

Object Tracking and Motion Detection System for Synthetic Biology Laboratory

Team Members: Chaitanya Kulkarni

Varshith H Anilkumar

A Brief Description on our project:

Procedures and protocols carried out in a synthetic biology lab or a wetlab can be extremely tedious and require a great deal of attention. It will be hard to keep track of all the protocols carried out by a person or a robot through a simple surveillance or a video recording system. The procedures for some of the more detail oriented tasks such as DNA assembly will require a person to keep in mind the minor details of the sequences and has a possibility to lead to confusion and human error. Similarly, due to the sensitivity of such tasks, automating such processes using a robot is also extremely difficult.

We propose a system based on computer vision which detects a moving person and has the ability to track any laboratory apparatus based on color. This system is proposed to be used in combination with NeuralTalk which is a deep learning project developed at Stanford University by Andre Karpathy and Li Fei Fei. Neural Talk predicts the gesture or the content of an image through textual annotations. This combined with our system can accurately keep a track of the activities and procedures carried out in the lab such as DNA Assembly and PCR.

Major Software Components:

Open CV: OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision applications, It has a lot of functions required to process pixels and feature vectors in images such as histograms. It requires Numpy which is a math library for huge matrix operations.

Scipy: SciPy is a Python-based ecosystem of open-source software for mathematics, science, and engineering. It contains many modules for performing such operations as graph arithmetic and scientific computing.

Imutils: A series of convenience functions to make basic image processing operations such as translation, rotation, resizing, skeletonization, and displaying Matplotlib images easier with OpenCV and Python.

Caffe:

- Caffe is a deep learning library. It was a project developed at Google.
- Caffe enables NeuralTalk to build a deepnet by configuring it's hyperparameters
- It has sophisticated layer creation objects which enables to build many different nets within the layers, such as the vision layer, a loss layer, an activation layer and a few others
- Each layer performs a different function and takes on different roles.
- This is made use of by Neural Talk which helps predict the gestures in images through text.

MotionDetector

Overview

The motion detector detects any moving object in the frame by constant processing of the delta frame threshold at each instance and performs a foreground segmentation which differentiates a moving object from the background and processes the difference in the threshold value of pixels.

Dependencies:

Python 2.7

Open CV 2.4 or greater, which can be installed via pip.

imutils

argparse

Running the code:

The program can be run through command prompt by going to the appropriate directory or through a suitable Python IDE. By default, it will run using your webcam, to run on a recorded video in your directory, please include the argument in the command while running it through command prompt (for ex: `python Motion-Detector.py --video videos/example.mp4` (note: the folder 'videos' has to be in the same directory))

ObjectTracker

Overview:

The object tracker tracks an object based on color. We set thresholds for identifying colors by converting the whole image into a hue saturation value, then a binary mask is created which detects the object of only that color based on the threshold values we specified.

Dependencies:

OpenCV 2.4 or greater

imutils

Numpy which is also a dependency for Open CV

Running the code:

The program can be run through command prompt or through a suitable IDE. By default it captures the video through your webcam, but, you can also run it for a recorded video on your device, by running the program on command prompt by specifying suitable arguments(for ex: `python Object-Tracker.py --video videos/example.mp4` , note: the directory should be the same as your `Object-Tracker.py` folder.