Real-Time Colour-Based Object Tracker

Nitin S. Tiwari

Department of Information Technology, Fr. Conceicao Rodrigues Institute of Technology, Vashi
Navi Mumbai, Maharashtra 400703 India
tiwarinitin1999@gmail.com

Abstract **–** Object tracking is an important domain in Computer Vision. It involves the process of tracing an object across a series of frames. Colour-based object tracking is a technique of tracking the objects on basis of the HSV colour model. HSV which stands for *Hue Saturation Value*, is a colour model that is often used in place of RGB in graphical programs. This paper gives a brief introduction about the project which describes how objects can be detected on the basis of colour segmentation in real-time using WebCam by using open source software tools such as Python, numpy and OpenCV.

Keywords: tracking, object, video processing, computer vision, colour, Python, OpenCV, numpy

I. INTRODUCTION

Computer Vision, often abbreviated as CV, is defined as a field of study that seeks to develop techniques to help computers "see" and understand the content of digital images such as photographs and videos.

Object tracking, using video sensing technique, is one of the major areas of research due to its increased commercial applications such as surveillance systems, mobile robots, medical therapy, security systems and driver assistance systems. Object tracking, by definition, is to track a moving object (or multiple objects) over a sequence of images or frames.

Following are the existing systems of object tracking:

- Threshold based segmentation
- Gaussian mixture model
- Colour histogram and gradient based segmentation

Threshold based segmentation:
Thresholding is the simplest method of image segmentation. From a gray scale image, thresholding can be used to create binary images. Binary images are produced from colour images by segmentation.

Gaussian mixture model:
A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters.

Colour histogram and gradient based segmentation:
In colour histogram based segmentation, a histogram is computed from all of the pixels in the image, and the peaks and valleys in the histogram are used to locate the clusters in the image. Color or intensity can be used as the measure.

II. LITERATURE SURVEY

A. HSV Colour Model

HSV colour model is an alternative representation of the RGB model designed in the 1970s by computer graphics researchers to more closely align with the way human vision perceives colour-making attributes.

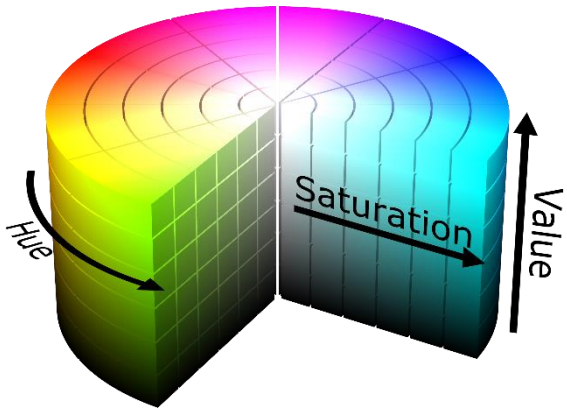

Fig 1: HSV Cylinder

B. Advantages of HSV Colour Model

1. HSV Colour Model makes use of upper and lower limit values of a particular colour which makes it possible to detect all shades of that colour between that range unlike RGB model which can detect only one definite colour of specified RGB value.
2. Any desirable colour can be detected in the image.

C. Drawbacks

1. Sensitive illumination changes lead to more background noise that affects the quality of desired output.
2. Varying light conditions, shadows and other occlusions are the factors to be considered.
3. For better results, the best suitable upper and lower limit HSV values are required to be known for the desired colour to be detected.

D. Applications

1. Colour-based object tracking is used in video surveillance.
2. Machine Vision Systems make use of colour-based tracking and detection for identification and quality testing of objects such as fruits, vegetables, etc.

III. PROPOSED SYSTEM

Colour-based object tracking takes place in the following steps.

1. LIVE WebCam Feed.
2. Conversion of BGR to HSV and Thresholding
3. Image Segmentation and Binary Masking
4. Colour Extraction

Step 1: LIVE WebCam Feed.
Take LIVE video feed from the user through the WebCam.

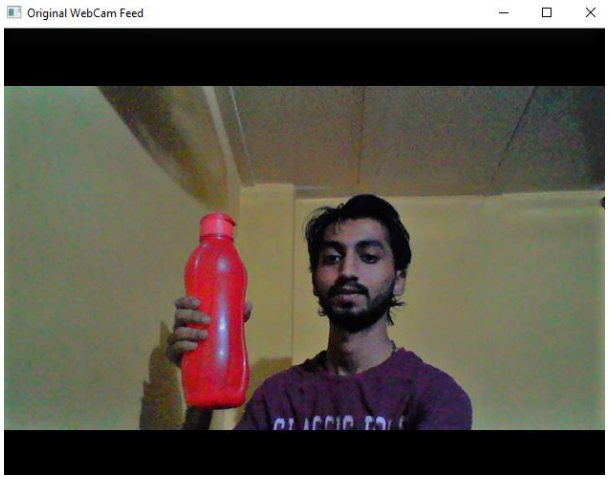

Fig 2: Screenshot of Original WebCam Feed

Step 2: Conversion of BGR to HSV and Thresholding.
All the pixels of the captured frames are then converted from BGR to HSV colour model in real-time using the **COLOR_BGR2HSV** function in OpenCV.

After the conversion of BGR to HSV model, an array of threshold values i.e. lower and upper limit HSV values are defined by using the formulae given below using the Python numpy library.

low = [H-X, 50, 50]
high = [H+X, 255, 255]

where H = computed HSV value
X = suitable factor

The threshold values vary from colour to colour.

Step 3: Image Segmentation and Binary Masking.
A *mask* is an image of the same size as the original image with white pixels indicating the region of interest and black pixels everywhere else, which is obtained by thresholding. This process is known as *Binary Image Segmentation* or *Binary Masking.*

Fig 3: Screenshot of Image Mask

Step 4: Colour Extraction.
The original captured frames are processed by performing bitwise AND operation between the original image and the mask to extract only the desired colour and masking the rest and the final result is obtained.

Fig 4: Screenshot of Colour Tracking

IV. CONCLUSION

Object Tracking is an important field in Computer Vision which consists of video processing and thus having tons of applications. As explained throughout this report, OpenCV was the preferred tool to dive into along with Python and numpy. This paper gives an idea about how objects can be tracked on the basis of their colour which can be helpful in various applications.

There is more to be done to achieve more polished functionality of the project by overcoming the drawbacks and improving the efficiency of the desired output. All source code and the documentation is freely available on my GitHub profile at https://github.com/NSTiwari/Real-Time-Colour-Based-Object-Tracking-using-Python-and-OpenCV

V. REFERENCES

[1] IEEE website [Online]. Available: https://ieeexplore.ieee.org/document/6731341

[2] Towards Data Science [Online]. Available: https://towardsdatascience.com/people-tracking-using-deep-learning-5c90d43774be

[3] Wikipedia [Online]. Available: https://en.wikipedia.org/wiki/HSL_and_HSV

[4] Stack Overflow website [Online]. Available: https://stackoverflow.com/

[5] Mark Meysenburg, “Image Processing”, GitHub [Online]. Available: https://github.com/mmeysenburg/image-processing