Version 1 Review

Project Overview

My project focuses on egg quality detection using computer vision techniques, specifically through color classification in the HSV color space. The goal is to distinguish between “Good Eggs” and “Bad Eggs” using a webcam or pre-recorded video. I implemented KMeans clustering to extract the dominant HSV color from a fixed region of interest in each frame. Based on the clustered color values, the system classifies the egg and labels the video feed accordingly. I designed this to work both in real time and using uploaded videos so that the professor or others can test it without needing physical eggs. The project is built in Python using OpenCV and scikit-learn.

Progress and Implementation

So far, I have completed the main components of the egg detection system. The current version supports both real-time webcam input and video file processing, which was critical for testing in environments where physical eggs may not be available. I created a fixed bounding box ([200:400, 180:460]) where the algorithm looks for the egg and applies HSV-based classification. KMeans clustering is used to determine the dominant color within this bounding box, and based on the hue, saturation, and value, the system categorizes the egg as “Good Egg,” “Bad Egg,” or “No Egg.” One important recent update I implemented was modifying the script so that the bounding box and label only appear if an egg is actually detected, rather than being drawn on every frame.

I recorded a sample egg video and used it to test the system. The script processed the video correctly and applied the appropriate labels in most cases. I submitted a pull request with my progress, and the professor reviewed it and responded positively. He provided a helpful suggestion via Pull Request #5 by implementing a more robust ellipse-fitting method to locate the egg dynamically, which avoids the rigidity of a fixed bounding box. This ellipse can also be used to generate a more accurate bounding box for HSV classification. I have reviewed this improvement and plan to integrate it into the next version.

Challenges and Reflections

While the current implementation works and produces expected results in many cases, the bounding box does not always behave as intended. In some frames, especially when the egg moves slightly or lighting changes, the system either fails to classify correctly or mislabels the region. I initially thought that using a fixed box would be sufficient, but I now realize that a more dynamic method, like the one suggested by my professor, will make the detection more accurate and consistent. I also encountered issues trying to upload my test videos to GitHub. Even though my video files are under 10 MB, they sometimes failed to show up on the repository. I suspect this may be due to special characters or spaces in the filenames, or perhaps I missed a step in the Git

add/commit process. I will retry using clean filenames and verify that the commit and push go through properly.

I had originally planned to test using various egg types or lighting conditions, but after working with the same eggs repeatedly, I found it more beneficial to focus on fine-tuning the algorithm using consistent test subjects. I do not plan to change the eggs going forward, as keeping the test set constant helps me focus on algorithmic improvements. One unexpected discovery in this process was the effectiveness of analyzing the V (value) channel in HSV for detecting whether an egg is even present in the frame. This helped me suppress false detections and improved the clarity of results.

Future Plan and Milestones

In the coming weeks, my priority will be to replace the fixed bounding box with the dynamic ellipse-based detection method provided by my professor. This should address many of the issues I've encountered with egg misalignment and inconsistent labeling. Once that is complete, I plan to implement a CSV logger to save the classification results for each frame so I can analyze performance quantitatively. If time permits, I may also build a minimal GUI using Tkinter to allow users to upload a video and see the output with a single click, instead of modifying script variables. Toward the final stages of the project, I will focus on writing the final report and preparing a clean, well-documented demo for submission.

Self-Assessment

Overall, I am satisfied with the progress I have made. The project is functional, and I have received valuable feedback from my professor, which I plan to incorporate into the next version. I believe I can complete all major goals within the remaining time. I also appreciate the flexibility in deadlines, as it allowed me to focus on getting the core logic working first before worrying about integration or polish. Some parts of the project, such as HSV threshold tuning and integrating video input, were easier than expected. Others, like getting the bounding box to behave properly and pushing large files to GitHub, turned out to be trickier. Still, I feel confident about where the project is going, and I'm motivated to refine it further in the coming version.