***International Advance Journal of Engineering Research (IAJER)***
*Volume 3, Issue 5 (May- 2020), PP 01-11*
*ISSN: 2360-819X*
***www.iajer.com***

**Research Paper** **Open Access**

# Object Detection Techniques for Mobile Robot Navigation in Dynamic Indoor Environment : A Review

Anteneh Tilaye[1], Rajesh Sharma[2], Yun Koo Chung[3]
*[1]Lecturer, [2]Assistant Professor, [3]Professor*
*[1,2,3] Department of Computer Science and Engineering Adama science and Technology university*

**ABSTRACT:-** The detection and recognition of objects is very important and challenging task in computer vision, as there is an increasing interest in building autonomous mobile system. To make mobile service robot truly ubiquitous to complex and dynamic indoor environments, they should be able to understand the surrounding environment beyond the ability to avoid obstacles, autonomously navigate, or build maps. Several researchers have proposed different techniques for recognizing and localizing indoor objects for mobile robot navigation. Different object detection algorithms from the early computer vision approaches until recent deep learning approaches are reviewed and compared. Based on the type of feature used the algorithms are classified in to two. The first class is based on a local feature like SIFT, SURF, etc. this class also includes the techniques that fuse these local features with 3D or Depth images. The second class is based on deep features, using deep neural networks for detecting objects in the images which id further classified into two based on whether these algorithms use one or two stages for detecting the object. Object detection for mobile robot navigation can be used for Assisting an elderly person or person with disability (Blind) to navigate in indoor environment, Home security and surveillance.

**Keywords:** *Object Detection, Deep Learning, Object Localization, Mobile Robot Navigation, Semantic Navigation.*

## I. INTRODUCTION

The Rapid evolution of technology and the development of robotic applications have made possible to create autonomous robots to assist or replace humans in ordinary tasks in an everyday setting. For instance, self-driving cars and autonomous robots are one of the hottest research areas right now. Service robots should coexist with humans in the same environment to provide support and assistance in housework and caring for elderly or disabled people. Service robots usually work in an environment that is unstructured, where they collaborate with humans to accomplish tasks, but industrial robots are usually fixed in an environment that is structured and has external safeguards to protect them [1]. These service robots should be able to build an internal representation of the environment and localize itself, have a set of capabilities that allows them to understand, interact and navigate in real environment and understand commands from humans through various methods.

Vision is the most important sense for navigating or moving and interacting with the environment safely. Robots must have the ability to process visual data in real-time to adapt to the change to the environment. So for a robot vision, the detection and recognition of an indoor object is one of the important research topics in computer vision. To make mobile service robots truly ubiquitous to complex indoor environments, they should be able to understand the surrounding environment beyond the capability of avoiding obstacles, navigating autonomously, and building maps. It's necessary to build a reliable and fast object detection system to enhance the performance of indoor robot navigation. In the past decade, there have been great advances in this area, though this issue still remains one of the most challenging problems in computer vision when a real-life scenario is considered.

There are successful commercial service robots currently available on the market like Roomba and Scooba [2], for simple vacuuming or sweeping the floor. These mobile robots can evolved by adding more complex tasks. This research paper reviews different approaches or techniques of object detection for mobile robot navigation in dynamic indoor environment including the general object detection algorisms. Different object detection challenges and application areas are also discussed. This paper is organized as follows Section II gives brief introduction to object detection and general purpose object detection. Section III discusses Mobile robot navigation. Section IV Review and comparison of different papers on Object Detection. Section V challenges in Object Detection and application. Section VI conclusion.

# 1 Object Detection

An object recognition algorithm identifies which objects are present in an image, on the other hand, an object detection algorithm not only tells you which objects are present in the image but also outputs bounding boxes (x, y, width, height) to indicate the location of the objects inside the image. Different approaches have been proposed over the past decades for detecting objects in an image or video. Those techniques can be classified into two based on the kind of features used. The first class is based on a local feature like SIFT, SURF, etc. this class also includes the techniques that fuse these local features with 3D or Depth images. The second class is based on deep features, using deep neural networks for detecting objects in the images. This class can further be classified into two based on whether these algorithms use one or two stages for detecting the object. Those techniques are discussed in brief detail on the following subsections.

## 1.1 Object Detection Based On Local Features

In Local feature-based object detection, one or more features are extracted from the image and the object is modeled by using these features. The object of interest is represented by the shape, size or color of the object. This class can be further classified based on the type of local feature used.

### 1.1.1 Color-Based Object Detection

An object of interest is represented by its color information. Using color information for detecting objects in an image is not always appropriate. The feature set can be built using color histograms, gradient orientation histogram, edge orientation histogram, the properties of HSV color space and SIFT or SURF descriptors.

Zhenjun Han et al. [3] proposed an object tracking algorithm by combining color histogram (HC) bins and gradient orientation histogram (HOG) bins which consider the color and contour representation of an object respectively. S. Saravanakumar et al. [4] represented the object of interest by the properties of HSV color space. An adaptive k-means clustering algorithm was used to get color values centroid of the object

The most popular feature detector and descriptors are FAST, BRIEF, ORB, SURF, SIFT, and others for object detection and tracking. FAST (Features from Accelerated Segment Test) [5] is a computational efficient feature detector, which selects interest point by considering the intensity of 16 pixels circled around the pixel under test. If 8 pixels around the pixel under test are darker or lighter than that pixel, then it’s selected as a key-point or interest point. FAST does not include orientation and multiscale feature. BRIEF (Binary Robust Independent Elementary Features) [6] is a fast feature descriptor which outperforms SIFT and SURF feature descriptor in terms of speed and recognition rate in many cases. BRIEF takes S by S patch around the key-point and creates a binary vector of size n (could be 128, 256 and 512). In the binary feature vector, 0 or 1 is set depending on whether the intensity of X is greater than the intensity of Y on a randomly selected pair (x, y).

ORB (Oriented FAST and Rotated BRIEF) [7] is an alternative to SIFT and SURF since SURF and SIFT are patented one should pay to use them for commercial use. It uses FAST feature detector with BRIEF feature descriptor. ORB modifies FAST feature detector to overcome orientation and multiscale problems. ORB is scaled invariant using a multiscale image pyramid to detect a key-point at a different scale. ORB uses Rotation-aware BRIEF to make it rotation invariant. It does this by steering BRIEF according to the orientation of the FAST key points. Another most popular algorithms are SIFT (Scale Invariant Feature Transform) [8] and SURF (Speeded up Robust Features) [9].

SIFT has four steps to detect and describe key points in the image. It finds potential key-point using the Difference of Gaussian by getting the Gaussian blurred image at different scale and finding the local extrema over scale and space by comparing each pixel with its 8 neighbors and 9 pixels in the next and previous scale. This local extremum is further refined by applying threshold value and 2x2 Hessian matrix to remove the edges [8]. Each key point is assigned an orientation to make it rotation invariant by making an orientation histogram with 36 bins covering 360 degrees. Lastly, 128 bin key point descriptor for 16 sub-blocks of 4x4 size having 8 bin orientation histogram is created [8].

SURF is created to speed up SIFT. SURF [9] uses Box filter instead of a Difference of Gaussian to approximate the Laplacian of Gaussian. Since convolution can be calculated with the help of an integral image, box filters can be convolved in parallel at a different scale. It uses the horizontal and vertical direction wavelet responses with Gaussian weight in the neighborhood of size 6 to assign orientation. Again for key-point description, SURF uses Wavelet responses in a horizontal and vertical direction with a neighborhood of size 20x20 around the key-point. SUFT feature descriptor has two versions, one with 64 dimensions and the other with 128 dimensions.

### 1.1.2 Shape-Based Object Detection

In Shape-based object detection, the object is represented by its shape. Usually done by extracting the contour of the object from the image. For extracting the contour RGB image or Depth images might be used. Huabo Sun et al. [10] proposed an object detection algorithm which detects edges with Canny method and extracts the contours of the object at different image resolution. L. Lu et al. [11] presented the representation of objects by HOG and PCA. The object is first transformed to the grids of Histograms of Oriented Gradient (HOG) descriptor and then apply Principal Component Analysis (PCA).

### 1.1.3 Template-Based Object Detection

Template-based object detection is usually done by matching the features between the template image of the object of interest and the image from the scene. This technique requires a template describing the object. This template can be fixed or deformable. It is a fixed template when the object shape does not change from a different viewing angle of the camera. For fixed template matching can be done using image subtraction or correlation between the image from the scene and the template. Correlation is immune to noise and illumination effects in the images, whereas image subtraction should be done in a restricted environment. Template matching on deformable objects can be performed by applying parameterized deformation transform on the prototype (also called prototype-based template) [12].

### 1.1.4 Motion-Based Object Detection

The most common motion-based object detection techniques are thresholding technique over the inter-frame difference, Optical Flow and Gaussian Mixture. Gaussian Mixture models each value of a pixel by using a mixture of Gaussian for background/foreground segmentation.

### 1.2 Object Detection Based On Deep Features

Object detection ideas begin by searching region on the image and performing classification on detected regions. Over the recent years, most of the state of art object detections are based on one-shot detection, where the detect objects in the image through one pass. Generally, object detection based on deep features can be classified into two by whether these algorithms use object proposal or perform one-shot detection.

### 1.2.1 Object Detection Based On Object Proposal

Sub-regions (patches) of the image is selected first and then apply the object recognition algorithm to these image patches to detect objects. The location of the objects is given by the location of the image patches where the class probability returned by the object recognition algorithm is high. The straightforward approach to select patches is a sliding window, where we Crop multiple images by sliding window and pass each cropped image through ConvNet, but it's computationally very expensive.

This problem is solved by Object proposal algorithms. There are several object proposal algorithms like objectness measure [13], selective search [14] (used in R-CNN and Fast R-CNN), Binarized Normed Gradients (BING) [15], etc. Selective Search is one of the most popular region proposal algorithm used in object detection. It is based on computing hierarchical grouping of similar regions based on color, texture, size, and shape compatibility [14]. Selective Search starts by over segmenting the image based on the intensity of the pixels using a graph-based segmentation method by Felzenszwalb and Huttenlocher [16]. At each iteration, larger segments are formed and added to the list of region proposals. Among object proposal algorithms Selective search is designed to be fast with very high recall. Binarized Normed Gradients (BING) [15] is another object proposal technique by Cheng et al. which is based on the observation that generic objects with well-defined closed boundaries can be discriminated by looking at the norm of gradients and it's the fastest.

It resizes the image to 8 by 8 and uses the norm of gradient as a sample 64D feature to describe it for training a generic abjectness measure. This is further binarized for efficient objectness estimation since it only requires a few atomic operations (ADD, BITWISE SHIFT, etc.). It can run 300fps on singe CPU laptops yielding a 96.2% object detection rate with 1000 proposals. As the number of proposals and color space increases, the detection rate also increases (99.5%). It's 1000 times faster than most popular alternatives selective search [14], Category-Independent Object Proposal [17], objectness measure [13].

There is a state of art general object classification and localization deep learning models like **R-CNN** (Region-Based Convolutional Neural Network), **Fast R-CNN**, **Faster R-CNN**, which are based on object proposal for localizing the object. Region-Based Convolutional Neural Network (R-CNN) [18] perform region search on the image using selective search and classifies each proposed region to one of the classes. It starts with small regions and hierarchically merges them to form a bigger region according to a variety of color spaces and similarity metrics [18]. After performing a selective search the output will be region proposals (~2k) which could contain the objects of interest. Each proposed region will be fed into the CNN model by resizing it so that the patch (region) will match the input of the model. The CNN will extract a 4096-dimension vector of features, which are fed into multiple SVM classifiers to classify these regions to one of the classes by producing a probability that it belongs to each class.

R-CNN was able to achieve a 62.4% mAP score for PASCAL VOC 2012 test dataset and a 31.4% mAP score over the 2013 ImageNet dataset. Linear regression is used to modify the coordinates of the bounding box. R. Girshick [19] introduced Fast R-CNN to reduce the time consumption that is imposed on the R-CNN. Fast R-CNN extracts the feature using CNN from the entire image instead of applying CNN on each proposed region. The object proposal algorithm is applied to the feature map produced by CNN. ROI pooling layer is used to resize the feature map to a valid region of interest (ROI) with fixed height and width as hyperparameters. Each ROI is fed into fully connected layers, followed by a Softmax classifier. Fast R-CNN also uses linear regression to modify the bounding box.

Fast R-CNN achieved a 70.0% mAP score over the 2007 PASCAL VOC test dataset, 68.8% for the 2010 PASCAL VOC test dataset and 68.4% for the 2012 PASCAL VOC test dataset. Faster R-CNN is proposed by Shaoqing Ren et al. [20], which uses Region Proposal Network (RPN) instead of a computationally expensive selective search. The entire image is fed into a pre-trained (on ImageNet) CNN model to extract the feature and RPN proposes a maximum of k regions. The classification and bounding box prediction on the proposed is done by using two fully-connected layers. Faster R-CNN is just the combination of Region Proposal Network and Fast R-CNN, in which Faster R-CNN replaces the selective search by RPN. Fast R-CNN obtained a 78.8% mAP score over the 2007 PASCAL VOC test dataset and 75.9% over the 2012 PASCAL VOC test dataset.

Faster R-CNN 34 times faster than the Fast R-CNN by using RPN. All the above models use object proposal algorithms like selective search and RPN (region proposal network) to localize objects. Although these models have high accuracy (achieved promising mean Average Precision (mAP)), they are computationally expensive, since they have to run classification for every proposed region.

### 1.2.2 One-Shot Object Detection

YOLO (You Only Look Once) [21] and SSD (Single Shot Detector) [22] predicts bounding boxes and class probabilities with a single network in a single evaluation instead of proposing objects and classifying each proposed window.

YOLO takes the entire image as input and divides it into S x S grid. Each cell in the grid predicts B bounding boxes with a confidence score. The confidence score is the probability that there is an object multiplied by the IOU (intersection over union) between the predicted and ground truth bounding box. The output of the final layer is a S x S x (C + B x 5) tensor corresponding to each cell the grid [21]. S represents the size of the grid cell and C is the number of estimated class probability. B is the number of anchor boxes, each having 4 coordinates and 1 confidence value. YOLO preforms non-maxima suppression at the end of the network to merge highly overlapping bounding boxes. YOLO has the total number of convolutional layers are 24 followed by 2 fully connected layers. It also have a less accurate and fast version with 9 convolutional layers and fewer filters [21].

YOLO achieved a 63.7% mAP score over the 2007 PASCAL VOC dataset and a 57.9% mAP score over the 2012 PASCAL VOC dataset. W. Liu et al. [22] have developed a Single Shot Detector (SSD) that is similar to YOLO. SSD also predicts the bounding boxes and class probability in a single shot with end to end CNN architecture. SSD uses feature maps at different positions of the network to predict the bounding boxes. The image will be passed through different convolutional layers which are having different sizes of filters. 10 x 10, 5 x 5, 3 x 3 filter sizes are used in SSD, whereas YOLO use 1 x 1 and 3 x 3 filters. To generate the bounding boxes SSD uses extra feature layers (convolutional layers with 3 x 3 filters) at different positions of the network [16]. They have obtained mAP scores of 83.2% over the 2007 PASCAL VOC test dataset and 82.2% over the 2012 PASCAL VOC test dataset [22].

J. Redmon et al. [23] introduced a second version of YOLO in order to increase accuracy while making it faster. Accuracy improvements are made by using batch normalization instead of dropouts, the high-resolution classifier (448 x 448 picture), convolution with Anchor Boxes, removing fully connected layers, Fine-Grained features where is used feature maps of different layers by reshaping them to the same dimension [23]. Speed improvements are achieved by replacing VGG16 by customized GoogLeNet (requires less operation) and using DarkNet to further simplify the backbone CNN used [23]. YOLOv2 achieved 78.6% mAP on VOC 2017. YOLOv2 is replaced with a more accurate and faster version called YOLOv3. YOLOv3 [24] replaced softmax function with independent logistic classifiers to calculate the class probability and uses binary cross-entropy loss for each label. They also introduced Feature Pyramid Network (FPN) like Feature Pyramid. YOLOv3 makes predictions at 3 different scales. It processes images at 30 frames per second (FPS) and has an mAP of 57.9% on the COCO test-dev on Pascal Titan X.

These models achieved promising mean Average Precision (mAP) and they can run real-time on computationally demanding machines [23]. Mobile robots do not come with heavy computing power so the detection algorithm response time should be sufficient enough that the robots can make a decision fairly quick [25]. The algorithm to be developed should work under these specifications.

## II. OBJECT DETECTION FOR MOBILE ROBOT NAVIGATION

A mobile robot is a robot that is capable of moving around the environment and not fixed or stationed to one physical location. Mobile robots have the following functional characteristics.

- **Mobility**: it should have total mobility relative to the environment (land, air, water).
- **Perception ability**: it should have the ability to sense and react to the environment.
- **A certain level of autonomy**: there should be limited human interaction.

One of the categories that can be taken as a mobile robot is service robots. Service robots are autonomous and mobile agents designed to assist or give service to humans in order to perform an everyday task in a domestic environment. The environment that human lives are meant for humans and service robots are expected to work in the same environment. Service robots need certain characteristics in order to provide these supports.

- It should be able to build an internal representation of the environment and localize itself in that environment.
- It should be able to navigate through the environment.
- It should be able to plan a path and what to do under different scenarios.
- It should understand and interact with the environment it resides in.
- It should interact and understand commands from humans through different means.

Perceiving the environment can be done with different sensors. The most important sensor to understand and navigate through the environment is a vision. The following subsections discuss more in a brief detail on mobile robot navigation for service robot navigation and localization and object detection for understanding the environment.

### a. Mobile Robot Navigation

For mobile robots, the ability to navigate in the surrounding environment is important. The robot should avoid situations like collusion, unsafe conditions, but also should accomplish its purpose to navigate in the surrounding robot environment. Robot navigation is the ability of a robot to reach a desired location with the ability to position itself in the current environment and plan a path to the desired location. Mobile robot navigation is can be defined as a combination of three fundamental components.

I. **Self-localization** is the ability to establish or realize its own location and orientation within the frame of reference or coordinate.

II. **Path planning**: is the realization of the current location and destination of the robot and planning how to navigate to the destination from the current location within the same frame of reference or coordinate. The robot should find the best route and navigate to the destination.

III. **Map building and interpretation**: is the ability of the robot to build and interpret the notation or map describing locations in the robot frame of reference. A map is the representation of the robot environment, in which the robot can refer to understand the environment layouts and locations.

Some mobile robot navigation systems have the ability to perform Simultaneous Localization and Mapping (SLAM).

### b. Simultaneous Localization and Mapping (SLAM)

SLAM deals with the problem of mobile robot navigation or building a map of an unknown environment while at the same time navigating the environment using the map and localizing itself [26]. It's a process where a mobile robot builds its own map and realizes its current location simultaneously while navigating the unknown environment. The trajectory of the platform and the location of the landmarks are estimated online without any prior knowledge of the environment. SLAM consists of different parts like Landmark extraction, data association, state estimation, state and landmark update. SLAM has many different steps and these steps can be implemented using a number of different algorithms [26]. The outline of the SLAM process is given below.

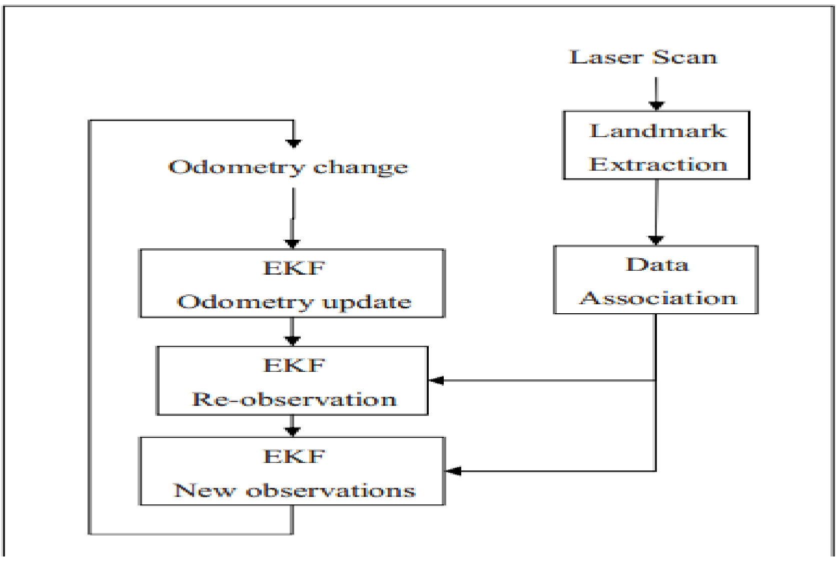

**Figure II.1 Outline of the SLAM Process**

**i. Odometry Data**

Odometry data is an approximate position of the robot measured by the movement of the wheels of the robot. Odometry data is an initial guess of where the robot might be in. we cannot use the odometry of the robot directly as it is often erroneous. To correct the position of the robot, laser scanners can be used to scan the environment.

**ii. Landmark Extraction**

We can use the laser scans of the robot environment to extract the features from the environment and re-observe when the robot moves around [26]. The feature extracted from the environment is called landmarks. These features are used to find out where the robot is or for a robot to localize itself.

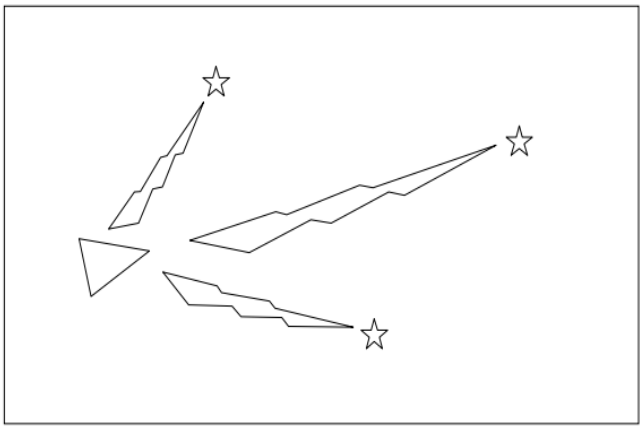

**Figure II.2 Landmark Extraction**

In the above diagram, the triangle is a robot and stars represent landmarks. The lightning represents the sensor measurement, where the robot measures the location of the landmarks by using its sensors. Characteristics of landmarks include it should be easily re-observable from different positions and angles. Individual landmarks should be distinguishable or unique from each other. It should be plentiful in the environment so that the robot doesn’t get lost. Landmarks should be stationary. There are multiple ways to do landmark extraction depending on the type of landmarks and sensors used. For example, spike landmarks are used for non-smooth surface and RANSAC is used for a smooth surface.

**iii. Data Association or Re-Observing Landmarks**

The extracted landmarks should be stored in a database, the landmark should be observed N times before it’s stored. After performing a new laser scan and extract the landmarks, each extracted landmarks are associated with the closest landmarks in the database. The nearest-neighbor approach can be used to associate a landmark with the nearest landmark in the database.

**iv. State Estimation and Landmark Update**

State or position estimation is done by Extended Kalman Filter (EKF) from the odometry data and landmark observations [26]. An EKF is the heart of the SLAM process. The EKF keeps track of an estimate of the uncertainty in the position of the robot and also the uncertainty in these landmarks it has seen in the environment [26]. After the landmark extraction and the data association, the SLAM process can be considered as three steps:

- Update the current state estimate using the odometry data. When the robot moves to the new position it will be updated using odometry update.
- Update the estimated state from re-observing landmarks. The extracted landmarks will be associated with the observation of landmarks it previously has seen. The re-observed landmarks are used to update the position of the robot in the EFK.
- Add new landmarks to the current state. The landmarks which have not been seen are added as a new observation.

## c. Related Works

Mobile robots do not have heavy computing power, so in order to make the robot move fairly quick, the response time to detect an object should be sufficiently efficient [25]. The object detection algorithm to be developed should work under these specifications. Various object classification and detection systems have been proposed over the last years. A most common approach is using a local feature of an RGB image to represent object features. H. Lee et al. [27] used structural features of pillar and hallway from RGB images with a decision tree for recognition and they did correlation with template mask to detect the pillar corner point.

Takács et al. [1] Applied Speeded Up Robust Features (SURF) Feature Detector and Descriptor with Bag of Words (BOW) and Support Vector Machine (SVM) for recognition of indoor objects (such as chair, fire extinguishers, and trash can), and they did SURF Localization on the classification result to localize the object in the frame. H. Jabnoun et al. [28], [29] used SIFT Feature and SIFT localization for identifying daily life objects. N. Diriba [30] proposed a sign detection algorithm for robot navigation using ORB and SURF, where the feature is matched to detect signs in the scene image. W. Obaid et al. [31] took color histogram of the patch for detection and SIFT feature matching between the model and the patch.

Some researchers [25], [32]–[34] applied depth information and 3D information for the segmentation of the object from the scene. In [32] Feature is extracted from depth image and RGB image and fed to SVM for classification, the techniques are designed specifically for every 3 objects. Hernandez et al. [33] added uncertainty calculation on top of [32] work. In [14] Point Cloud or 3D data is used to segment a horizontal plane and detect the object placed on it and they explore 2D classification algorithms with SURF and Scale Invariant Feature Transform (SIFT) features. Astua et al. [25] used two methods, contour extraction, and FLANN Matching, to segment the object and the size of a contour and SURF features are used with correlation and FLANN to classify the Images.

Recently, deep learning also gained increasing attention in the computer vision area for recognition, detection and Image segmentation. X. Ding et al. [35] proposed a Convolutional Neural Network (CNN) architecture with a selective search for object proposal and Detection Fusion to refine indoor object recognition for indoor object recognition. Y. chang [36] also proposed a Faster R-CNN based algorithm for object detection for ROS based mobile robots using GPU-accelerated computing. The deep learning approaches presented above do not have real-time speed.

## d. Comparision of Related Works

The following tables illustrate previous works on object recognition and localization. The selected papers are from 2014 onwards.

**Table II.1 Previous Works on Object Recognition and Localization Part 1**

| | **Object Recognition to Support Indoor robot navigation, 2015 M. Takács et al.** [1] | **Object Classification in Natural Environments for Mobile Robot Navigation, 2016 Hernandez et al.** [32] | **Object Detection Techniques Applied on Mobile Robot semantic Navigation, 2014 C. Astua et al.** [25] |
|---|---|---|---|
| **Dataset (Data collection)** | 244 Images, 8 classes taken by 640 x 480 resolution Kinect sensor. | Depth with RGB images, 3 objects, taken by ASUS Xtion Pro Live | Depth images, taken by Kinect sensor |
| **Preprocessing or enhancement** | No preprocessing is used to enhance the image | Equalization, morphological operations, Gaussian filter, and thresholding. | equalization, morphological transformations |
| **Segmentation or object proposal** | No segmentation techniques is applied. SURF localization is used. | Contour extraction, Hough transform, and watershed.<br>A different technique for each object. | Two methods,<br>Contour extraction<br>FLANN Matching |
| **Feature extraction** | SURF.<br>Bag of visual Feature | Geometric features,<br>Closet – solidity, extent, | Size of a contour for the first method, |

| | | | |
|---|---|---|---|
| | (BOW) is used to create a codebook. | circularity, handle ratio.<br>Chair – circularity.<br>Screen – circularity, extent and aspect ratio. | Speeded Up Robust Feature (SURF) for the second method<br>Combination of both. |
| **Classification** | Support Vector Machine (SVM) | Support Vector Machine (SVM).<br>Two Approaches - One against all and one against one. | Correlation for the first method.<br>Fast Library for Approximate Nearest neighbor (FLANN) for 2nd |
| **Evaluation** | 85 % accuracy | 1st - (81.58%) closets, chairs (72.79%), screens (65.60%)<br>2nd - 81.97%, 76.56%, 60.13% | Not very computationally demanding |
| **Limitations** | The system does not recognize multiple objects in the same frame.<br>Low recognition rate. | Low recognition rate.<br>A small number of objects.<br>The techniques are not generalizable. | Efficiency depends on How the robot moves.<br>Position and the distance of the robot from the object. |

**Table II.2 Previous Works on Object Recognition and Localization Part 2**

| | **Object recognition for vision-based navigation in an indoor environment without image database, H. lee et al. 2014** [27]**.** | **Indoor Object Recognition Using Pre-trained Convolutional Neural Network, X. Ding et al. 2017** [35]**.** | **Visual substitution system for blind people based on SIFT description, H. Jabnoun et al. 2014** [28]**.** |
|---|---|---|---|
| **Dataset (Data collection)** | 24 RGB images captured smartphone camera (640x360), 4 classes.<br>No image database | Public indoor dataset (18 categories) and private FoV (17 categories). | Private video frames, the number of classes are not specified. |
| **Preprocessing or Enhancement** | Edge extraction | Scaled (to 256x256). Since Caffenet expects with that size. | All images are converted to grayscale. |
| **Segmentation or Object Proposal** | Correlation with template mask to detect the pillar corner point. | Selective search method is used to generate ROI with bounding boxes. | No localization of an object. |
| **Feature Extraction** | Structural features of the pillar, hallway and hallway entrance. | CNN pipeline is used to extract features | SIFT (Scale Invariant Feature Transform) |
| **Classification** | Decision tree.<br>Comparing metrics of four features (pillar, hallway, hallway entrance) | CNN by using Caffenet as a reference model. | SIFT features of the target image are matched with the database |
| **Evaluation** | From 24, 17 images are well recognized<br>Recognition rate -70.8%<br>50% pillar, 25% entrance, 83.3% hallway, 90% absence. | Mean average precision (mAP) of 84.2%.<br>Detection fusion is used to reduce misclassification. | The evaluation technique is not stated. |
| **Limitations** | Low recognition rate<br>Not robust when rotation, obstacles | The developed algorithm does not run in real-time. | The system does not recognize multiple objects in the same frame. |

**Table II.3 Previous Works on Object Recognition and Localization Part 3**

| | **Object Detection and Identification for Blind People in Video Scene, H. Jabnoun, 2015** [29]. | **Real-Time color object recognition and navigation for QUARC QBOT2, W. Obaid et al. 2017**[31]. | **Adding Uncertainty to an Object Detection System for Mobile Robots, C. Hernandez et al. 2017** [33]. |
|---|---|---|---|
| **Dataset (Data collection)** | Private Dataset, 4 videos sequences (daily life objects). | Images captured by Microsoft Kinect. | Depth with RGB images of 3 objects, taken by RGB-D Camera |
| **Preprocessing or Enhancement** | This paper is different from the [28], in this they used color information | Split the scene into patches of 30x30 pixels | equalization, morphological operations, Gaussian filter and thresholding based on [32] |
| **Segmentation or Object Proposal** | SIFT localization is used. | Every patch is represented by Averaged histogram of RGB values of every pixel with its 8 neighbors<br>Apply Histogram intersection between every patch and determine the highest intersection | Contour extraction, Hough transform, and watershed.<br>A different technique for each object based on [32] |
| **Feature Extraction** | SIFT (Scale Invariant Feature Transform) from the color image. | Color histogram for detection<br>SIFT features | Geometric features.<br>Closet,solidity,extent, circularity, handle ratio.<br>Chair – circularity.<br>Screen – circularity, extent and aspect ratio. |
| **Classification** | SIFT features of the target image are matched with the database | - Performing SIFT between the model and the patch with the maximum intersection | Support Vector Machine (SVM)<br>uncertainty calculation is added on top of [32] |
| **Evaluation** | 95% true positive when the scale is 5 for the SIFT algorithm. | - 86% match in 1.2 seconds on windows 10 Intel core i5. | The detection rate is the same as [32], only uncertainty is added |
| **Limitations** | Detection failure caused by the quality of the image, the size of the target object (small), the high speed of the video scene. | Low recognition rate<br>Recognition and Localization of multiple objects in the same frame are considered. | Low recognition rate.<br>A small number of objects.<br>The techniques are not generalizable. |

As shown in the above tables, simple features like color histogram, SIFT or SURF are used to represent object features and in some of the works Depth images along RGB image are used. Some applied deep features for indoor object recognition. Object proposal techniques (i.e. Selective search), SURF or SIFT localization and Contour Detection are used for localizing objects in the image. Even though these previous related works scored promising success, they suffer from problem or limitations like low recognition rate, a small number of objects, techniques are not generalizable, not robust when rotation and occlusion, not recognizing multiple objects at the same frame, slow speed (not real-time) and other problems.

In order to overcome these problems, it is necessary to build a reliable and fast classification system to enhance the performance of indoor robot navigation.

## III. CHALLENGES IN OBJECT DETECTION

Object recognition and detection algorithms have many limitations because of changes in resource, illumination, scale, and other factors. Major challenges in object detection include the following.

- **Illumination**: The light changes throughout the day, which affects the image of an object. Weather conditions and shadows also can affect the image. The same object in different illumination may look different, making it difficult for the algorithm to discriminate.
- **Scale**: An object may appear in different sizes on different images. The selected feature should be robust enough to handle this change.

- **Rotation**: object recognition algorithm should handle rotation, as an object can appear rotated in the image.
- **Occlusion**: This is when an object is not completely visible or some part of an object is hidden. The algorithm should handle this condition.
- **Position**: the position of the object should not affect the success of the algorithm in recognizing the object.
- **Resource**: most recent algorithms approaches are based on deep learning which are usauly computationaly intensive.

## IV. APPLICATION OF OBJECT DETECTION FOR MOBILE ROBOT NAVIGATION

The application of the proposed work is to allow a mobile robot to differentiate between the objects in a scene to obtain properties. It would be used to assign a meaning to the environment and use this information for Semantic Navigation, Scene Recognition, and Environment categorization.

Generally the outcome of this research has a significant impact in agricultural monitoring, security, military and rescue operations. Specifically since the thesis only focus on the indoor environment it can be applied to service robots. In the world, 285 million peoples are estimated to be visually impaired and 82% of the peoples greater than 50 years and above were blind [37], [38]. We can overcome this problem by developing a visual substitution system that could help blind people to navigate around.

Some of the scenarios where this could be applied include.

- Assisting an elderly person or person with disability.
- Assisting Blind person to navigate in indoor environment.
- Message or object delivery service.
- Home security and surveillance
- Robotized wheelchair
- Floor cleaning

## V. CONCLUSION

Object detection techniques for mobile robot navigation are presented in this paper, including the general purpose object detection algorithms. For the detection of objects in an indoor simple local features like color histogram, FAST, ORB, SIFT or SURF are used to represent object features and in some of the works Depth images along RGB image are used which requires computational resources. Object proposal techniques (i.e. Selective search), SURF or SIFT localization and Contour Detection are used for localizing objects in the image. The general purpose object detection based on the deep features are computationally intensive however Mobile robots do not come with heavy computing power so the detection algorithm response time should be sufficient enough that the robots can make a decision fairly quickly. Eventhough there are several advanced progress in object detection for indoor mobile robot navigation it still required a lot of effort in order to be used in real life scenario where it assist humans in daily activity.

## REFERENCES

[1]. M. Takacs, T. Bencze, M. Z. Szabo-Resch, and Z. Vamossy, “Object recognition to support indoor robot navigation,” CINTI 2015 - 16th IEEE Int. Symp. Comput. Intell. Informatics, Proc., pp. 239–242, 2016.

[2]. IROBOT Corporation, “iRobot: Vacuum, Mop, &amp; Lawn Mower.” [Online]. Available: https://www.irobot.com/. [Accessed: 19-Sep-2019].

[3]. Q. Y. J. J. Z. Han, “ONLINE FEATURE EVALUATION FOR OBJECT TRACKING USING KALMAN FILTER,” IEEE Access, 2008.

[4]. C. G. S. A. S. Saravanakumar, A. Vadivel, “HUMAN OBJECT TRACKING IN VIDEO SEQUENCES,” vol. 2, no. 1, 2011.

[5]. D. G. Viswanathan, “Features from Accelerated Segment Test (FAST) Deepak Geetha Viswanathan 1.”

[6]. E. Senn, “BRIEF: Binary Robust Independent Elementary Features,” vol. 2011, no. 10/12, 2011.

[7]. E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: and efficient alternative to SIFT and SURF,” 2011 IEEE Int. Conf. Comput. Vis., 2011.

[8]. D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” Int. J. Comput. Vis., pp. 91–110, 2004.

[9]. L. Van Gool, “SURF : Speeded up robust features SURF : Speeded Up Robust Features,” no. July 2006, 2016.

[10]. H. Sun, L. Yan, P. Mooney, and R. Liang, “A new method for moving object detection using variable resolution bionic compound eyes,” Int. J. Phys. Sci., vol. 6, no. 24, pp. 5618–5622, 2011.

[11]. W.-L. L. ; J. J. Little, “Simultaneous Tracking and Action Recognition using the PCA-HOG Descriptor,” 3rd Can. Conf. Comput. Robot Vis., 2006.

[12]. Y. Zhong, A. K. Jain, and M. P. Dubuisson-Jolly, “Object tracking using deformable templates,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 5, pp. 544–549, 2000.

[13]. B. Alexe, T. Deselaers, and V. Ferrari, “Measuring the objectness of image windows,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 11, pp. 2189–2202, 2012.

[14]. J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders, “Selective search for object recognition,” Int. J. Comput. Vis., vol. 104, no. 2, pp. 154–171, 2013.

[15]. M. M. Cheng, Y. Liu, W. Y. Lin, Z. Zhang, P. L. Rosin, and P. H. S. Torr, “BING: Binarized normed gradients for objectness estimation at 300fps,” Comput. Vis. Media, vol. 5, no. 1, pp. 3–20, 2019.

[16]. P. F. Felzenszwalb and D. P. Huttenlocher, “Efficient Graph-Based Image Segmentation,” pp. 1–26, 2015.

[17]. I. Endres and D. Hoiem, “Category-independent object proposals with diverse ranking,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 2, pp. 222–234, 2014.

[18]. R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Region-Based Convolutional Networks for Accurate Object Detection and Segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 1, pp. 142–158, 2016.

[19]. R. Girshick, “Fast R-CNN,” Proc. IEEE Int. Conf. Comput. Vis., vol. 2015 Inter, pp. 1440–1448, 2015.

[20]. S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, 2017.

[21]. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 779–788, 2016.

[22]. W. Liu et al., “SSD: Single shot multibox detector,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9905 LNCS, pp. 21–37, 2016.

[23]. C. World, “YOLO9000:Better , stronger , faster,” no. April, 2007.

[24]. J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” 2018.

[25]. C. Astua, R. Barber, J. Crespo, and A. Jardon, “Object detection techniques applied on mobile robot semantic navigation,” Sensors (Switzerland), vol. 14, no. 4, pp. 6734–6757, 2014.

[26]. M. R. B. S. Riisgaard, “SLAM for Dummies : A Tutorial Approach to Simultaneous Localization and Mapping,” vol. 80, no. 4, pp. 699–705, 2000.

[27]. H. Lee et al., “Object recognition for vision-based navigation in indoor environments without using image database,” Proc. Int. Symp. Consum. Electron. ISCE, pp. 1–2, 2014.

[28]. H. Jabnoun, F. Benzarti, and H. Amiri, “Visual substitution system for blind people based on SIFT description,” 6th Int. Conf. Soft Comput. Pattern Recognition, SoCPaR 2014, pp. 300–305, 2015.

[29]. H. Jabnoun, F. Benzarti, and H. Amiri, “Object detection and identification for blind people in video scene,” Int. Conf. Intell. Syst. Des. Appl. ISDA, vol. 2016-June, pp. 363–367, 2016.

[30]. N. Diriba, “A HYBRID ALGORITHM FOR FAST DETECTION AND RECOGNITION OF SIGNAGE AND OBSTACLE AVOIDANCE FOR ROBOT NAVIGATION,” 2019.

[31]. T. R. and m. B. W. Obaid, “Real-Time Color Object Recognition and Navigation for QUARC QBOT2,” Int. Conf. Comput. Appl., 2017.

[32]. A. C. Hernández, C. Gómez, J. Crespo, and R. Barber, “Object Classification in Natural Environments for Mobile Robot Navigation,” Proc. - 2016 Int. Conf. Auton. Robot Syst. Compet. ICARSC 2016, pp. 217–222, 2016.

[33]. A. C. Hernandez, C. Gomez, J. Crespo, and R. Barber, “Adding uncertainty to an object detection system for mobile robots,” Proc. - 6th IEEE Int. Conf. Sp. Mission Challenges Inf. Technol. SMC-IT 2017, vol. 2017-Decem, pp. 7–12, 2017.

[34]. R. Pedro, “Object recognition for a service robot,” 2015.

[35]. X. Ding et al., “Indoor object recognition using pre-trained convolutional neural network,” ICAC 2017 - 2017 23rd IEEE Int. Conf. Autom. Comput. Addressing Glob. Challenges through Autom. Comput., 2017.

[36]. Yeong-Hwa Chang ; Ping-Lun Chung ; Hung-Wei Lin, “Deep learning for object identification in ROS-based mobile robots,” 2018.

[37]. W. H. Organization, “GLOBAL DATA ON VISUAL IMPAIRMENTS,” 2010.

[38]. D. H. A. and Z. Y. B. Fashe Markos Cherinet, Sophia Yoseph Tekalign, “Prevalence and associated factors of low vision and blindness among patients attending St. Paul’s Hospital Millennium Medical College, Addis Ababa, Ethiopia,” 2018.