

Mimic Game

This project allows up to get familiar with Afectiva API by developing basic feature such as displaying feature points used to analyze expressions and show dominar emoji that summarize the most probably emotion of that person.

Display Feature Points

The first function that we have to complete is display features points

It received 3 variables: canvas, img and face.

- 1) canvas: is our container to draw. I choose yellow as the color for the points.
- 2) img: is the image itself passed to the function. in this case we are not doing anything.
- 3) face: is the object that contains all the results of the Afectiva API process, including the feature point locations for the given image and the relevant values such as the dominant emoji.

I loop over those points and using the canvas context draw them using the feature point x and y.

Show Dominant Emoji

This function has the same signature.

In this case i use the font Verdana size 48px to render the emoji to the screen.

I choose one particular point (index 10). I render the emoji as text on that particular x, y coordinate.

The Game

Global Variables

I used updates, currentEmoji, score and total;

- updates: count the numbers of times the function it's been called.
- currentEmoji: store the emoji displayed.
- score: number of successful matches.
- total: total of emojis displayed.

game

This is the main function

- Check if they are matching
- If the function loop over 100 times, I move on to the next emoji.
- update the score (even if no changes has happened)
- If the total emojis displayed are 10, I stop the game.

This function it's been called repeatedly by the event "onImageResultsSuccess".

I can probably use some timer instead of number of updates, but I think in this case seems to be working OK as there is a relationship between number of updates and time (refresh rate).

nextEmoji

Just randomly select another emoji form the list and update the total by one.

init

Reset score, total and number of updates. Display the new reseted score and called the nextEmoji function.