

# Interactive Random Art Algorithm

Student name – Rishi Sankhla  
Supervisor Name - Andy Lomas

## Abstract

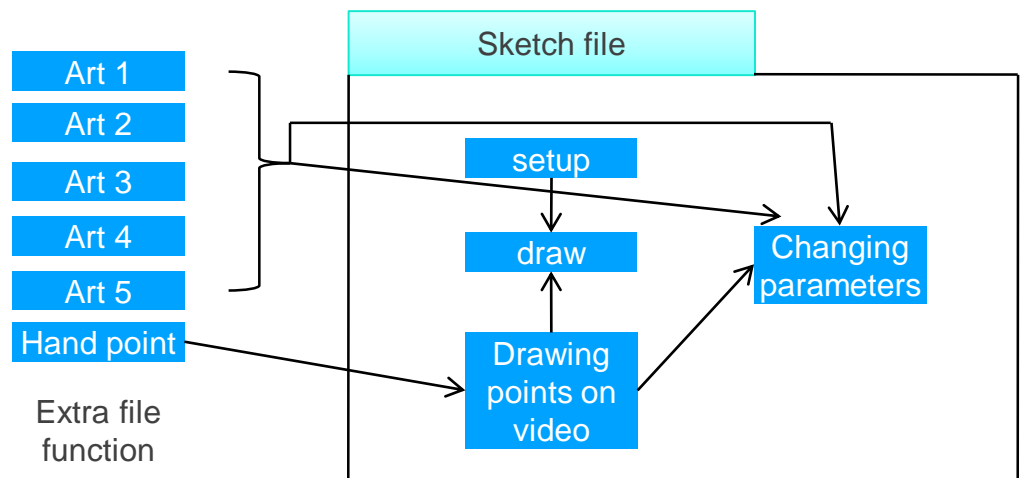
Through this project user can interact with the pre-generated 2D art forms and can save their state using the hand movements. To run this application all we need is a computer with a webcam, as it's programmed in p5.js. With the help of this platform artists can play with other artist drawings.



Working screenshot

## Introduction & Background

As there are many drawing application in the market these days, but there are very few apps where user can interact with the art, so this new concept could be useful for beginner artists to play with other art forms and can save the state of their drawings on the top of other artist drawings. So to implement this project there are two major type of algorithms, i.e. random art generating algorithm and hand detection algorithm.



Interaction of files and functions

## Specification & Implementation

To implement the working of this project I have used object oriented methodology, as we can see from above diagram I have written five random art generating algorithms and one hand point detection algo, which means creating their individual object, then calling them inside the sketch file accordingly.

Another type of methodology I have used here is experimental so that I increase the functionality of the project iteratively based on the testing. Therefore the development process started from writing the requirements, then building design for it, after design I wrote the code for it, then after performing the testing, I wrote the code for new functionality.

## Testing & Evaluation

I was able to implement 5 random art generating algorithm, I have also implemented the interaction system using hand detection algorithm. And in the last I had implemented save state system too, through which user can save their drawings.

User average numbers:

- usefulness 6.5
- required improvement 4.6
- easy to use 8.2
- overall software speed 5.9
- Average time spending of user 17 mins
- overall software rating 7.1

After testing also realised that multiple classes and objects had reduced the frame rate, especially when we are running video in p5.js

## Conclusions & Future Work

In conclusion MVP is working and user enjoyed the overall experience, but because of iterative objects, it takes a lot of time to perform all the computation and making the computer slower, hence it is taking extra time to reload the application. In future we can add one more feature through which user can also upload their new algorithm and can configure it with our hand detection algorithm.