

21 - Prática: HSV com Opencv (III)

Higor Miller Grassi

Descrição da Atividade: No vídeo, foi desenvolvido um código em Python utilizando a biblioteca OpenCV para capturar vídeo em tempo real e detectar as cores vermelha, verde e azul em uma imagem, com a detecção foi realizada com base no espaço de cor HSV, criando máscaras para isolar as regiões de interesse em cada cor.

Na atividade realizada, o objetivo foi desenvolver um código para detectar e isolar as cores vermelha, verde e azul em uma transmissão de vídeo utilizando a biblioteca OpenCV. O código foi estruturado para capturar imagens em tempo real a partir de uma webcam, processá-las e exibir, separadamente, as regiões que correspondem a cada uma dessas cores. A abordagem utilizada foi baseada na segmentação de cores no espaço de cor HSV (Hue, Saturation, Value), que é mais eficiente do que o espaço RGB (Red, Green, Blue) para esse tipo de tarefa.

Primeiramente, o código abre o dispositivo da câmera utilizando o comando `cv2.VideoCapture(0)` e captura os frames do vídeo. Após capturar a imagem, o código converte a imagem do formato BGR (Blue, Green, Red) para o formato HSV, através da função `cv2.cvtColor()`. O espaço HSV é vantajoso para a detecção de cores porque separa claramente a matiz (a cor) do brilho e saturação da imagem, facilitando a criação de máscaras de cores específicas. Isso torna a detecção mais robusta e menos sensível a variações de luz, por exemplo.

Para cada cor, foi definida uma faixa específica de valores de Hue (matiz) no espaço HSV, para o vermelho, a faixa de matiz vai de 161 a 179, para o azul vai de 94 a 126 e, para o verde, a faixa vai de 35 a 85, e a função `cv2.inRange()` foi usada para criar uma máscara binária, onde todos os pixels dentro dessa faixa são marcados como "1" (branco) e os outros como "0" (preto). Em seguida, a operação `cv2.bitwise_and()` foi aplicada para isolar essas regiões de cor específica na imagem original, mantendo apenas as áreas correspondentes à cor desejada.

Ao final do processamento, o código exibe quatro janelas, uma mostrando o vídeo original e outras três mostrando as máscaras resultantes para o vermelho, azul e verde, respectivamente. Dessa forma, o usuário pode visualizar as regiões da imagem que correspondem a cada uma dessas cores, e o código continua executando até que o usuário pressione a tecla "Esc", encerrando a captura e liberando os recursos. Esse processo de detecção de cores pode ser usado em diversas aplicações, como rastreamento de objetos ou análise de cenários em tempo real.

Conclusão: A atividade permitiu aplicar técnicas de processamento de imagem para a detecção e segmentação de cores em tempo real, utilizando o espaço de cor HSV, e o código desenvolvido demonstra como isolar e exibir regiões específicas de cores, o que pode ser útil em diversas aplicações, como rastreamento de objetos e análise de cenas.