Trademark Analysis & Identification Tool (TRAIT)

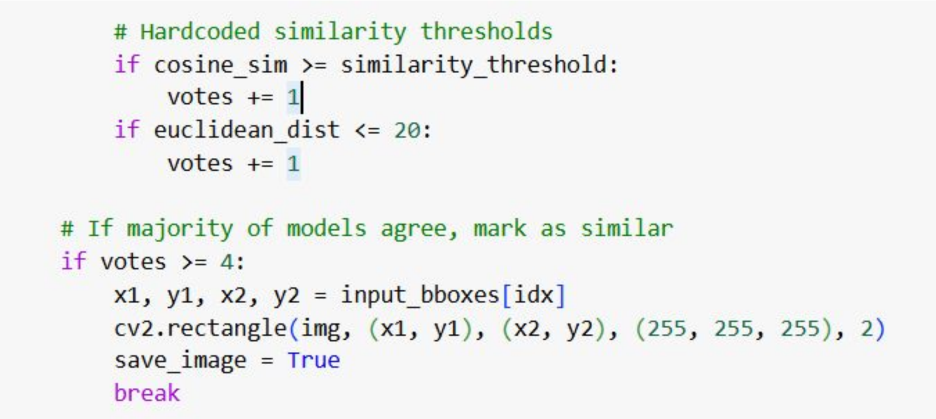
Team 11 Sprint 6 Lane Keck, Logan Taggart, Caleb Stewart

New voting system to help compare logos!

We will use all three embedding algorithms we found: BEiT, CLIP and Resnet.

We will use cosine similarity and euclidean distance.

If the images pass either of these comparison methods, they will receive a vote.



```
    # Hardcoded similarity thresholds
    if cosine_sim >= similarity_threshold:
        votes += 1
    if euclidean_dist <= 20:
        votes += 1

# If majority of models agree, mark as similar
if votes >= 4:
    x1, y1, x2, y2 = input_bboxes[idx]
    cv2.rectangle(img, (x1, y1), (x2, y2), (255, 255, 255), 2)
    save_image = True
    break
```

Pros and Cons to new approach

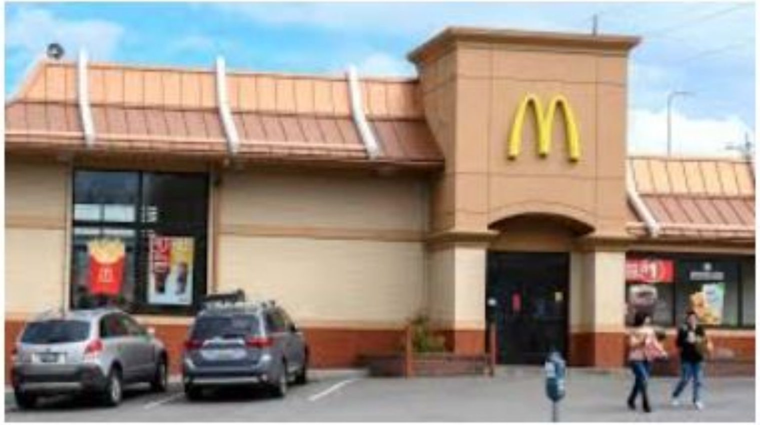
Pros

- Robust
- Reduces false positives

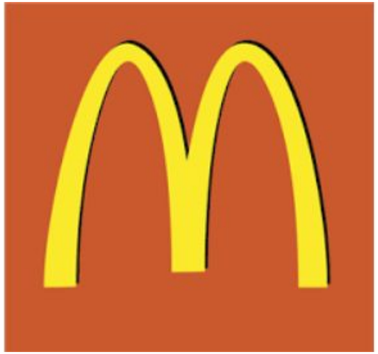
Cons

- We must fine tune a hardcoded threshold
- Could be computationally expensive

Example

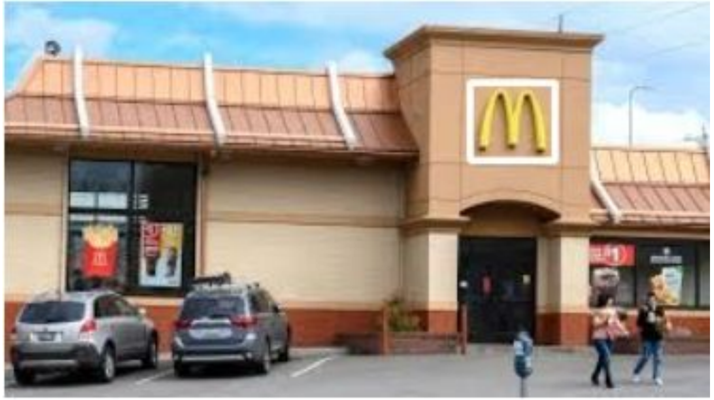


Input Image



Reference Image

Example



Output Image

Logo it cropped from restaurant!

Implementation of New Algorithm

After testing our new algorithm to compare logos with each other, implementation of it within out backend was rather easy.

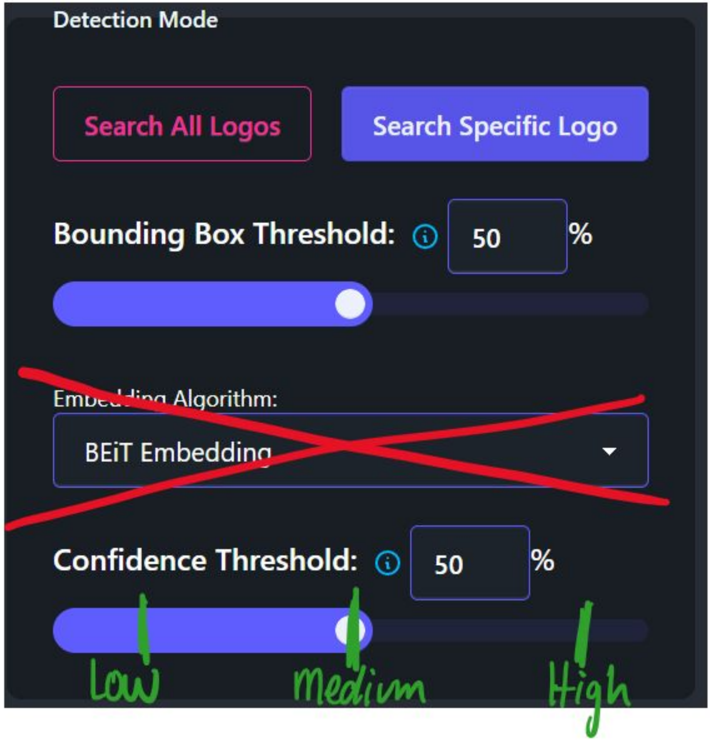
We already had the basic layout from our old algorithm, so adding more checks was straightforward.

Implementation of New Algorithm

Changes to the frontend

- Remove the option to select embedding algorithm
    - No longer needed
- Change the confidence threshold to tick marks
    - This will abstract the number of votes an image will need to pass
    - Ex. Low = 2, Medium = 3, High = 4
    - Or something along the lines of this

This will take away complexity from the user and make it easier for them to use our product



Detection Mode

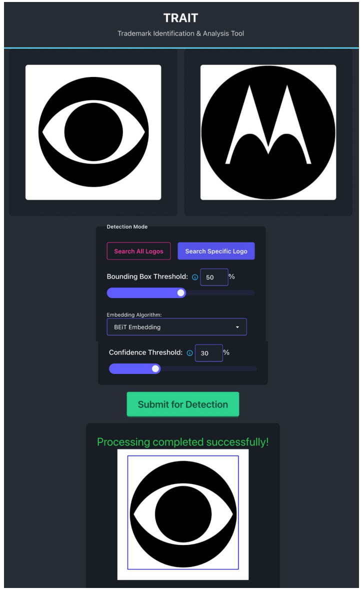
Bounding Box Threshold: 50 %

~~Embedding Algorithm: BEiT Embedding~~

Confidence Threshold: 50 %

Improvement on False Positives

As you can see below with the new algorithm, we have been able to eliminate many of the false positives we were previously dealing with.



TRAIT

Trademark Identification & Analysis Tool

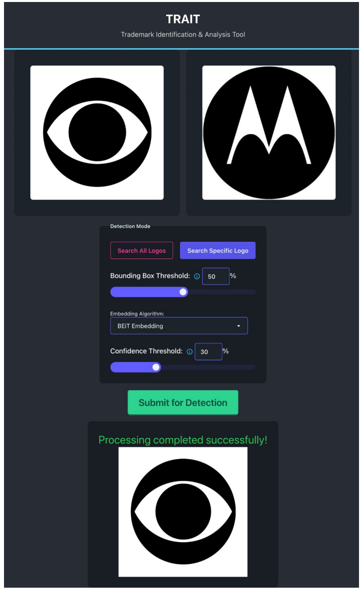
Detection Mode

Bounding Box Threshold: 50 %

Embedding Algorithm: BEiT Embedding

Confidence Threshold: 30 %

Processing completed successfully!



TRAIT

Trademark Identification & Analysis Tool

Detection Mode

Bounding Box Threshold: 50 %

Embedding Algorithm: BEiT Embedding

Confidence Threshold: 30 %

Processing completed successfully!

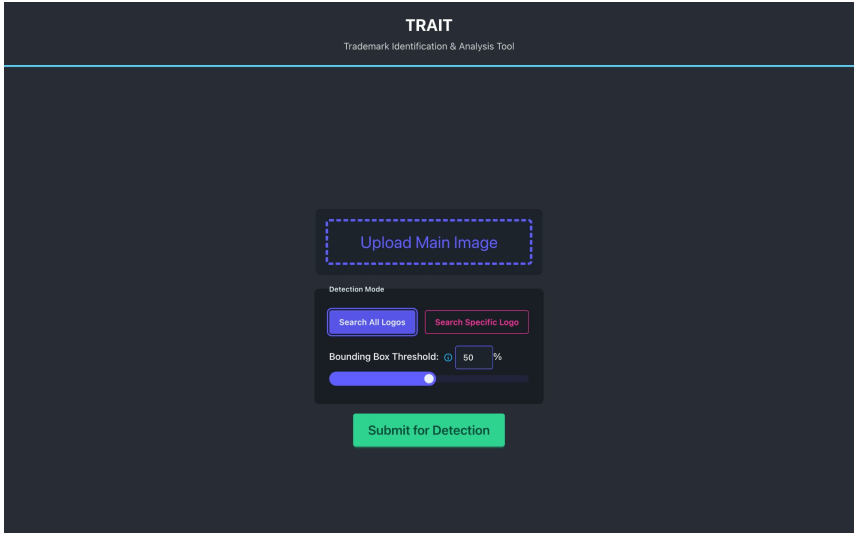
Electron.js

Application has now been converted to work as a desktop application

Since we used Electron.js to wrap the project it appears exactly the same as it did before as it runs using Chromium

When application is launched, it starts the front and back end server, and launches the Electron desktop application

Electron.js

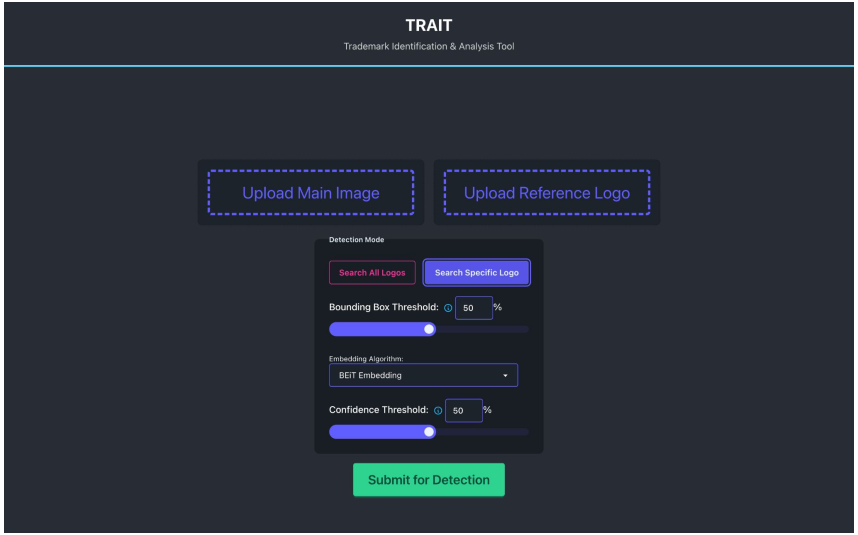


TRAIT

Trademark Identification & Analysis Tool

Detection Mode

Bounding Box Threshold: 50 %



TRAIT

Trademark Identification & Analysis Tool

Detection Mode

Bounding Box Threshold: 50 %

Embedding Algorithm: BEiT Embedding

Confidence Threshold: 50 %

What’s next?

We’ve almost reached our goal of being able to process still images by the end of the quarter.

- This seems very achievable

What is left for this quarter is just a matter of sending the correct statistics from the backend to the frontend and displaying results.