1. Model Architecture:

I selected MobileNetV2 as the classification system.

MobileNetV2 is specifically designed for embedded vision applications. It utilizes depth wise separable convolutions, which significantly reduce the computational cost and number of parameters compared to standard CNNs. This architecture allows us to achieve high accuracy while keeping the final model size well below the 10 MB limit

2. Data Preparation & Preprocessing :

Synthetic Dataset Generation: Made datasets for both red and blue colour box(scroll).

Grayscale Invariance: To make the model robust against lighting colour shifts I converted all training inputs to grayscale. This forces the model to learn the *morphological features* (stroke shapes) of the characters rather than relying on colour values.

Random Rotation: +-20 degree to account for movement and alignment.

3. The Hybrid "Gatekeeper" Architecture:

To meet the inference latency requirement of < 20 ms, I implemented a two-stage hybrid pipeline instead of relying solely on the neural network:

Stage 1 (HSV Filter): I use efficient HSV colour thresholding to detect Red regions of interest (ROI). If no valid Red box is detected, the system immediately returns a "Search" status.

Stage 2 (CNN Inference): Only when a high-confidence Red box is found is the image cropped and passed to the CNN.
Benefit: This prevents the heavy CNN from processing background noise (walls, floor, blue blocks), significantly reducing average power consumption and latency.

Remarks and Drawbacks (will try to fix it):

1. In my model I assume I know which colour will be given to our team (blue or red). (The model I have submitted is with red block as our colour.)
2. It is susceptible to background if there is an object similar to the box.
3. It is sometimes assuming my face as a box (specific expression).