**Human Segmentation and Mask Generation in Images**

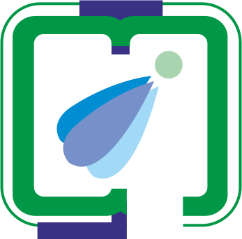
***By***

Gursimar Singh

(Roll No. 2015101)

**Supervisors:**

|  |  |
| --- | --- |
| Internal  Dr. Atul Gupta  Associate Professor  PDPM IIITDM Jabalpur | External  Mrs. Sheena KP  Senior Manager  Canon ISDC, Bengaluru |



Electronics & Communication Engineering, B.Tech 2015

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, DESIGN AND MANUFACTURING, JABALPUR

29th May, 2018 - 14th June, 2018

# **Introduction**

Human segmentation in natural environments is a difficult task because of the random changes produced in natural scenes: illumination changes, moving objects, moving camera position etc. Because of the nature of the problem, a common way to proceed is to discard most part of the image so that the analysis can be performed on a reduced set of small candidate regions. The problem statement is to generate a mask segmenting the human from the background. We have only the body key point information in order generate a mask.

# **Progress**

During the period of this report I have worked on various image processing techniques for segmenting the human from the background. These include the following:

1. Weighted Graph Cut algorithm
2. Simple Linear Iterative Clustering (SLIC) superpixel segmentation
3. Color based Gaussian Mixture Model
4. Color based Region growing (Watershed algorithm)
5. **Weighted Graph Cut algorithm**
   1. **Approach**

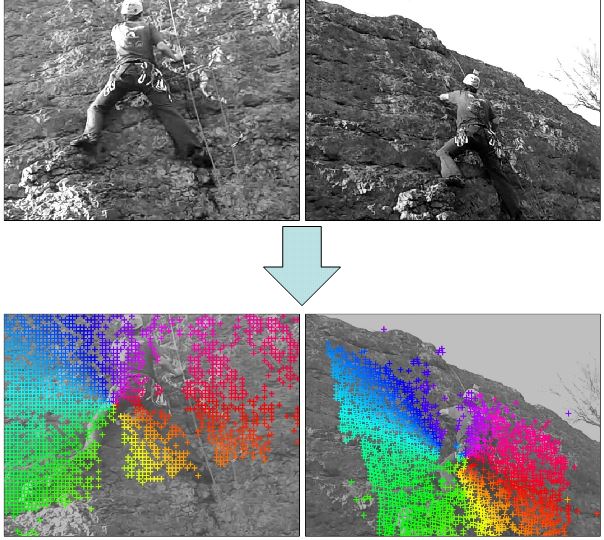
A feature matching approach was tried to track the person. The basic idea of this approach is to initialize a bounding rectangle for the first frame. Features are extracted of the points within that ROI and then matched in the next frame. Various features descriptors and matchers are available but they do not return dense correspondences between the images. A robust and dense matching algorithm was needed. Thus, a combined deep matching and deep flow approach is used.

* + 1. **Deep Matching and Deep Flow**

Deep Match is a 6 layer convolution based algorithm which finds dense correspondences between two images. It was developed by Jerome Revaud in 2013. Deep Matching relies on a deep, multi-layer, convolutional architecture designed for matching images. It can handle non-rigid deformations and repetitive textures, and can therefore efficiently determine dense correspondences in the presence of significant changes between images.

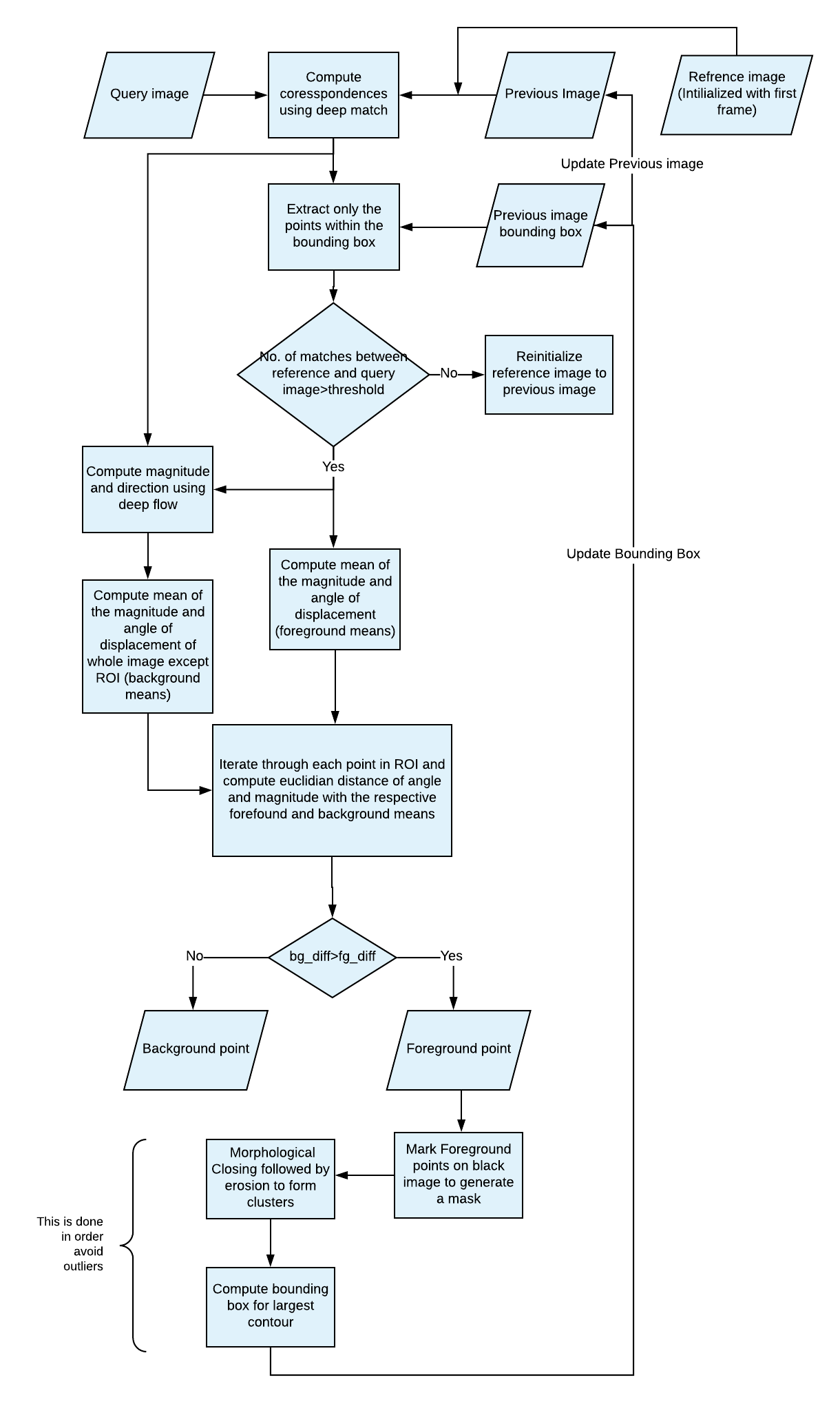
Deep Flow is an optical flow algorithm based on Deep Match. It computes the flow vectors of each pixel of the query image with respect to the reference image. It was introduced in the paper

*DeepFlow: Large displacement optical flow with deep matching Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui and Cordelia Schmid, Proc. ICCV‘13, December, 2013.*

**For the subsequent frames of a video sequences there were about 95% correspondences detected for each pair. In the following image each cross is match:

*Figure 1.1: Dense correspondences in subsequent frames (Image Source: INRIA)*

The challenge with tracking using deep match was that the ROI initialized or the bounding box detected also contained portion of background pixels which also got matched in the subsequent frame. Therefore, there was need to separate the foreground pixels from the background pixels else the error would accumulate and bounding box will continuously increase.

The motion information of each pixel was available, which could be used for segmentation. In some cases both the foreground and the background were moving, but would have different means and variances. Thus, considering them as features a segmentation approach was applied

*Figure 1.2: Algorithm Flow*

1. **Human body Key point tracking:**

The task was to track the essential key points of a human in subsequent frames of a video sequence.

* 1. **Approaches**

Various approaches were tried from optical flow tracking, particle filter and feature matching.

The major challenge was poor quality of images and large motion between frames.

**2.1.1 Optical flow tracker**

Optical flow algorithms are widely used for motion tracking in video sequences. It takes points (x, y) as input and tracks those points. One of the most widely used algorithm is Lucas-Kanade tracker. It has implementation in OpenCV as KLT tracker. The basic idea of KLT tracker is tracking the intensity changes around the point of interest. A good explanation has been provided in the following link:

*https://docs.opencv.org/3.3.1/d7/d8b/tutorial\_py\_lucas\_kanade.html*

**2.1.1.1 Result**

The tracker failed in many cases and sometimes gave results that were out of the image especially as the point approached boundary.

**2.1.1.2 Possible reasons**

* Random intensity variation around the point of interest for example in tracking shoulder point, there is a lot of variation in the skin intensity.
* Random motion of the subject and missing frames in the video sequence

**2.1.2 Tracking using feature matching**

Feature matching is a well-known technique used for image classification, object detection and image stitching. It is used to find correspondences between two images of same scene from different angle or with some motion.

Many feature detectors are available which are scale and rotation invariant like SIFT, SURF, ORB etc. SIFT and SURF are not available for free commercial use whereas ORB is free algorithm. Some other available options are GFTT and AKAZE. They are free but not scale and rotation invariant.

There is no python implementation available to compute descriptors for specific points in an image. The algorithms detect automatically the best points which contain features and then compute there descriptors.

**2.1.2.1 ORB feature matching**

Using ORB descriptors with hard coded key point vectors, combined with constraints in localized ROI’s produced results better than optical flow approach. Results are satisfactory for body parts which have less amount of motion and less variations in background. Used RANSAC, distance based curve fitting to reject outliers and tried to integrate the results with kalman filter but not much improvement was observed.

**2.1.2.2 Deep Match feature matching**

Deep Match produced more dense correspondences (*Figure 1.1*), thus there is high probability that our point of interest will have a match in the next image. Even if the exact point is not available the closest point in a restricted neighbourhood can be selected for which a match exists.

# **Result and Discussions**

During the duration of this report, I worked on two tasks namely human tracking and body key point tracking. They are cases when they fail and there is a scope of improvement.

**Person Tracking:**

The result of the person tracking approach were satisfactory. The tracking is good when the frames are continuous but fails when there are missing frames.

**Further improvements**

* Find way to detect false detection and reinitialize bounding rectangle either by manual intervention or some algorithm.
* Make use of variance of the motion information to improve segmentation of foreground and background

**Human Key Point labelling:**

The results are satisfactory, but there is an offset error which needs to be compensated.

# **Conclusion**

The body key point tracking algorithm has provided satisfactory results and could be used for mask generation.

The person tracking algorithm is still under review and currently I am working to improve its performance.