

# GRAPHICS PROGRAMMING - FINAL

## Project Overview:

For this project, I'll be developing an image processing application that relies on the webcam for a range of exciting functionalities. I'll be experimenting with tasks such as image segmentation and color space conversions. Additionally, I'll implement face detection features with filters like face blur or pixelation. On top of that, I'll be adding two more extensions: one is a game, and the other is a fun mode. It's all about exploring image processing while making the most out of webcam technology!

## Findings on Image Segmentation using Color Channels:

During the image segmentation process, I explored the utilization of various color spaces such as HSV (Hue, Saturation, Value) and CMYK (Cyan, Magenta, Yellow, Black). These color spaces were chosen for their effectiveness in segmenting images based on different color characteristics.

## Challenges Encountered and Solutions Implemented:

One major challenge I encountered was managing the segmentation functions efficiently. Initially, I had separate functions for each color segment, with and without thresholding. However, consolidating these functions into a single one posed difficulties, particularly in returning and calling them within the drawing process. After conducting thorough research, I addressed this issue by returning the segmentation results as an array and then calling them using indices in the drawing function. This approach streamlined the code and improved its maintainability.

## Progress Toward Project Completion:

In progressing towards completing the project objectives, I encountered significant challenges, particularly with implementing the extension feature. The focus was on integrating a fun mode allowing users to draw using hand movements, reminiscent of childhood games. However, a major obstacle emerged from the constant refreshing of the background in each frame, disrupting the seamless drawing experience.

The main issue stemmed from the need to refresh the background to prevent other elements like draw boxes, landmarks, and pose from being overwritten. However, I also needed the drawing from the right hand to be overwritten to appear as lines. Despite multiple attempts, I couldn't find a satisfactory solution to reconcile these conflicting requirements.

## Uniqueness of the Extension Idea:

First Extension, I utilized the objectdetect.js library to detect facial features, specifically the head. Drawing inspiration from my childhood memories, I recreated the classic game "Hen Catch Eggs" with a modern twist. Using facial detection technology for movement controls adds a novel dimension to the gameplay. Players control a hen to catch falling eggs while

avoiding missing three. It's a simple yet engaging game that seamlessly blends nostalgia with technology for a delightful gaming experience.

Second extension: With the hand-drawing challenge still persisting, I opted for a different direction and created a fighting game experience. In this iteration, players engage in dynamic battles against formidable opponent, utilizing a range of combat moves to secure victory. Every 10 seconds, the opponent throws objects, making the game harder and needing quick reactions. Additionally, I included entertaining videos for players to mimic, adding an element of amusement and engagement to the gameplay.

### **References:**

- 1.Images - <https://www.canva.com/>
- 2.Video - <https://www.youtube.com/watch?v=wu4NYMmk8pk>
- 3.PoseNet - <https://youtu.be/Olo-DIOkNVg?feature=shared>
- 4.Music - <https://pixabay.com/>