# DIP Project Proposal

## ShadowDraw: Real-Time User Guidance for Freehand Drawing ( Project ID: 29 )

Github link

**Team Members: Ishan Bansal (201530193), Swapnil Daga (201531063)**

**Main goal(s) of the project**

The major aim of the project as presented in the attached paper is to create a real-time system that is able to guide the drawing of objects. As the user draws an image, ShadowDraw continuously keeps on updating the shadow image underlying the user's strokes. The shadow so formed is not made up of a single image of the dataset but is the combination of the various relevant images present in the database, and the contribution of each of the image is determined by its similarity with the user's sketch. Hashing techniques are used for both local and global similarity for improving performance.

The major goal which we are aiming in our project is to display the most relevant images from the database given the user's image sketch. Thus, given an input image (which is the drawing made by the user) we will display the images from the datasets claiming those to be the possible drawing that the user might want to draw. The method which we use for this purpose will be inspired from the "ShadowDraw: Real-Time User Guidance for Freehand Drawing" paper.

**Problem definition (What is the problem? How things will be done ?)**

**Problem:** Since a large portion of the population does not have a good hand in drawing images, as a result, the images drawn by peoples are not always close to the actual image which they want to form. Therefore our intention is to give a suggestion of possible images from what the user has been already sketching so that the user can refer to that image and continue his drawing. This will improve the quality of the drawing since drawing image observing the image of the final object itself is much easier as compared to drawing image from our own imagination. Therefore given a dataset of images and the input image by user, our task is to find the images from the dataset which maximum resembles the final image which the user wants to form.The paper shows improvement in drawing of individuals like this:

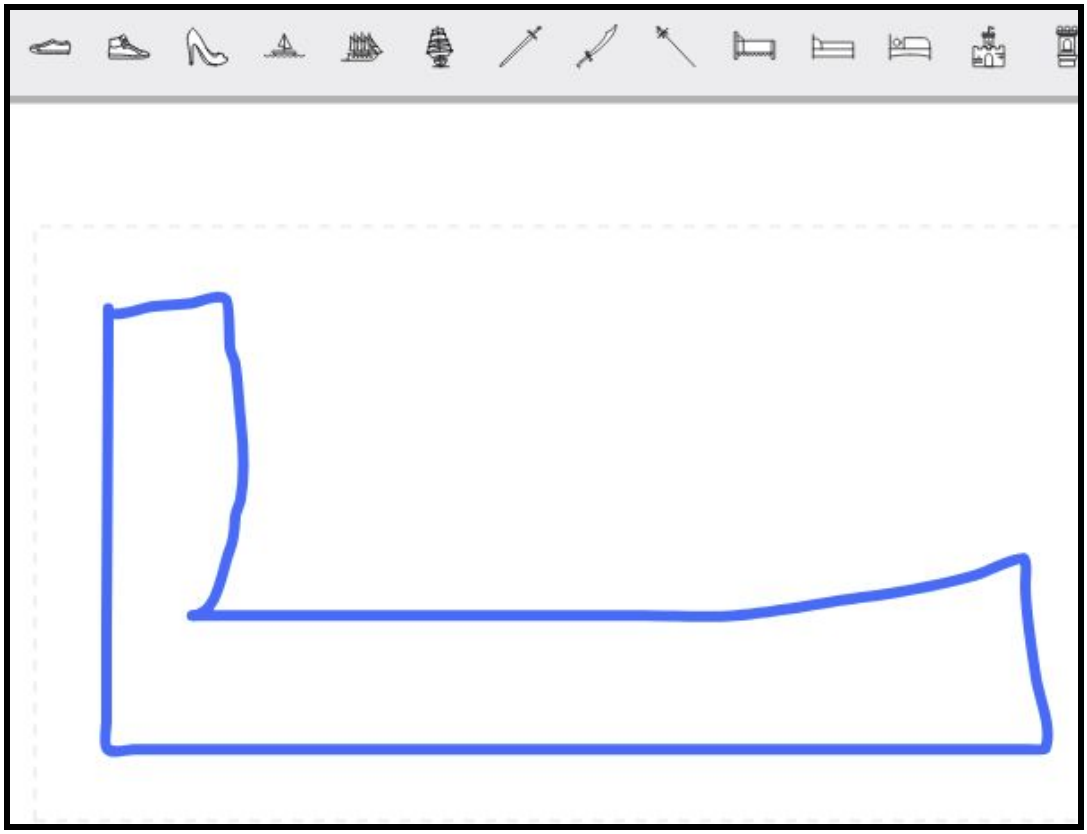

**Solution :**

The major tasks involved in this process are:

1. The construction of an inverted file structure that indexes a database of images and their edge maps.
    a. Since we are taking the natural image, therefore, a number of edges are there which are in the background and are not important since the user is not going to draw them, therefore, we use the long edge detector for this purpose.
    b. The encoding of these images is done depending on the histogram of edge positions. For this, the use of the hashmap is been done to store these values.
2. A query method that, given user's, strokes retrieve matching images aligns them to the evolving drawing and weights them based on a matching score.
    a. The user image is also encoded in the same manner as the images in the dataset over the patches and is been compared with the inverse map formed of the images in our datasets.
    b. Depending on the comparison with the images of the datasets, weights are been assigned to each of the images.
3. The user interface displays top relevant images from the database as the user draws.
    a. The images which get the maximum weight are the one which are taken and displayed for the user.

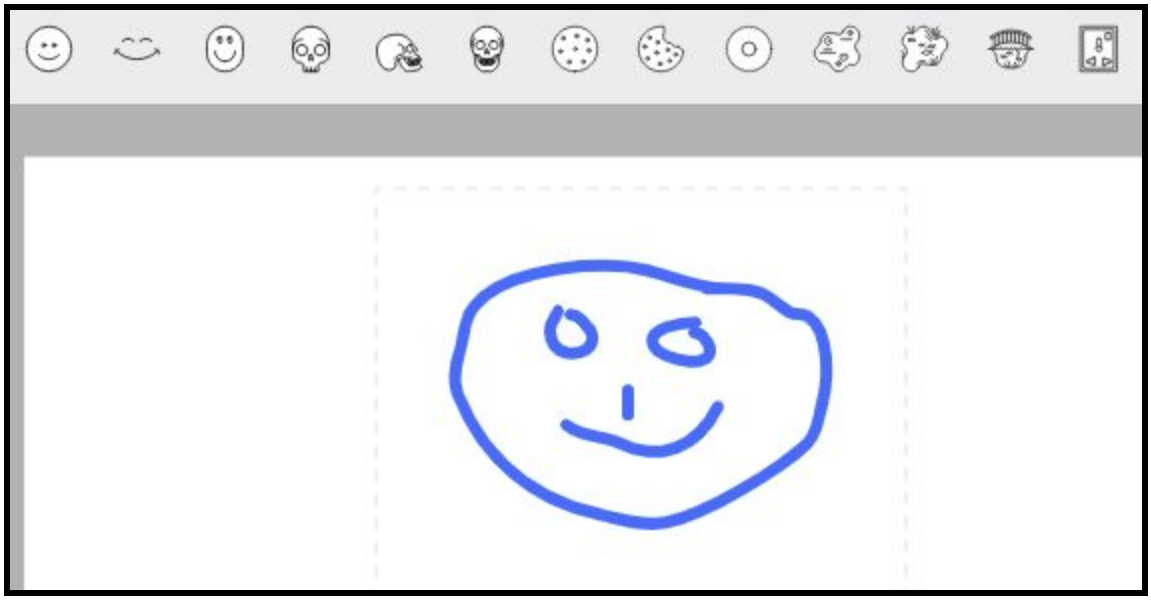
**Results of the project (What will be done? What is the expected final result ?)**

On giving an input image, our code will display the most appropriate choices from the dataset of the objects that the user might want to form.
Our objective would be to give suggestion somewhat like below:



The below one is a drawing and above ones are the suggestions. ***The exact format of representation may differ but this is a general idea.***

Another example would be as follows:



**Team members and tasks for each member (What will each team member do?)**

**Tasks:**
***Form the dataset: (Ishan)***
1. Finding the images .
2. Take the edges and extract the one which is most appropriate
3. Encode the images and put them into the hashmap

***Comparing Image: (Swapnil)***
1. Clear the strokes to form appropriate edges
2. Compare the images from the hashmap with the user image and assign the corresponding weights.
3. Take the most relevant ones and display it on the screen.

The above distribution of work is not fixed and may change.

**What are the project milestones and expected timeline?**

| Task | Timespan |
|---|---|
| Finding the images | 1-7 Oct |
| Taking the edges which are most appropriate | 8 Oct-12 Oct |
| Mid exams | 12 Oct -16 Oct |
| Encode the edges/putting into the hashmap | 17 Oct -23 Oct |
| Buffer period/Debugging period | 24 Oct -26 Oct |
| Clear the strokes from the input image | 27 Oct - 2 Nov |
| Compare the image from the hashmap/assign weight | 2 Nov -13 Nov |
| Extracting most relevant images and displaying it on screen | 13 Nov -15 Nov |
| Buffer/Debugging period | 15 Nov- Presentation |