Tidier: Automating Vacation Montage Generation with the Twelve Labs API

By Umer Qureshi – umerqis21@gmail.com – github.com/UmerQureshi21

Overview

Travelers often record hours of footage from their vacations, making it difficult to quickly revisit specific moments. For example, finding shots of the Eiffel Tower may require scrolling through minutes of unrelated clips. In order to fix this problem, I developed Tidier, an app that builds automated vacation montages, so that I can combine all the instances of what is important to me, without the waste of time. This is where the Twelvelabs API showed me its true capabilities.

Technical Architecture

The backend is built with Spring Boot, which manages all API routes, video storage, and FFmpeg command-line operations, while Spring Data JPA and Hibernate handle PostgreSQL database access. Users upload videos that contain the moments from their vacation that they want to view, and any non-MP4 videos are converted using FFmpeg. The videos are uploaded to the Twelve Labs index. Users then select which videos to include in the montage, provide what they want to view, and a title, and click Generate. Twelve Labs searches for the topic in each video and returns timestamps for each instance of that topic to FFmpeg, which trims the clips and combines them into a final montage. The Spring Boot server then serves the completed montage to the React frontend for viewing and download.

Design Decisions

A couple things are different from this application compared to other projects built with Twelvelabs. First is that I used Java to interact with the API, which does not have an SDK at the moment, as opposed to python and Node. This design choice was to show the seamless integration that Twelvelabs provided with enterprise backend systems. Spring boot is used a lot by banks and other enterprise software because Java, in its object oriented nature, is easily expandible and modifiable. Tidier, therefore, proved that Twelvelabs is able to be easily integrated into enterprise systems and is readily expandable, along with its simplicity in personal projects as well.

I chose FFmpeg for video converting, trimming, and combining, because it is able to do the aforementioned tasks without reencoding the videos, thus reducing CPU overhead, and getting high quality montages.

TwelveLabs is the obvious choice for video analyzing tasks over other tools like Azure Video Indexer, because of its pure simplicity. The TwelveLabs playground on their site is super easy to use, making experimentation and testing fast and intuitive. Unlike more complicated enterprise platforms, TwelveLabs focuses specifically on video understanding and timestamp generation and is better at retrieving that data than those platforms, which meant we could quickly extract precise moments from vacation footage without needing to build custom computer vision models or deal with complicated setup. Furthermore, the pricing was amazing for students like myself who love experimenting with different AI tools, which opens a lot more possibilities for me to build what I am passionate about in the future.

Challenges & Solutions

For Tidier, I had to learn about FFmpeg, which is a command line tool that allows you to combine videos without reencoding. This was new to me as it wasn't a framework with clear documentation, so I had to do a lot of researching as to what modes I should do, what files work with it, etc. The main

problem was figuring out what to write so that the combined video can be something playable on my mac finder. Instead of giving up, I broke the problem into smaller steps: I first got trimming to work on a single clip, then worked on preserving audio tracks, and finally focused on concatenating multiple files. I spent a lot of time experimenting with different flags, looking through FFmpeg forums, and running small test cases until I understood exactly how FFmpeg handles codecs and container formats.

Results and Future Work

Tidier successfully generates personalized video montages based on what topic the user inputs, making it easy to view the meaningful moments from their vacations. Looking ahead, several enhancements could make the app even more powerful and user-friendly.

One idea is Montage Topic Suggestion, which would use the Twelve Labs API to automatically summarize each video and suggest relevant montage topics in order to help the user decide.

Another feature is a Smart Montage Library. Twelvelabs will give tags for each summarized video such as “beach trips,” “food tours,” or “family moments”, allowing montages to be stored in a personal library. They will be filtered or searched by the tags. Together, these features would allow users to organize, access, and relive their favorite moments more quickly and intuitively, all because of the AI capabilities of Twelve Labs.