Douglas Eck. "A neural representation of sketch drawings." arXiv preprint arXiv:1704.03477 (2017).

System Architecture

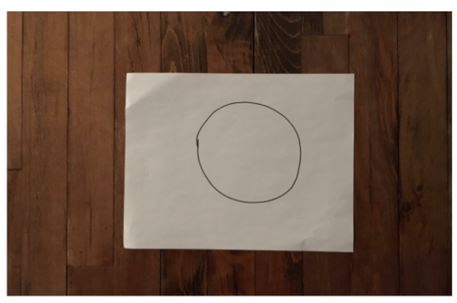
The model consists of two main component, one is the grasshopper definition and embedded Python scripts that handles image processing, toolpath generation, simulation, and communication with the robot.

HAL add-on for Grasshopper is the core engine behind the robotic control and simulation.

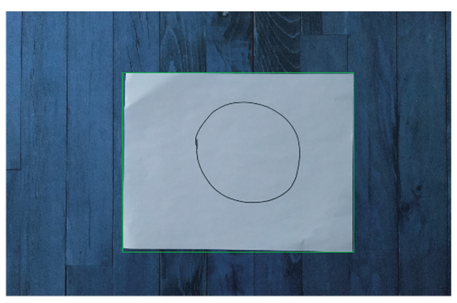
The other component is Sketch_RNN code that generates the doodle based on the user drawn seed. It is using Magenta library.



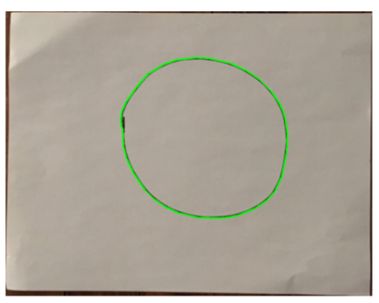
User Interaction



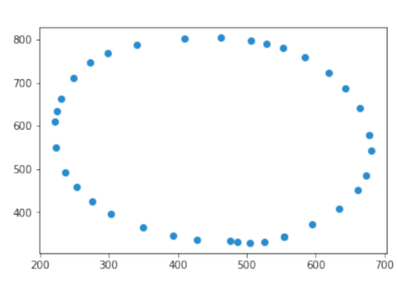
1.
User initiates the process by drawing a stroke on the canvas;



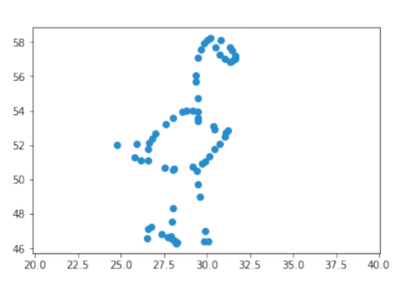
2.
Using OpenCV, cnvas boundaries will be detected and perspective will be corrected;



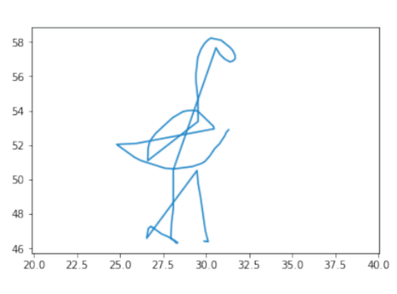
3.
User's drawing will be detected and converted into a vectroized bounday;



4.
Vertices of the boundary will be passed into the Sketch-RNN.;

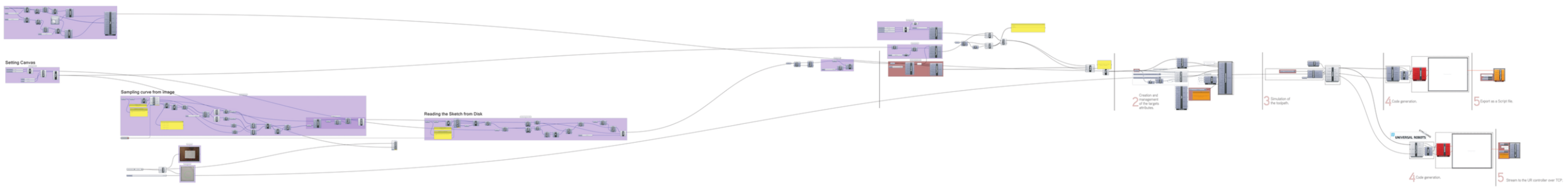


5.
The stroke will be fed into the RNN and to generate the doodle.



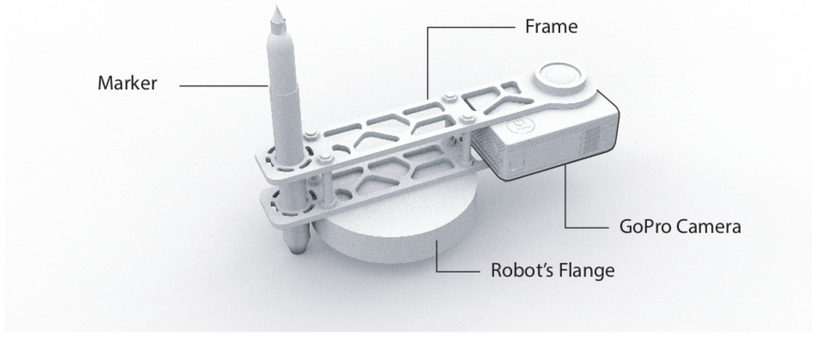
6.
The stroke will be sent back to Grasshopper to generate the toolpath.

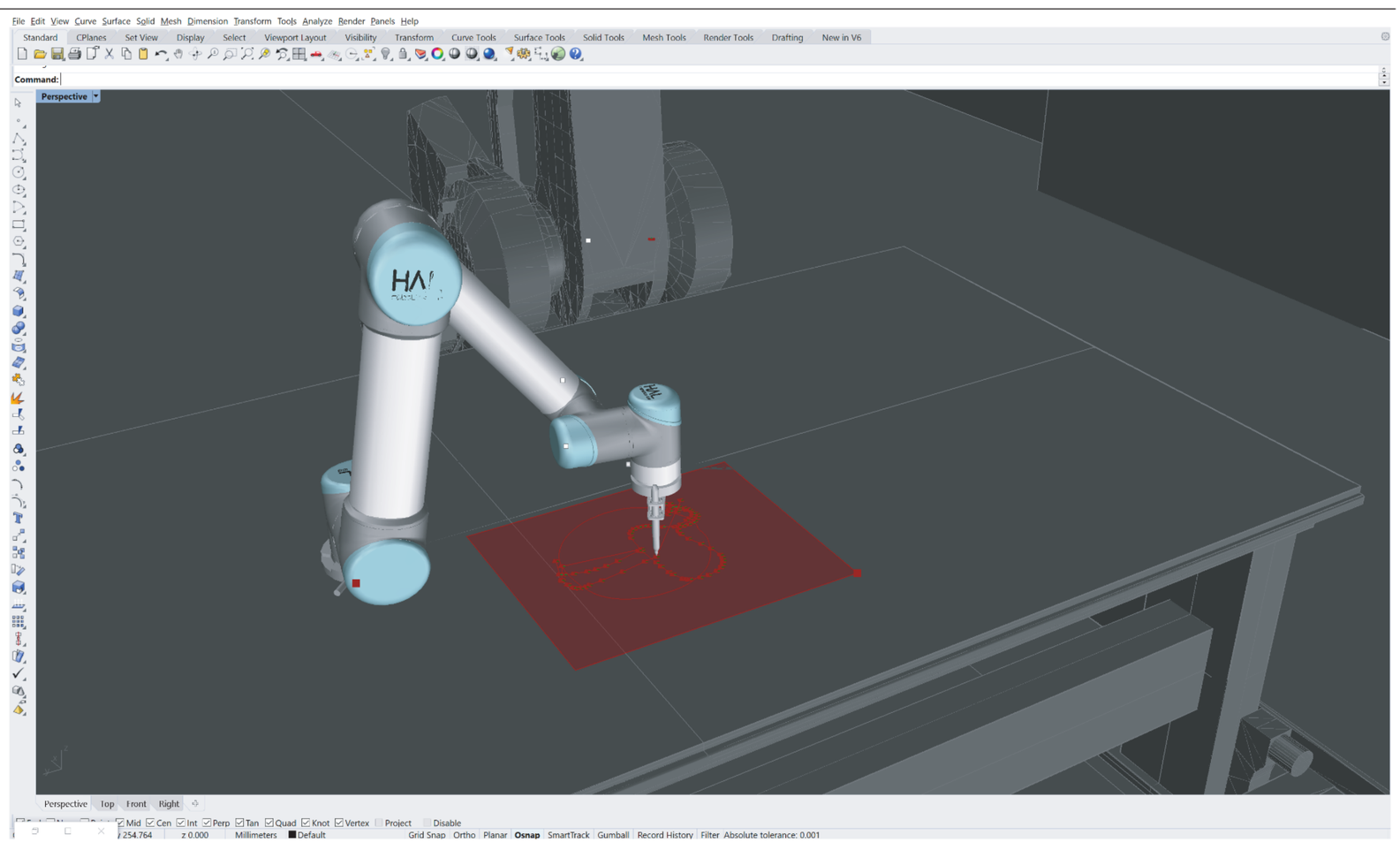
Grasshopper Definition / Python Scripts



Robot Setup | Image Processing | Converting doodle into toolpath | Post Process and Simulation | Code Generation and Communication

Tool and Simulation





Next Steps:

- Implementing the model on the physical robot and perform the real-time interaction;
- Improving the rnn performance to generate more accurated doodles.
- Improving the accuracy of mapping method to have the stroke and sketch matching each other on the canvas.