

# Real-Time Object Detection and Recognition Using Internet of Things Paradigm

**Shrddhey Kumar Jain**<sup>1</sup>

<sup>1</sup> Department of Electronics and Telecommunication, Sinhgad College of Engineering, Vadgaon(Bk), Pune, 411041, India

Email: shrddheykain@gmail.com

**Supriya O. Rajankar**<sup>2</sup>

<sup>2</sup> Department of Electronics and Telecommunication, Sinhgad College of Engineering, Vadgaon(Bk), Pune, 411041, India

Email: supriya.rajankar@gmail.com

**Abstract**—Internet of Things is an emerging field wherein a lot of classical approaches can be inculcated. One such approach is found in image processing domain. It is real-time object detection and recognition. Object recognition is considered as a complicated process because the object can be of any shape, size or color. Object detection can be performed with effectiveness by using various prevalent techniques such as Scale Invariant Feature Transform (SIFT), a faster version known as Speeded-Up Robust Features (SURF) and the combination of two very efficient algorithms called as Oriented FAST and Rotated BRIEF (ORB) and so on. Although different techniques are dedicated to the different type of objects. In this paper, an effort has been made to combine the object recognition technique with Internet of Things (IoT) concept. The IoT device acting as an input is the camera that captures the image. The object present in the image is detected and recognized. After that, its information is extracted through the internet and displayed on the screen along with the recognized object. The recognition takes place using the pre-existing database. The database consists of the objects that have salient features which would make the task of recognition unambiguous. The bag of features method is considered in order to make recognition effective. The effective use of Internet of Things is carried out by establishing communication between a camera which acts as an input device and visual output devices. This communication takes place over Internet protocol. In the case of object detection, various parameters such as rotation invariance, scale invariance, intensity change, orientation invariance and partial object detection are also considered to make the system robust. Time consideration is carried out to make the system work in real time.

**Index Terms**—Bag of features, Internet of Things (IoT), Object detection, Object Recognition, Oriented FAST and Rotated BRIEF (ORB), Scale Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF)

## I. INTRODUCTION

Object recognition boasts of being one of the most favorite topics researched under computer vision. The fundamental step in computer vision is to detect as well as recognize an object in order to perform any task related to it. Since computer vision is emerging these days in various applications, it is quite necessary to have an efficient system that can detect, track and recognize the object. There are many systems already in existence for the task of object recognition. A wide variety of techniques are introduced which help to detect and recognize the object and the field is still undergoing plenty of research. The need of object detection can be visualized in applications of industrial automation, military, and other commercial appliances.

The Internet as a resource acts in various fields of computing from basic searching to banking, social Networking to online booking and so on. The new terminology Internet of Things came up with an idea of Everything-as-a-service. This can be considered as a revolution in computing and communication technology [1]. Internet of Things refers to the communication between the physical things with a valid physical address over the Internet protocol without human intervention. The interaction of human to the device is also a part of this concept. This concept of advance inter-networking would be applicable on a large number of applications such as smart home, industrial automation, environmental surveillance, medical applications and much more [2].

Objects that are present nearby are hardly recognized by a machine unless it is taught to. The database for the machine vision would be limited due to memory requirement and hence many of the objects would be unknown to the machine. It is very difficult to get information about such objects because no resource is available which can tell us about it. The solution to this problem is to extract some information to detect an object and recognize it with the help of a resource having knowledge about that. One such omnipresent resource is the web or the internet which has an adequate amount of information available about every object in existence. For

the purpose of involving the concept of the web, the camera can be brought online as a node which can extract relevant features and send the features to the internet by which the information can be extracted. Successful information extraction about the object from the internet can lead to a wide range of future works.

The work described in the paper sheds light on the concept of Internet of Things and its application for Object recognition. A brief introduction of the object recognition techniques is given in the next section. The subsequent sections describe the design of the system followed by the steps undertaken for the software design of the system. For this purpose, the aforementioned object detectors and descriptors are implemented and their performance is compared. The results obtained are described thoroughly in the results section and appropriate conclusion is drawn. This conclusion is stated and justified at the end of the paper.

## II. OVERVIEW OF INTERNET OF THINGS AND OBJECT RECOGNITION TECHNIQUES

The introduction of Internet of Things architecture and about the popular object recognition techniques in the research field is provided in this section. This information is further exploited to design and implement the system for object recognition via implementation of Internet of Things. There are mostly self-created databases that are used for the purpose of object detection and recognition. This makes the comparison of techniques a bit chaotic since the standard object database is not available. However, since the approaches are more or less defined in a fundamental manner, object recognition techniques almost always give a substantial amount of success.

### A. *Internet of Things*

The basic architecture of Internet of Things consists of three layers namely the perception layer followed by the network layer and lastly, the application layer [3].

- The Perception layer—It is used for object identification and to gather information about it. It consists of sensors and actuators which detect the identity of the object and then the gathered information is transferred to another layer. One can say perception layer works as an input parameter to the system.
- The Network layer—This layer transmits the information grabbed from the perception layer to the other devices connected to the network. It is formed by the wired or wireless devices which can provide access to the network or the internet. This layer acts as a communication medium acting between two or more devices connected to the network.
- The Application layer—It is the user interface part of the system. The application layer provides the required functionality to the system by facilitating the solutions that can be applied to IoT technology which can satisfy the needs of the user.

Bringing the objects online in order to communicate them with other resources connected to the Internet is the ideology followed by the Internet of Things. To avail this functionality many different technologies are used which includes RFID (Radio Frequency Identification), Sensor network technologies, Bluetooth tags, Optical tags and so on. In other words, the development is aimed towards the input devices communicating with other physical objects [4]. Problem of memory and storage can be solved by using various new techniques dedicated to Internet of Things [15].

### B. *Object Detection and Recognition Techniques*

Object recognition is generally performed by matching the features present in the image with the pre-existing image available in the database. There are two methods in general which can be used for image matching. One is the gray-level matching and the other one is the image feature based matching. Gray-level matching is generally scaled variant and is not much efficient in image matching hence, also in object recognition, whereas the image feature based matching is generally scale invariant because it matches the feature present in the pre-existing image and that in the target image. That is why in most of the existing image matching systems, the latter method is effectively used. In this technique, the important features that are present in the database image and the captured image are extracted. The matching algorithm helps to match the features of both the images based on different matrices. The different matrices include Gaussian Mixture Model, Hausdorff distance, Principal Component Analysis, etc. Various other parameters also affect in the efficient objection recognition such as time taken by the system for recognizing the object and the percentage of error if the object detected is incorrect.

## III. SYSTEM DESIGN AND IMPLEMENTATION

The framework of the proposed system is shown in fig. 1. Each block is further explained. The system consists of image detection and recognition system and the Network Interface. In this system, the camera is connected to a platform board which is connected through network i.e. the board is online, thus the camera seems to be online. The camera captures the image and then its features are extracted. Based on the features that are extracted, the location of the object can be detected, thus the object is detected. The next target is to recognize the detected object. For this purpose, the class tree of the database available on another server is accessed through the internet. The features of the image obtained from the database are extracted and the matching of interest points of both the database image as well as captured image is done. If the matching is beyond the desired threshold, the object is considered to be recognized. Now, based upon the pre-existing information about the object, it can be linked with the available resources on the internet like Wikipedia or any other application can be made for providing the relevant information.

There are three primary modules of the system that are considered for the implementation. These modules are in turn called as

- Database creation
- Software system design
- Hardware system design

These are explained in the following manner.

A. Database creation

The database that is created for this purpose consists of 10 objects. 5 variants of each object are taken which are normal, rotated version, change in orientation, partial object and intensity change. All these variations are as shown in fig. 2. This 10 object database with its variants is used to perform 5-fold cross-validation for improving system performance and setting the best fit values for the classifier. After that, the database can be modified in real time for adding new objects into it.

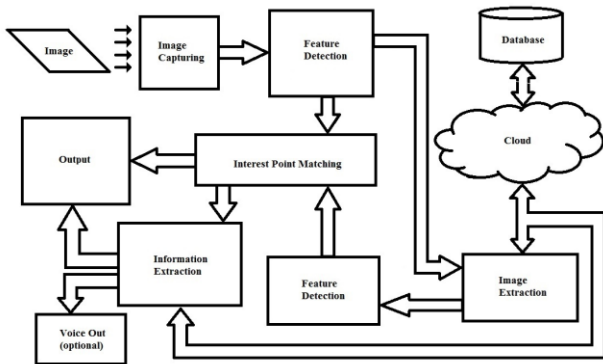


Fig.1. Block Diagram for Object Detection and Recognition through Internet of Things.

B. Software system design

The software system design is carried out by implementing different techniques for object detection and recognition. These techniques are preferred as per the application for what they are used. For this system design, the techniques taken into consideration are Scale Invariant Feature Transform [SIFT] [5], a faster version due to inclusion of integral images called as Speeded-Up Robust Features [SURF] [6], and the combination of two very efficient algorithms called as Oriented FAST and Rotated BRIEF [ORB] [7]. The reason for considering these techniques is that they are model-based techniques and based on feature extraction. Features are the properties of the image consists of blobs, corners, and edges. Due to this, these techniques become rotation invariant, orientation invariant and intensity invariant.

1. Scale Invariant Feature Transform [SIFT]:

SIFT makes use of Difference of Gaussian (DoG) detector that extracts the blobs that are present in the image by performing the approximation of the Laplacian equation  $L_{xx}^2 + L_{yy}^2$  [8]. The diffusion equation that is

given in the scale-space theory [9], [10] gives a basis to show that the Laplacian gives a correspondence with the derivative of the image that is in the direction of the scale [11]. It can be seen that the difference of the neighboring points in a certain direction is almost equal to the derivative in that direction. Hence, image differences at different scales are almost equal to the derivatives with respect to that scale. The process of Gaussian blurring leads to the generation of images at various scales. The difference of Gaussian is the edge detector that is used in the Scale Invariant Feature Transform. The steps that are taken to implement the SIFT are given below:

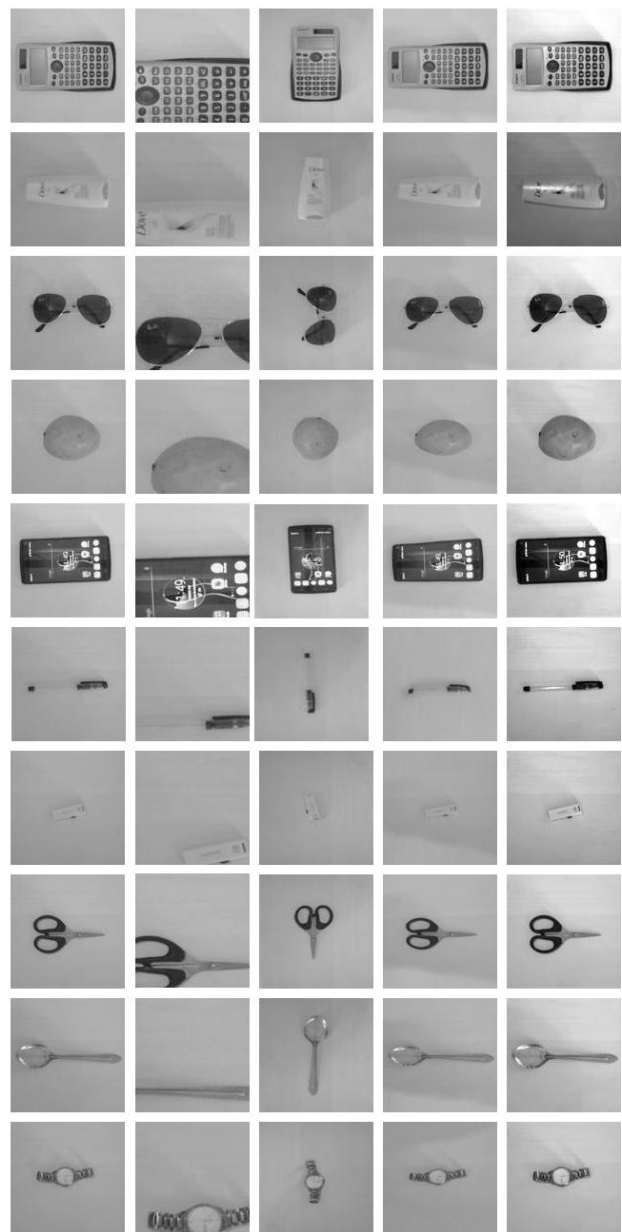


Fig.2. Objects from database, Row: (a) Calculator (b) Conditioner (c) Sunglasses (d) Mango (e) Mobile (f) Pen (g) USB drive (h) Scissors (i) Spoon (j) Watch; Column: (1) Normal object (2) Rotated version (3) Partial object (4) Orientation change (5) Intensity change.

1. Detection of Scale-Space Extrema: Detection of points of interest is carried out in this step. These points termed as 'keypoints' according to the

framework of SIFT, are the maxima points or the minima points of the Difference of Gaussians that are present at the multiple scales. A DoG image  $D(x, y, \sigma)$  is given by

$$D(x, y, \sigma) = L(x, y, k_i \sigma) \quad (1)$$

Where,

$$L(x, y, k_i \sigma) = l(x, y) \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$$

2. **Keypoint Localization:** Scale-Space extrema detection gives a large number of keypoints, not all of which are stable. The following procedure is adopted to localize a keypoint.
  - a. Interpolation of nearby data: This step improves the matching and the stability of the keypoint.
  - b. Discarding the keypoints that are low in contrast: Low contrast keypoints offer higher sensitivity to noise and are unstable. Hence, they are discarded.
  - c. Elimination of edge responses: For stability enhancement, keypoints that have poorly determined locations are eliminated even though they show high responses to the edge.
3. **Orientation assignment:** Keypoints are assigned orientations which depend on the directions of the local image gradient. Due to this, the algorithm becomes invariant to rotation. Consider  $G(x, y)$  an image smoothed by Gaussian filter. It is taken at keypoint's scale  $\sigma$ . For an image sample  $\sigma$ , the magnitude of the gradient denoted by  $m(x, y)$  is given by

$$m(x, y) = \sqrt{a(x, y)^2 + b(x, y)^2} \quad (2)$$

Whereas the orientation  $\theta(x, y)$  is

$$\theta(x, y) = \tan^{-1} \frac{b(x, y)}{a(x, y)} \quad (3)$$

Where,

$$a(x, y) = L(x+1, y) - L(x-1, y)$$

and

$$b(x, y) = L(x, y+1) - L(x, y-1)$$

A histogram is created in which each 10 degrees is covered by a bin making it a 36 bin histogram. Dominant

orientations are indicated by the peaks that are present in the histogram.

4. **Keypoint descriptor:** The keypoint descriptor vector that is obtained has 128 elements. Even though the dimension is high, it is preferred because descriptors that have lower dimension give a poor performance.

The SIFT descriptors show invariance to affine changes that are on a minor scale. Fig. 3 describes the SIFT features extracted from the image.



Fig.3. SIFT features of the image; the colored circles depict the robust SIFT keypoints

## 2. Speeded Up Robust Features [SURF]:

In the usage of SURF, the concept of integral images is put to use. An integral image  $I_{\sum(x)}$  obtained at a location

$X = (x, y)$  is a representation of the summation of all pixels in the input image  $I$  that are present in a rectangular region obtained from the origin and  $X$ .

$$I_{\sum(x)} = \sum_{i=0}^{i \leq x} \sum_{y=0}^{j \leq y} I(i, j) \quad (4)$$

Once the computation of integral image is done, four additions are sufficient to calculate the summation of intensities present in any upright rectangular region. Moreover, the time taken for the calculation is independent of the area of the rectangle. Speeded-up Robust Features is a feature detector-descriptor, which is scale-invariant, and is based on the combined approach of Hessian Matrix and the Haar wavelet responses that are distributed within the neighborhood of the interest point [12]. The Hessian matrix forms the basis for the detector because it performs well for the computation of time and accuracy. Hessian-based detectors provide more stability and repeatability than Harris corner detectors. It uses DoG (Difference of Gaussian) which is responsible for bringing the speed at a low cost with a trade-off in accuracy [13]. In SURF, the selection of location and scale both are carried out by using the Hessian matrix determinant. For a particular point  $X = (x, y)$  present

in the image  $I$ , the Hessian matrix  $H(X, \sigma)$  in  $X$  at scale  $\sigma$  is given as follows

$$H(X, \sigma) = \begin{bmatrix} L_{xx}(X, \sigma) & L_{xy}(X, \sigma) \\ L_{xy}(X, \sigma) & L_{yy}(X, \sigma) \end{bmatrix} \quad (5)$$

Where,  $L_{xx}(X, \sigma)$  denotes the convolution of the Gaussian second order derivative  $\frac{\partial^2}{\partial x^2} g(\sigma)$  with the image  $I$  in point  $X$ , and similarly for  $L_{xy}(X, \sigma)$  and  $L_{yy}(X, \sigma)$ . For practical 2D applications, the Gaussian derivatives need to be discretized and cropped. SURF tests a simple alternative for avoiding this process. Gaussian derivatives are approximated by using LoG and Box filters the evaluation of which can be done very fast using integral images.

**SURF descriptor:** As is the case with all the descriptors, SURF also has to take care of the effects of photometric changes. The primary step is to fix an orientation which is reproducible depending upon the information obtained through the region surrounding the interest point. Construction of a square region is carried out which has an alignment to the selected orientation and the extraction of the SURF descriptor is carried out. The steps are described further in detail.

1. **Assignment of Orientation:** In order for the descriptor to be rotation invariant, an orientation that is reproducible is observed for the interest points. In order to accomplish this, calculation of Haar-Wavelet responses is carried out in x and y-direction and in a circular region with a radius of  $6s$  surrounding the interest point with 's' denoting the scale at which the interest point is present. Integral images are used for fast filtering. Only six responses are required for the computation of the response in x or y-direction at any scale. The wavelet responses are calculated and weighted with a Gaussian filter that is centered at the interest point. After that, the representation of responses is carried out as vectors in space with horizontal response strength denoted by the abscissa and the vertical response strength denoted by the ordinate. The estimation of the dominant orientation is carried out by undertaking the summation of all responses within a sliding orientation window covering an angle of  $\pi/3$ . The responses are summed and the new vector is obtained. The longest such vector assigns its orientation to the interest point.
2. **Descriptor components:** For the purpose of extraction of the descriptor, the primary step is to construct a square region that is centered at the point of interest and whose orientation is assigned according to the procedure explained in the previous step. The region is divided into sub-regions of a square shape. The response of the Haar wavelet in the horizontal direction is called  $dx$  and similarly that in the vertical

direction is called  $dy$ . These are with respect to the orientation of the selected interest point. For the increase of the robustness towards the localization errors and some geometric deformations,  $dx$  and  $dy$  are first weighted with the help of a Gaussian filter that is centered at the point of interest. The summation of the wavelet responses over  $dx$  and  $dy$  for each sub-region is carried out and this forms the first set of the feature vector. The absolute values of the Haar response  $|dx|$  and  $|dy|$  are also extracted for the information about polarity changes. Thus, the descriptor vector is given as follows. Note that this vector is four-dimensional. Fig. 4 describes the SURF features extracted from the image.

$$v = (\sum dx, y, \sum |dx|, \sum |dy|)$$



Fig.4. SURF features of the image; the colored circles depict the robust SURF keypoints.

### 3. *Oriented FAST and Rotated BRIEF [ORB]:*

ORB algorithm, proposed with a foundation of FAST and BRIEF algorithms, is a technique for the description of feature points by the usage of a binary string [7]. The detection of a feature point in ORB is carried out by making use of the improved FAST feature detection technique, and the descriptor is constructed using an improved BRIEF feature descriptor. As the foundation of this derivative is very high-speed techniques, ORB boasts of an absolute advantage in speed. The most attractive feature of this algorithm is its speed and rotational invariance. Another usable feature is that it has a reduced sensitivity to noise. Fig. 4 describes the ORB features extracted from the image. The steps applied for implementation of ORB algorithm are as follows.

1. The position of key points is obtained by using the concept of SUSAN detector in FAST algorithm.
2. By using Harris detector, N-best points are selected.
3. With the help of Scale-Pyramid transform, the descriptor is made scale-invariant.
4. Orientation is given with the help of Intensity Centroid.
5. Binary descriptors are extracted with the help of BRIEF algorithm. Steered BRIEF is obtained.

6. Low correlative pixel blocks are obtained by using greedy algorithm.
7. Thus, a final descriptor is obtained.



Fig.5. ORB features of the image; the colored circles depict the robust ORB keypoints.

Table 1. Implementation steps of feature detection techniques and their execution time

Feature Detector	Steps for Implementation	Execution time (seconds)
<i>SIFT</i>	<ol style="list-style-type: none"> <li>1. Scale-Space Extrema Detection</li> <li>2. Keypoint Localization</li> <li>3. Orientation Assignment</li> <li>4. Keypoint Descriptor</li> </ol>	15.11
<i>SURF</i>	<ol style="list-style-type: none"> <li>5. Scale-Space Extrema Detection</li> <li>6. Keypoint Localization</li> <li>7. Orientation Assignment</li> <li>8. Keypoint Descriptor</li> </ol>	2.43
<i>ORB</i>	<ol style="list-style-type: none"> <li>9. Scale-Space Extrema Detection</li> <li>10. Keypoint Localization</li> <li>11. Orientation Assignment</li> <li>12. Keypoint Descriptor</li> </ol>	6.54

#### 4. Bag of Features model:

Bag of features (BoF) or Bag of words (BoW) model can be used to classify the images by considering the whole feature as a word. Since the number of features detected from the image is not fixed, the number of features can be in thousands. So in order to limit the size of features to a limit Bag of features is created. The steps to create bag of words is as follows:

- Organize the various images according to their category and create separate sets for each category of the image.
- Read the image from database and extract its keypoints using certain feature based algorithm.
- Obtain the feature vectors from the obtained keypoints which are also known as descriptors and perform the clustering operation like K-mean clustering.
- From the vocabulary obtained by clustering, find the histogram and normalize it.

- Form a matrix to store such histograms of each image. This matrix is nothing but Bag of words.

#### C. Hardware system design

Hardware system design primarily consists of the creation of the Iot node from the web camera. For this purpose, the camera is made to communicate with Raspberry Pi which acts as an intelligent device for the system. The hardware arrangement of the system is shown as follows in fig. 6; Source of images: Camera<sup>1</sup>, Raspberry pi<sup>2</sup>, Router<sup>3</sup>, Monitor<sup>4</sup>.

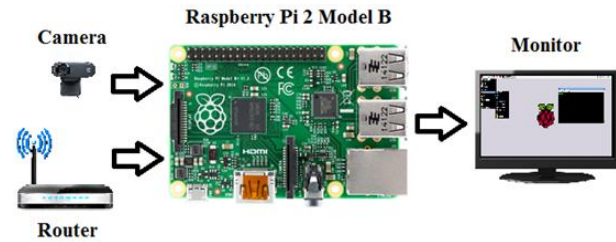


Fig.6. Hardware arrangement of the system.

The most important task accomplished in this module is the creation of an Iot node. For the purpose of creation of the node, first of all, Raspberry Pi was configured as per the requirement. The steps undertaken for the configuration are as follows.

- Installation of RASPBIAN JESSIE Operating System in Raspberry Pi.
- The configuration of Static IP Address for establishing a connection with the router.
- Programming was done in Python 3.4.1. This program for object recognition was run in OpenCV virtual environment which was previously created in RASPBIAN JESSIE.
- The information related to the created database was extracted from the internet.
- The information regarding the recognized object is displayed on the computer screen.

## IV. RESULTS

The above mentioned database is initially created for finding the best fit model of object recognition. The images from database are taken and SURF features are detected from them. The reason to choose SURF algorithm is its speed of recognition. The histogram from these SURF features is obtained for creating the Bag of words. For classification of objects Support vector machine is used with common error value  $C = 100$ , gamma  $g = 0.01$  and kernel as Radial Basis Function (RBF). These values are obtained by performing the grid-

<sup>1</sup> <http://www.logitech.com/assets/31886/c270glamourlgsilver.png>

<sup>2</sup> <https://upload.wikimedia.org/wikipedia/commons/3/31/RaspberryPi2Model>

<sup>3</sup>

<http://cdn.business2community.com/wpcontent/uploads/2015/03/Place-WiFi-Router.gif>

<sup>4</sup> <http://pngimg.com/upload/laptopPNG5897.png>

search operation with 5-fold cross-validation. The efficiency obtained is 99% taking 100 random states.

Table 2. Comparison of feature detection techniques on the basis of parameters: Approach, Building block and Scale Invariance

Feature Detector	PARAMETERS		
	Approach of Detection	Building Blocks	Scale Invariance
<b>SIFT</b>	Corner and Blob detection	Difference of Gaussian Detector	YES
<b>SURF</b>	Corner and Blob detection	Integral Images	YES
<b>ORB</b>	Corner Detection	FAST detector	YES

Implementation of SIFT is carried out in Python 3.4.1 with OpenCV 3.0.0. The Raspberry pi is used with raspbian OS for installation. For setting up particular versions of python and OpenCV, the virtual environment is created. Fig. 7 shows the result of the match found using SURF descriptor. The feature threshold taken is 5000. Time taken for execution is 15.11 sec. Fig. 8 shows the result of the match found by using SIFT descriptor having feature threshold equals to 5000. Time taken for execution is 2.43 sec. Fig. 9 shows the result of the match found using ORB technique. The feature threshold taken is 5000. Time taken for execution is 6.54 sec.

Table 3. Comparison of feature detection techniques on the basis of parameters: Rotation invariance and Repeatability accuracy and efficiency

Feature Detector	PARAMETERS	
	Rotation Invariance	Repeatability Accuracy and efficiency
<b>SIFT</b>	YES	Pretty good efficiency and robustness. (Better than FAST)
<b>SURF</b>	YES	Up to five times faster than SIFT, very robust and efficiently accurate
<b>ORB</b>	YES	As efficient as SURF. Robustness not as good as SURF.

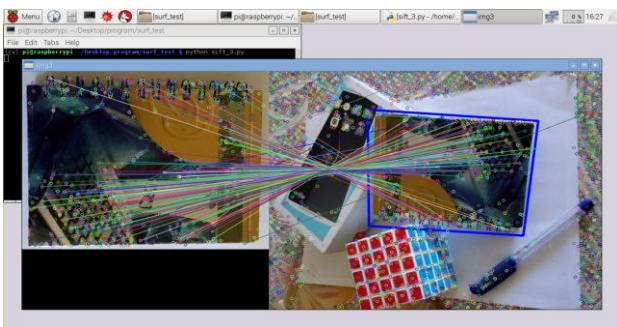


Fig.7. Object Detection using SIFT algorithm; Time of Execution: 15.11sec; Feature threshold: 5000.

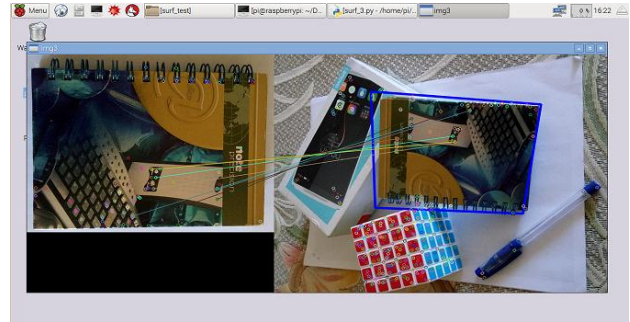


Fig.8. Object Detection using SURF algorithm; Time of Execution: 2.43sec; Feature threshold: 5000

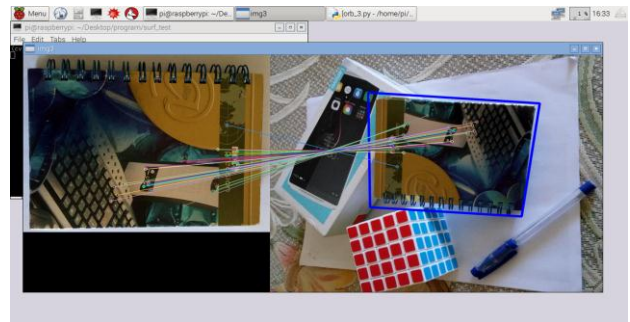


Fig.9. Object Detection using ORB algorithm; Time of Execution: 6.54sec; Feature threshold: 5000

In order to find the suitable method for object detection, a comparison of the algorithm is carried based on the change in rotation angle, intensity level shift and scaling factor. The created database is utilized for comparison. The procedure for comparison of algorithms is carried out in python 3.4.1 with openCV 3.0.0. The python script generates a .csv (comma separated values) file. It contains recognition rate values of a particular parameter which are then used by the Matlab R2013b to plot various graphs. Keypoints are first extracted from the image. These keypoints are used to match the features with the original version of the image. In order to find the recognition rate, the k-Nearest neighbor (kNN) method is used as a matching strategy with the distance ratio taken as 0.7. The generalized Hamming distance is then calculated which gives the most accurate matching [14]. Fig. 10 shows the graph plotted by change in rotation angle of the image to the change in recognition rate of different algorithms that can be used for object recognition. From this graph, it can be clearly observed that due to optimized number of keypoints SIFT perform better than SURF as well as ORB on rotation of the object. Fig. 11 shows the graph, Intensity shift versus Recognition rate. The intensity level shift is carried from a range of -150 to 150 for the unsigned 8-bit binary-valued image. ORB performance for intensity change is better than other two because the variation in recognition rate is less. Fig. 12 gives the graph between the scale changes to the recognition rate. The scaling factor variation gives almost the same result. Since the overall performance of SURF is good considering recognition time as an important parameter for real-time implementation, SURF is chosen as a method for object recognition.

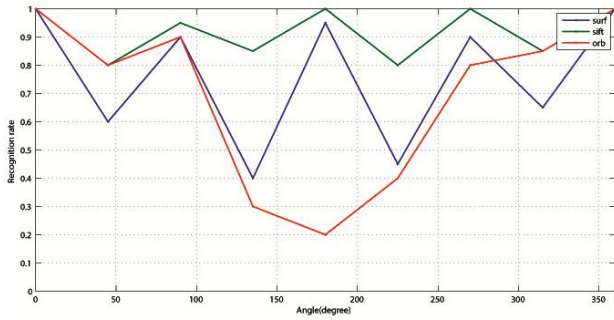


Fig.10. Rotation versus Recognition rate graph for SURF, SIFT and ORB methods.

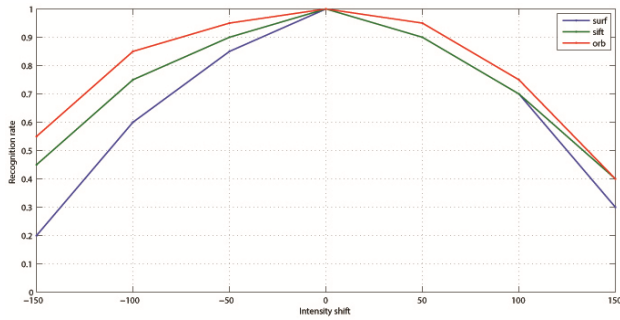


Fig.11. Intensity variation versus Recognition rate graph for SURF, SIFT and ORB methods.

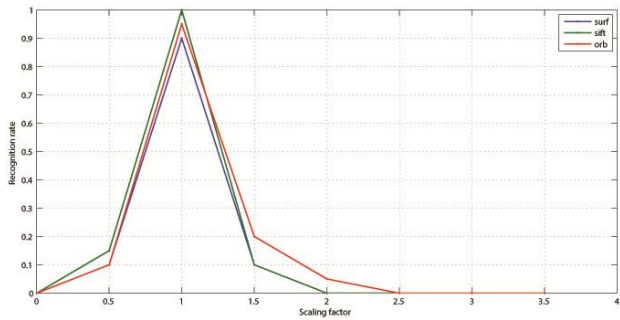


Fig.12. Scaling factor versus Recognition rate graph for SURF, SIFT and ORB methods.

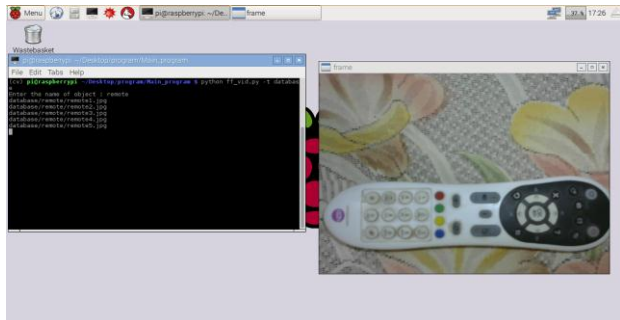


Fig.13. Real time database creation.

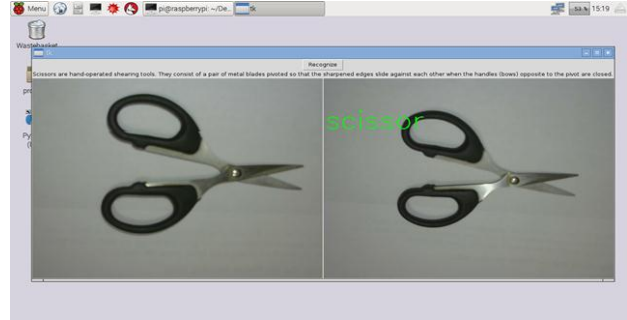


Fig.14. Recognition of Object and Extraction of its information from Internet.

To enable the addition of more objects in the database, a real-time database creation python script is written. The user has to enter the name of the object and then the object is captured from the camera and it will be added in the database.

The camera captures the image in real time. The object is detected and recognized by using the SURF algorithm with 'Bag of word' technique. The information about the recognized object is displayed in a few lines. This information is obtained from the internet. Hence, the camera is said to be the IoT node. The process of recognition of an object in an image is illustrated in the image below. The object is recognized from the database that is created beforehand for this purpose.

### V. CONCLUSION

The proposed system is a highly efficient one for object detection and recognition. Moreover, having been installed in a portable intelligent device, the system is quite easy to carry around. The SURF algorithm that is used for object recognition has proven to be very efficient in real-time object recognition. The key factor of IoT paradigm is using the web to obtain information without human interference. This objective is achieved in the proposed system. The internet connection that is demonstrated in this system established through Ethernet. A wireless connection would highly improve its mobility. The work undertaken is successful in creating an Internet of Things node from a camera which connects to the internet via Raspberry Pi as the intelligent device. The objects in the image are detected via SURF algorithm. This algorithm has known to outperform the rest giving a high recognition efficiency. The GUI is able to display the information about the recognized object from the information sites available on search engines, for example, Wikipedia. Other outputs such as audio can be added into the system. This system would find its applications into various domains. However, if implemented on a commercial basis, the data cannot be stored on a local device due to the large amount. It has to be stored on the cloud. The larger the database more is the time required to sift through it. Hence, a grid creation is advisable.



## ACKNOWLEDGMENT

I would like to express my deep gratitude towards my Head of Department, Dr. M.B. Mali, department of Electronics and Telecommunication SCOE, Pune for his encouragement for this paper. I earnestly thank all those who helped us in making this a significant achievement. I am grateful to Er. Shrey Kumar Jain M.Tech Computer Science, SGSITS Indore. I also thank all my friends for their help and support.

## REFERENCES

- [1] M. C. Domingo, "An overview of the Internet of Things for people with disabilities," *J. Netw. Comput. Appl.*, vol. 35, no. 2, pp. 584–596, 2012.
- [2] L. Atzori, A. Iera, and G. Morabito, "From 'smart objects' to 'social objects': The next evolutionary step of the internet of things," *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 97–105, 2014.
- [3] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context Aware Computing for The Internet of Things: A Survey," *IEEE Commun. Surv. TUTORIALS*, vol. X, no. X, pp. 1–41, 2013.
- [4] T. Quack, H. Bay, and L. Van Gool, "Object Recognition for the Internet of Things," *First Int. Conf. Internet Things (IoT 2008)*, vol. 4952, pp. 230–246, 2008.
- [5] S. Keypoints and D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
- [6] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," *Lect. notes Comput. Sci.*, vol. 3951, p. 14, 2006.
- [7] E. Rublee and G. Bradski, "ORB: an efficient alternative to SIFT or SURF."
- [8] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3951 LNCS, pp. 430–443, 2006.
- [9] T. Lindeberg, "Scale-space: A framework for handling image structures at multiple scales," pp. 1–12, 1996.
- [10] A. P. Witkin, "Scale-Space Filtering," in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 1019–1023, 1983.
- [11] T. Lindeberg, "On the axiomatic foundations of linear scale-space: Combining semi-group struct with c vs. scale invariance," 1994.
- [12] K. Mikolajczyk and C. Schmid, "Scale & Affine Invariant Interest Point Detectors," vol. 60, no. 1, pp. 63–86, 2004.
- [13] K. Mikolajczyk, K. Mikolajczyk, C. Schmid, and C. Schmid, "A performance evaluation of local descriptors,"

*IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.

- [14] Tian Tian, Ishwar Sethi, Delie Ming and Nilesh Patel, "A Zoned Image Patch Permutation Descriptor," *IEEE Signal Processing Letters* 22(6), May 2015.
- [15] Quanqing Xu, Khin Mi Mi Aung, Yongqing Zhu, Khai Leong Yong, "Building a large-scale object based active storage platform for data analytics in the internet of things," *Springer Science J Supercomput*, 2016 DOI 10.1007/s11227-016-1621-2.

## Authors' Profiles



**Mr. Shreddhey Kumar Jain** is from Sagar, Madhya Pradesh and was born on 23 May, 1993. He earned his Bachelor's degree from Walchand Institute of Technology, Solapur. His main areas of interest are VLSI and Image processing & computer Vision.

He is currently pursuing his Masters' Degree in Electronics Engineering (Digital Systems) from Sinhgad College of Engineering, Pune. He has published a research paper on the partial fulfillment of his Master's thesis entitled "Performance Analysis of Computationally Efficient Model Based Object Detection and Recognition Techniques" in International Journal of Science Technology and Engineering.

Mr. Jain has been an engaging student during his post graduate and graduate studies. He has participated in and won various technical and non-technical competitions. His awards have been mostly for his programming skills. He has participated in workshop on the Intel Galileo processor organized jointly by Intel and Sinhgad College. He has also been certified by IIT Powai for his excellence in C programming, SciLab and LaTeX.



**Mrs. Supriya O. Rajankar**, female, is Associate Professor in Department of Electronics and Telecommunications Engineering at Sinhgad College of Engineering, Pune. She earned her master's degree from Government College of Engineering, Pune. She is a member of ISTE and IETE. She has a credit of being an author of two books published by John Wiley Publication. Her main area of interest is in the field of Signal Processing and Image Processing. She is currently working on a research project in the field of Signal Compression.

**How to cite this paper:** Shreddhey Kumar Jain, Supriya O. Rajankar, "Real-Time Object Detection and Recognition Using Internet of Things Paradigm", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.9, No.1, pp.18-26, 2017.DOI: 10.5815/ijigsp.2017.01.03