Samsung Flip - Accidental Erasing Problem

Sarthak Garg   Varun Gupta

Overview

The Samsung Flip is an innovative digital flipchart that helps to drive more productive and efficient business collaboration without the hassle. With the Samsung Flip interactive UHD display, your team can work smarter, faster and better. Business meetings can take place anywhere, anytime, and you can take the stress out of the logistics.

The Flip is an easy to use device which feels just like a writing board and it uses a passive pen technology which means that you can easily write using any passive object pen , marker , etc. It also detects the thickness of the object and on that basis it works in 3 modes pen , highlighter and eraser. It also allows simultaneous writing and erasing.

.
Problem Statement

There is the functionality of erasing using the palm. This functionality is provided as it make erasing stuff easy and the user feels just like he is working on a real writing board. But the problem with this functionality is that sometimes when the user doesn't really want to erase but due to unintended palm touches on the screen while writing, the devices senses it as an erase event and erases the part written on the board.

Aim

The aim of this project is to make the device capable of differentiating between intended and unintended touches so that the user gets a better experience.

Approach

1. In the Flip, there are events for different kinds of cases. When you place a pen or eraser on the display, it is a pen down event and various values such as x coordinate, y coordinate and radii are given. If the pen isn't lifted, then it is in writing operation and during this time no information is there about radii (size). If there are multiple events, then they are assigned different IDs

2. The first approach that we thought of was to apply Machine Learning and to make a model which will consist of the following features:
    - Relative velocity of eraser with respect to pen

- Size of the eraser
- Shape of the patch
- Distance of the palm with respect to the pen

3. The problem with the above approach was that we had to reject the eraser as soon as the eraser down event is detected and so we won't really be able to get the relative velocity of the eraser wrt to pen as our code need to classify the eraser as intended or unintended as soon as it's down event is triggered.

4. Another problem that we faced was that there was no way of classifying the shape of the patch as the device only gave radius_x radius_y and radius as the output so we couldn't use that feature either.

5. So now as two features out of the four were crossed out we changed our approach from machine learning to using simple bounds method in which we tried to determine the bounds on size of erase and distance of the palm with respect to the pen.

6. We made a python script that extracts parameters including :
   - distance of erase pen down event from the just previous pen writing event.
   - Whether the pen writing event has a y coordinate greater than the erase pen down event (this is an assumption that hand touching would happen below the pen touch, which is generally the case)
   - The radii values of the erase pen down event .
7. We tried to apply machine learning to get the best parameters for logistic regression which classifies as intended erase and unintended erase. However, we only had a data of around 200 entries and so accuracy on test set was low. So we decided to assume the probability of accidental erase to be standard normal distributed with respect to the various parameters and found mean and standard deviation of the parameters and then found out bounds that gave reasonably high probability.
8.
9. If the bounds are satisfied, then it is accidental erase and that entire erase pen down event is rejected. So even if there is subsequent accidental erasing after that, it will be rejected. This improves the performance.

Bounds

x_diff: Difference in x coordinate of eraser and simultaneous pen events

y_diff: Difference in y coordinate of eraser and simultaneous pen events

Radius: Value of the radius parameter for that particular erase touch down event

sq_diff: Square difference between erase and pen event.

TEST1-

Do not erase when x_diff<190 && y_diff<195 && radius<48 && sq_dis< 370

TEST2-

Do not erase when x_diff<225 && y_dif<270 && r<64 && sq_dis<370

Results

The results that we got from samsung after testing our approach was that our simple bound method was able to classify and reject most of the common cases and worked well, thus improving the user experience. But there were some corner cases that still need to be taken care of.