Indian Journal of Science and Technology, *vol 8(19), DOI: 10.17485/ijst/2015/v8i19/76711, August 2015*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Implementing and Optimizing Template Matching Techniques for Home Automation

Edwin Jose Kundukulam* **and Abraham Sudharson**

School of Electronics Engineering (SENSE), VIT University, Chennai - 600127, Tamil Nadu, India;
edwin.jjose@gmail.com, abrahamsudharson.ponraj@vit.ac.in

Abstract

Background/Objectives: In analyzing the modern world with huge amounts of digital visuals, the image analysis techniques play a major role. It helps in monitoring the organizing the visual data. The contents or objects in the image have significance on image recognition. **Methods/Statistical Analysis:** There are various algorithms which recognize the real world object. This system proposes a model which uses modified template matching technique based on SURF algorithm and squared difference error method. **Findings:** The template matching is done based on image features comparison. SURF algorithm of template matching is based on keypoint extraction from images whereas the squared difference error algorithm of template matching is based on pixel feature comparison. To analyze the performance of the algorithms in real world, both the algorithms been implemented using embedded microcontroller Raspberry pi and Pi Camera. **Conclusion:** The Electrical appliances used in home were considered as objects to be recognized. The appliances the detected via Pi Camera. This algorithm can be further explored in areas like multiple object recognition and various other autonomous machine vision system.

Keywords: Home Automation, Raspberry Pi, Squared Difference Error, SURF, Template

1. Introduction

In analyzing the modern world with huge amounts of digital visuals, the image analysis techniques play a major role. It helps in monitoring and organizing the visual data. The contents or objects in the image have significance on image recognition. It would be definitely an advantage if the methods for image analysis could detect a particular image or video without any human intervention. In the image, there are objects which are useful in determining the characteristics of a particular image. An object will have maximum features for uniquely identifying an image. So there is a need for object recognition methods for real time machine vision applications. The idea behind object recognition is to extract the features of the object from an image by matching with a set of known features. For humans it is possible to detect an object in real world easily. But when a machine is considered, it is not possible to recognize an object by itself. Thus object recognition techniques are implemented in such a way that it is less complex and more efficient. Also, the implementation of the techniques in the real time scenario is also a factor.

There are various approaches for object detection. Most of the successful methods are done by collecting local descriptors of the image for object detection. The object recognition techniques can be broadly classified based on the approach.

The object can be recognized based on templates, colors, shapes and local or global features. In[1], the authors proposed a method for template matching based on morphological transformations used in Inertial Navigation Systems (INS) for tracking and detecting ground objects. The method is altitude and rotation invariant. In[2], the template matching is done with a simple similarity measure named as SSD (Sum of Square Differences) which results in matching. In[3], Fahad Khan proposed the use of

*Author for correspondence

color attributes for detection based on criteria like feature combination, photometric in variance and compactness. For shape based features matching, in[4] and[5] authors proposes algorithm by extracting the edges in an image using Gradient Vector Grinding (GVG). For detecting an object using local and global features, authors in[6, 7] proposes a technique to find objects with varying features. In[8], the Scale Invariant and Feature Transform (SIFT) algorithm is proposed which is very efficient but computationally intensive. Also, SIFT algorithm is scale, rotation and illumination invariant.

Open CV is an open source library widely used for image and video processing applications[9]. In[10], the authors proposed a method to perform image operations like scaling, rotation, edge detection using Open CV and QT and implemented it in mobile platform. The authors in[11] gave a brief overview of Open CV and also about the functions and the libraries used in the image transformations. In this paper, an application of Object detection algorithm based on template matching is implemented in raspberry pi. This algorithm is used to control electrical home appliances in home. The paper is organized as follows. Section II discusses the template matching technique used for object recognition. Section III presents the overall flow of how template matching function is implemented using OPEN CV functions. In Section IV, a brief introduction to hardware setup for object recognition is explained. In section V, the experimental procedure is discussed. Section VI gives the overall results, discussions along with some limitations. Finally in section VII, the conclusion and future scope is elaborated.

2. Template Matching

In this process of template matching, a template of the image is compared with the actual image to find the match. After receiving the input image, this method compares the received image with the template stored in the database. Template matching is generally done based on pixel or feature. Templates are frequently used for recognition of characters, numbers, objects, etc. In the Figure 1, a template of alphabet A is compared with the input image for matching.

Open CV helps in template matching with its predefined libraries. The searching for the template depends upon the size of the image. More the size of the image, more time it will take for appropriate template match.

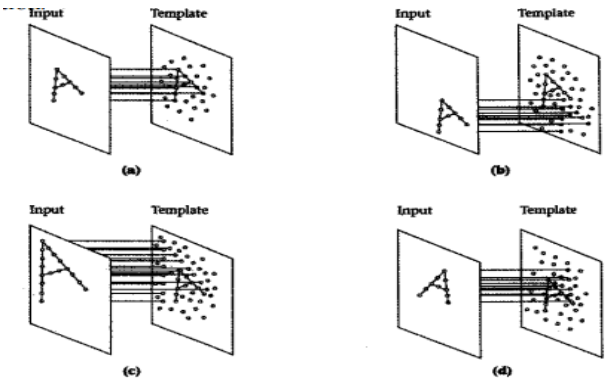

Figure 1. Template matching technique.

2.1 Template Matching using Squared Difference Error Method

The template image and the captured image were read and converted to grey scale images**.** The squared error method works on pixel feature comparison of template image and the original image. Based on the equation of squared difference error and Open CV library , the matched result was found out as shown in figure below. The Figure 2 shows matching result for FAN.

The darker intensities denotes good matching region and brighter intensities signifies least matching. Based on the maximum and the minimum matching, the template was recognized.

Also the resolution of template image was kept smaller than the captured image. For appropriate matching the distance was kept fixed at 10-15cms. Template matching based on squared difference error is dependent on the distance and it will also vary when the object is rotated.

So based on whether the matching occurs or not, the electrical appliance gets activated via GPIO pins of raspberry pi.

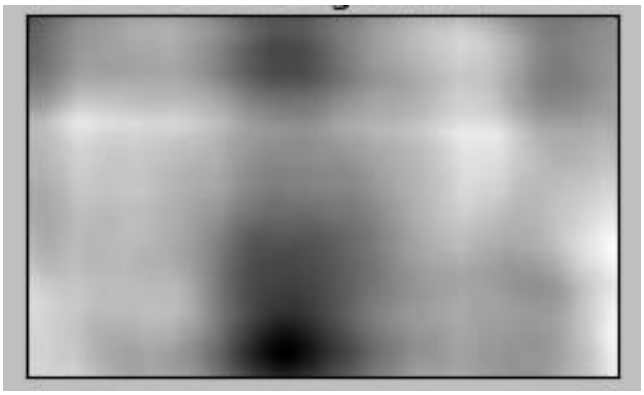

Figure 2. Matching result for fan.

2.2 Template Matching using SURF Algorithm

The template image and the original image is first read from the database of raspberry pi. The image is further converted to grey scale because the Open CV image processing operations are done based on grey scale images. Also image processing is faster. Initial Hessian threshold value is set for SURF key point extraction. The threshold value determines the number of key points to be detected and searched. Different threshold value was set and experimented for appropriate matching. More the Keypoints more time it will take for template matching. And moreover, it also consumes the processing capabilities of CPU. In this system, the threshold value is varied between 300 to 1000 and the response time is observed. The Keypoints and the Descriptors were extracted from the Captured image and the template image. Mapping of Keypoints were done based from the template image to original image. Depending upon the accuracy of matching, red color was given to denote perfect match and blue color was given to show imperfect match. If the number of matched keypoints are more than that of unmatched keypoints, the overall image is recognized with template. The figure shown below shows how the matching is done based on key points.

Based on the number of keypoints which corresponds to actual matching the device would either be switched ON or not? The final output was further converted back to binary and the corresponding keypoints are matched SURF algorithm is scale invariant as well as rotation invariant which means even if the distance between the picamera and the object is varied, it would still find the keypoints. Also if the object is rotated it would still find matching. So tests were conducted of the efficiency of SURF algorithm in such conditions. Appropriate lighting conditions were given for good matching.

3. Methodology

The first algorithm considered for template matching is based on Speeded Up Robust Feature Transform (SURF) algorithm. It is a Keypoint based template matching technique. This algorithm is useful when an object needs to be identified between two images in a computer vision system. Also, we can track the movement of the object when we compare both the images. For this template matching algorithm, we consider the keypoints of template image and the original image. Figure 3 shows the flowchart of automation based on Keypoint based template matching. Basically a Keypoint is used to describe an object which is independent of position, rotation and lightning.

In the process of matching, at first the keypoints are extracted from the template image and then the same keypoints are searched for in the original image. This will result in a more robust object detection system which is rotation invariant as well as scale invariant. For a very good match, the object should have lots of textures and with colors and shapes.

The second algorithm is based on squared difference error method. This algorithm does the matching by pixel matching or comparison. In the process of template matching using squared difference error method, the template image pixels are compared with the original image from left to right and top to bottom. Figure 4 depicts the flowchart of matching using squared difference error method algorithm.

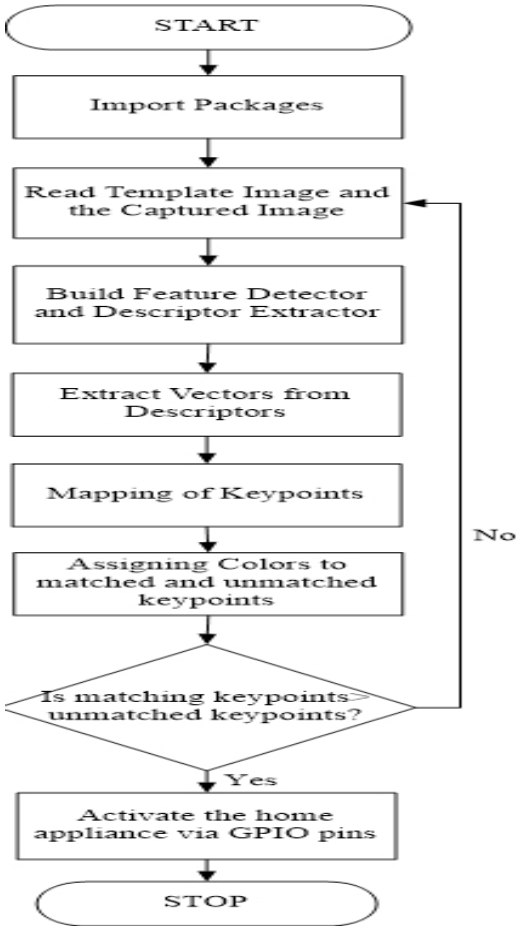

Figure 3. Flowchart for SURF algorithm based automation.

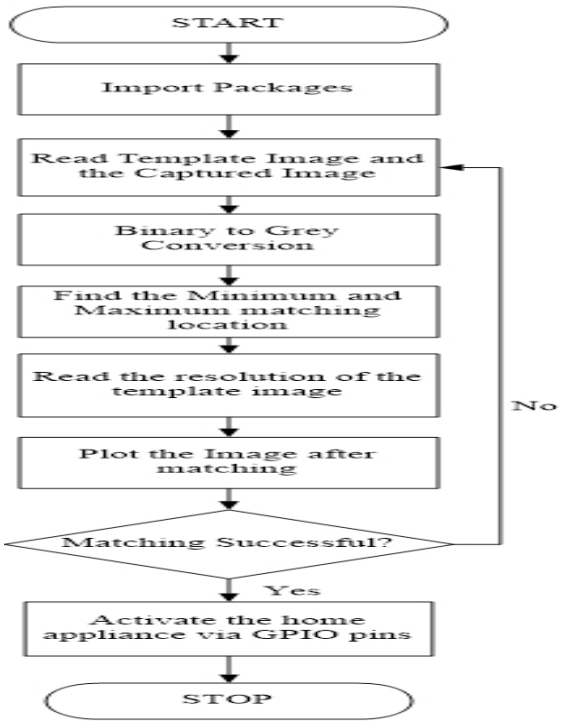

Figure 4. FlowChart of template matching using squared difference error algorithm.

After each comparison, a metric is calculated to represent to obtain the amount of similarity. Also a threshold value is needed for improving the overall quality of images. In this method, it computes the square difference between the template and the source image, a best match in this case has a zero value. The template image pixels at a particular location (x, y) is compared with the corresponding location of the original image[9]. For perfect match the resultant value should be zero. Equation 1 is used to compute the matching.

$$R_{ccoeff} = \Sigma_{i,j}\left[\left(t(i,j) - f(x+i, y+j)\right]^2\right) \quad (1)$$

4. Experimental Procedure and Results

The template chosen and the image captured for home automation using SURF algorithm and Squared difference error algorithm is as shown in Figure 5. The image resolution of the template image should be less than that of the original image. The resolution of the original image considered for the experiment was 1024 X 768 and for template image it was 320 X240.

Figure 5. Image captured by camera and the template stored.

The methodology of template matching using raspberry pi involves the camera which is interfaced with the raspberry pi. It takes the still image of the appliance. If the image is detected, the template matching using OPEN CV is done. The template image should be smaller than the original image. So once if the template is image is recognized the corresponding appliance is controlled via raspberry pi. Template Matching is a method for searching and finding the location of a template image in a larger image.

In the Figure 6, the basic setup for performing template matching is shown. In the setup, the induction motor is kept in front of the camera. A template is also created over the induction motor. The raspberry pi camera is interfaced with raspberry pi via Camera Serial Interface (CSI). Similarly tests are conducted for fan and light using predefined templates.

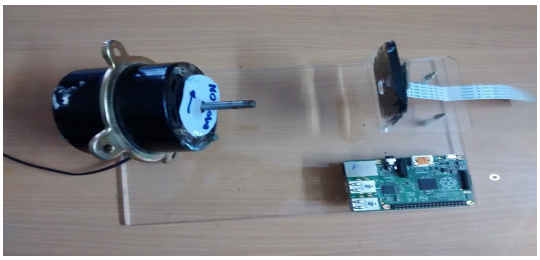

Figure 6. The setup for template matching.

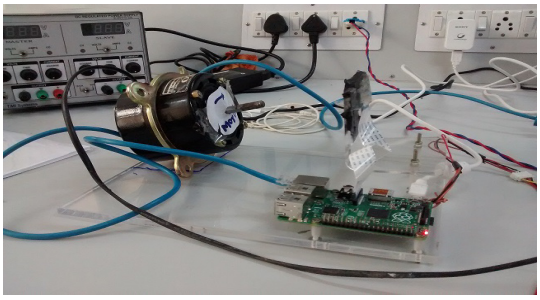

Figure 7. The setup after connection to desktop and hardware.

Figure 7 shows the basic setup after all the connections. The electrical devices are driven by GPIO pins of raspberry pi via the relay circuit. The raspberry pi is connected to laptop via Ethernet cable. And separate power supply is provided for the appliances to be controlled.

Before setting up the experiment, a database of templates that should be matched is stored. So initially the raspberry pi camera would capture the image along with the template of all the appliances to be controlled. Within the image captured, it scans for the template image using template matching method. If template matching is successful, it will display the result. Using SURF algorithm the template matching is done for Fan as shown in Figure 8.

The SURF algorithm depends upon the Hessian threshold value. It determines number of keypoints to be extracted. So tests were conducted to find the optimum value of threshold value that needs to be set for appropriate matching.

The response time of the controller with different threshold value was observed. Figure 9 shows the window of response time for SURF algorithm for hessian threshold of 1000. Total number of keypoints detected are 257 and the response time is 14.84 seconds. When the hessian threshold value was varied between 300 to 1000, the average response time was 12.25 seconds. So, although the SURF algorithm is rotation and scale invariant, the response time is comparatively more.

Figure 8. Template matching using SURF.

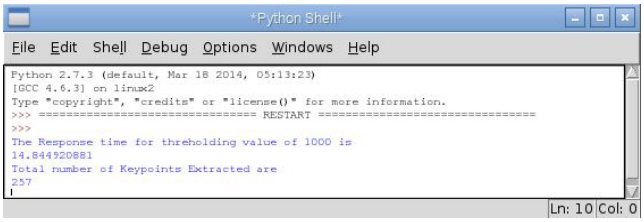

Figure 9. Response time of SURF algorithm.

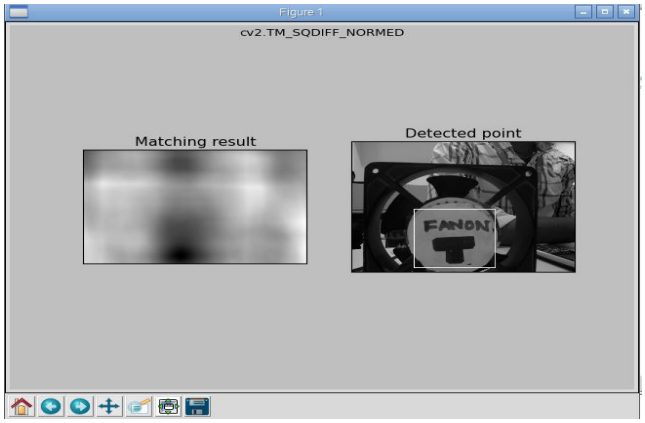

Figure 10. Template matching for FAN.

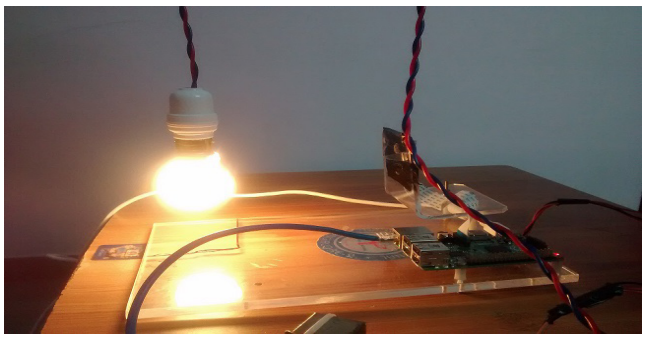

Figure 11. Bulb activated via GPIO pins after template matching.

For template matching using Squared difference error algorithm results, Figure 11 shows two windows i.e Matching result window and detected point window. Matching result window is displayed to show the maximum and minimum template matching after scanning. Based on the matching result window we get the detected point i.e the template image from the input image. In the Figure 10, a template matching window for FAN is shown. The highlighted square box is the template signifies the appropriate match.

The response time of Squared difference error algorithm is comparatively less. Based on the experiments conducted on all the three appliances, the average response time was between 0.5- 1 seconds. It can be further reduced if the template has good features to match.

Based on the template matching done by either SURF or Sqaured difference error method, the device gets turned on as shown in Figure 11.

Similarly the template matching on induction motor and Fan resulted in the activation of device.

5. Conclusion and Future Scope

In this paper, the concept of object recognition using is proposed for home automation using template matching. The algorithms used were SURF and squared difference error method algorithm. Home automation is done based on template matching. Since the template matching technique requires large database of image templates for correct object recognition, it must be used in the applications where limited objects needs to be detected. Although the SURF algorithm provides much more robust object detection which is scale invariant and rotation invariant, the response time is much more when compared to squared difference error algorithm. For a controller with better processing capabilities, SURF algorithm would be a better choice. Squared difference error method has less response time. But the object needs to be placed at a fixed distance. So there is a trade off between the processing capability of the controller and the algorithm used. This concept of object recognition using raspberry pi controller can be further developed in areas including biometric recognition, surveillance, industrial inspection, robotics and many more. For a good object recognition various other factors that needs to be considered like the lighting condition, the rotation invariant feature, positioning and occlusion handling. An efficient object recognition algorithm along with its hardware compatibility will help in binding the physical world and digital world.

6. References

1. Hu W, Gharuib AM, Hafez A. Template match object detection for inertial navigation systems. Scientific Research (SCIRP). 2011 May; 2(2): 78–83.
2. Shectman E, Irani M. Matching local self-similarities across images and videos. IEEE International Conference on Computer Vision and Pattern Recognition, Minneapolis, MN; 2007 Jun 17-22. p. 1–8.
3. Khan F, Muhammad R, et.al. Color Attributes for Object Detection. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI; 2012 Jun 16-21. p. 3306–13.
4. Berg A, Berg T, Malik J., Shape matching and object recognition using low distortion correspondences. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR); 2005. p. 26–33.
5. Moballegh H, Schmude N, Rojas R. Gradient vector griding: an approach to shape-based object detection in robocup scenarios.
6. Papageorgiou C, Poggio T. A trainable system for object detection. International Journal of Computer Vision. 2000; 38(1): 15–33.
7. Dong L, Yu X, Li L, Hoe JKH. HOG based multi-stage object detection and pose recognition for service robot. 11th International Conference on Control Automation Robotics and Vision (ICARCV); Singapore; 2010 Dec 7-10. p. 2495–500.
8. Mauricio M, Denisii S. High level computer vision using openCV. 24th SIBGRAP Conference on Graphics, Patterns and Image Tutorials; Alagoas; 2011 Aug 28-30. p. 11–24.
9. Deepthi RS, Sankariha S. Implementation of mobile platform using QT and OPEN CV for image processing applications. IEEE Conference on Open Systems; Langkawi; 2011 Sep 25-28. p. 284–9.
10. Culjak I, Abram D, Pribanic T, Dzapo H, Cifrek M. A brief introduction to OpenCV. 2012 Proceedings of the 35th International Convention MIPRO; 2012 May. p. 1725–30.