

Trademark Analysis & Identification Tool (TRAIT)

Team 11 Sprint 10 Lane Keck, Logan Taggart, Caleb Stewart

Our Goal For the Quarter

Once again our goal is getting our model to detect logos in videos uploaded by a user.

- Last quarter we got our model to process single frames!

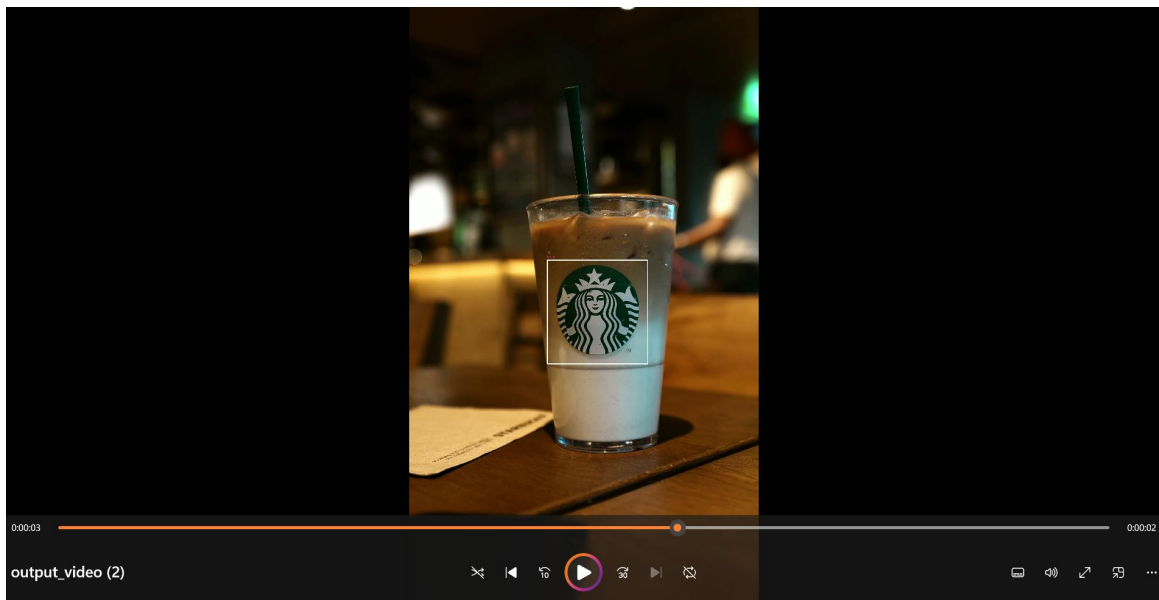
Then we will return metrics such as how many times the model saw the logo, and how long the logo was in the video back to the user.

So this quarter we are focusing on retooling the logic we built last quarter to now work on video footage rather than just singular frames.



Progress So Far

- Our model is accurately detecting logos in every fifth frame!



We chose to analyze every fifth frame due to the long computation times that we found even a short clip/video can take.

Current Problem

Although we are able to detect a logo every 5th frame, our model is not able to classify the logos. This means our model thinks that a logo in separate frames are two completely different logos.



As an example, these images are 5 frames apart. Our model thinks that these Starbucks are two different logos.


Using FAISS - Facebook AI Similarity Search

Since our model only detects logos without classifying them, we need a way to determine whether a logo detected in frame 0 is the same as one detected in frame 5.

To group each unique logo throughout the video and avoid duplicates, we use FAISS.

FAISS acts like a vector database: we represent each logo as a feature vector, and FAISS allows us to query these vectors to find others that are visually similar.

This enables us to consistently track the same logo across multiple frames—even if it appears in different positions, sizes, or lighting conditions.



FAISS Logic

Analyze every 5th frame

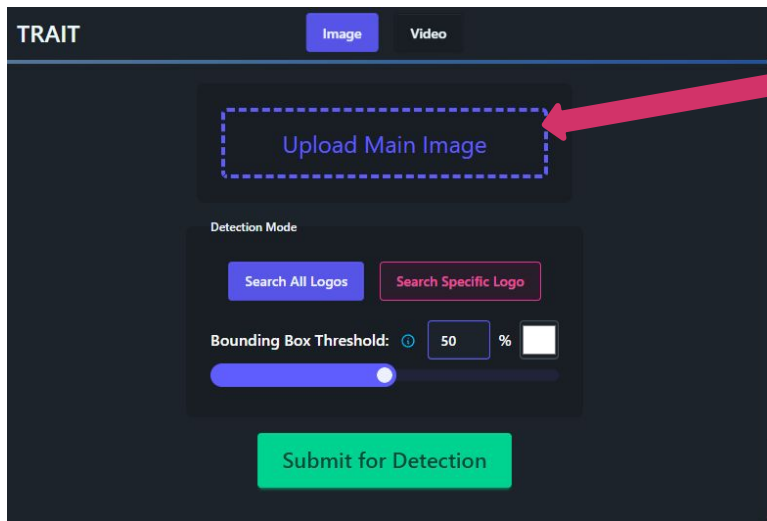
For each detected logo region

- If FAISS is empty
 - Add the logo feature embedding to FAISS
- If FAISS has existing entries
 - Search for a similar embedding (This returns a L2 distance measurement)
 - If a match is found -> Increment that logos screen time
 - If NO match is found -> Add the new logo embedding as a separate entry in FAISS



Frontend Design

We need to update the existing frontend components for the video processing aspect of the project.



The video processing page will look very similar to our image processing page.

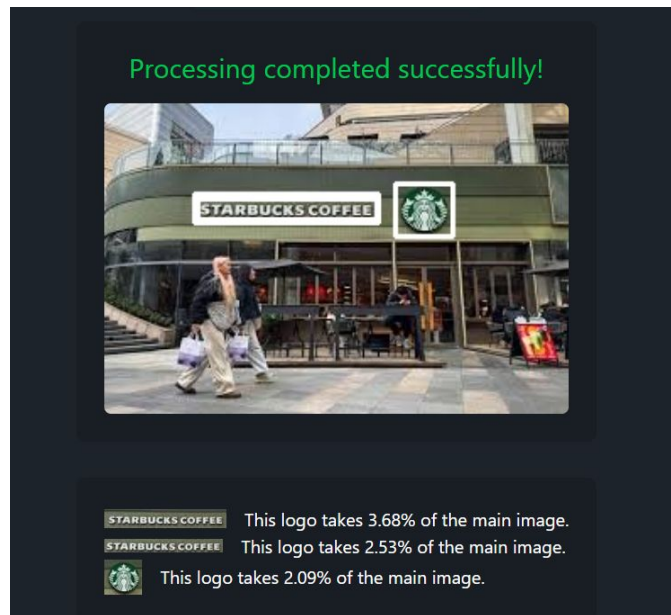
If after further testing the processing continues to be too slow we are considering adding a toggle to give the user the option to process the video at a slower or faster speed, leaving the option to them whether they want to trade off speed or accuracy.

What metrics will the video processing return?

Like our image processing, the video processing will return cutouts of each logo that was found within the video.

But for videos, it will also be returning the combined amount of time that each logo was seen for. This can look something like the amount of seconds, or alternatively Logo A appeared in 10 frames out of 250 frames. Logo A appeared in $(10/250)\%$ of the video.

We could also provide users with the option to either download the processed video with bounding boxes overlaid and/or let them view it directly within our application.



Challenges

- Even when we call the model on every fifth frame, it takes around a minute to complete a 10 second video.
- If we can make this process faster, that would be great. - Maybe FAISS can help use with this?
- We need to figure out proper balance between processing speed and the amount of frames we are checking -Maybe allow for user to choose between slower or faster processing?



What's Next?

Demonstrate and validate the effectiveness of FAISS for logo similarity search, and integrate it into the application backend.

Repurpose existing frontend components to build the video processing interface.

Define the exact specifications for the metrics and insights to be returned to the user

Implement targeted logo detection within video content, allowing users to search for a specific logo across frames.

Try to figure out what we need to do to make the boundary box transformations more smooth so that it doesn't appear to the user that it's being redrawn in between frames.

