Foreign object detection (FOD) using multi-class classifier with single camera vs. distance map with stereo configuration

by

Haoyuan Lin

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Computer Engineering

Program of Study Committee:

Arun K. Somani, Major Professor

Raj Aggarwal

Sigurdur Olafsson

Iowa State University

Ames, Iowa

2015

DEDICATION

I would like to dedicate this dissertation to my parents and friends without whose support and understanding I cannot complete this work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

First, I would like to express my deepest gratitude to my major professor Dr. Somani. Without him this thesis work is impossible to be done. I worked with him three years and benefited a lot from his insightful guidance and knowledge. I would like to specially thank him for his constant effort to give me research challenge opportunity and free thinking opportunity.

I would also like to thank Dr. Aggarwal and Dr. Olafsson for taking their precious time to serve on committees of my final oral thesis defense and providing me with constructive feedback on my research. Dr. Aggarwal is very kindly and significant to provide feedback from customers and make the project in the stable track. I learn a lot for him not only the way to solve the problem but also the way to solve the problem to satisfy customers need. Dr. Olafsson is the professor who leads me into the fascinating data science area. I have taken three key subjects that Dr. Olafsson are teaching in Iowa State University. Without their support and teaching, the algorithms and solution in this thesis can not be proposed.

Research is often inspired by the whole research group. I have the fortunate to study at Iowa State University and research at the Dependable Computing and Networking Laboratory. I would like to thank all my friends and research group members. I would like specially thank Cory, Parijat, Teng, Piyush, Koray, Pavan, David in this period.

ABSTRACT

Detection of objects of interest is a fundamental problem in computer vision. Foreign object detection (FOD) is to detect the objects that are not expected to be appear in certain area. For this task, we need to first detect the position of foreign objects, and then compute the distance to the foreign objects to judge whether the objects are within the dangerous zone or not. The three principle sources of difficulty in performing this task are: a) the huge number of foreign objects categories, b) the calculation of distance using camera(s), and c) the real-time system running performance. Most state-of-art detectors focus on one type or one class of objects. To the best of our knowledge, there is no single solution that focuses on a set of multiple foreign objects detection in an integrated manner. In some cases, multiple detectors can operate simultaneously to detect objects of interest in a given input. This is not efficient.

The goal of our research is to focus on detection of a set of objects identified as foreign object in an integrated and efficient manner. We design a multi-class detector. Our approach is to use a coarse-to-fine strategy in which we divide the complicated space into finer and finer sub-spaces. For this purpose, data-driven clustering algorithm is implemented to gather similar foreign objects samples, and then an extended vector boosting algorithm is developed to train our multi-class classifier. The purpose of the extended vector boosting algorithm is to separate all foreign objects from background. For the task of estimation of the distance to the foreign objects, we design a look-up table which is based on the area of the detected foreign objects.

Furthermore, we design a FOD framework. Our approach is to use stereo matching algorithm to get the disparity information based on intensity images from stereo cameras, and then using the camera model to retrieve the distance information. The distance calculated using disparity is more accurate than using the distance look-up table. We calculate the initial distance map when no objects are in the scene. Block of interest (BOI) is the area where distance is smaller than the corresponding area in the initial distance map. For the purpose

of detecting foreign objects, we use flood fill method along with noise suppression method to combine adjacent BOI with higher confidence level.

The foreign object detection prototype system has been implemented and evaluated on a number of test sets under real working scenarios. The experimental results show that our algorithm and framework are efficient and robust.

CHAPTER 1. INTRODUCTION

1.1 Objective and Difficulty

The robust detection of foreign object is a prerequisite for Advanced Driver Assistant System. In recent years, the number of approaches to detect object using kinds of sensors has grown rapidly. There are several research focus on using different sensors to detect foreign object, such as infra-red Bertozzi et al. (2006), lidar Mertz et al. (2013), radar Mohammadpoor et al. (2013) and ultrasonic Lee et al. (2014). Spinello and Arras (2012) focus on fusing different type of sensor information to make decision. Besides the high price of these sensors, a study Administration et al. (2012) National Highway Traffic Safety Administration NHTSA found that the performance of ultrasonic and radar is poor and limited in range and pointed that camera based solutions are the most promising technology which can better know the shape of object.

Our objective is to detect the objects that are not expected to be appear in certain area. These objects are defined as foreign objects in this paper. For this task, we need to first detect the position of foreign objects if the foreign objects exist in the scene, and then compute the distance to the foreign objects to judge whether the objects are within the dangerous zone or not. Different application may tolerant different latency in detection. Latency is counted as the number of frames missed before a detection occurs.

There are three principle source of difficulty in performing this task. Firstly, the number of foreign objects categories could be huge since any objects except normal objects is defined as foreign objects. The solution has to handle all categories of foreign objects. Secondly, the foreign object distance is significant to be measured. For safety issue, any close foreign objects are dangerous. The distance can be easy to compute using lidars or radars by measuring the

time it takes to return to its source. But the distance computation is relative hard using cameras. Thirdly, the alert has to be alarmed immediately as soon as the foreign objects are detected. This requires the system must have real-time processing performance.

Most object detection research focuses on how to design both accurate and fast detector for one particular class of objects Dalal and Triggs (2005), Tang et al. (2012), Ding and Xiao (2012). In order to detect foreign objects, multiple detectors are deployed simultaneously, each detecting the respective objects for a given input. This is not efficient since each frame has to be processed by all detectors, and a limit number of detectors are hardly to cover all categories of foreign objects. On the other hand, many applications do not care about the categories of the objects.

1.2 Motivation of Our Approach

Detecting objects but not necessarily classifying them may be a requirement for many safety applications. The classification step can be considered as redundant for the system. In other applications, our first interest, the most useful information is whether any of the objects is present or not. If it does, the location and size may be of secondary interest. The application of this technical is not only for driving assistant system but also for robot that can search and find target automatically. A large pool of objects that should be located in the direction of moving can be recognized by this system.

We consider objects of interest as foreign objects which can be classified in one ensemble positive class. The merit of this method is by considering the similarity among all foreign objects, a group of similar features can be used and shared in ensemble positive class. Foreign objects can be of any arbitrary shape with certain volume in a specified range. Based on the experiments, we observe that the deformation of objects may degrade the performance of such detectors.

In our applications, the number of normal objects categories is much smaller than the number of foreign objects categories. In our work, we firstly detect normal objects instead of detecting foreign objects. In this way, the number of objects categories is reduced sharply, so the detector is easy to design. In order to detect foreign objects, we detect all objects and then

ignore those normal objects.

1.3 Overview of Our Approach

For object detection with multiple categories of objects, we use a divide-and-conquer strategy. Our detector is based on a tree-structure which can be easily adopted for a coarse-to-fine strategy to handle multiple category objects. Each node uses a cluster algorithm to split the data into several clusters before training the classifier. Since different categories of objects can also be similar in certain parts, several different categories of objects can be handled at one node in the tree. Each node in the tree is a strong classifier that puts emphasis on the diversities among these clusters. The tree-structure detector is based on the shape discrimination.

For the purpose of objects detection, we consider using two cameras to generate disparity map and then calculate the distance map using camera configuration parameter. We use flood fill method to divide the frame into separate areas. The flood fill method groups adjacent areas with the same distance into a separate area. After detecting all objects, three discrimination is followed to ignore the objects that are not qualified as foreign objects. Distance discrimination is followed to filter out the object outside the dangerous zone. Size discrimination is applied to filter out small area blocks which is not dangerous. At last step, non-foreign object detector is applied to ignore the normal objects. The merit of this method is to consider these foreign objects as a block with certain area and not to consider which categories they belong to. The workflow of this method is shown in Fig. 1.1

1.4 Outline of Thesis

The rest of the paper is organized as follows. Chapter 3 describes the multi-class classifier method using machine learning algorithm to detect multiply categories of objects and demonstrate the correctness of the algorithm. Chapter 4 describes the FOD framework using depth map. Then we compare the performance of FOD frameowork to the multi-class classifier method. Lastly, we conclude in Chapter 5.

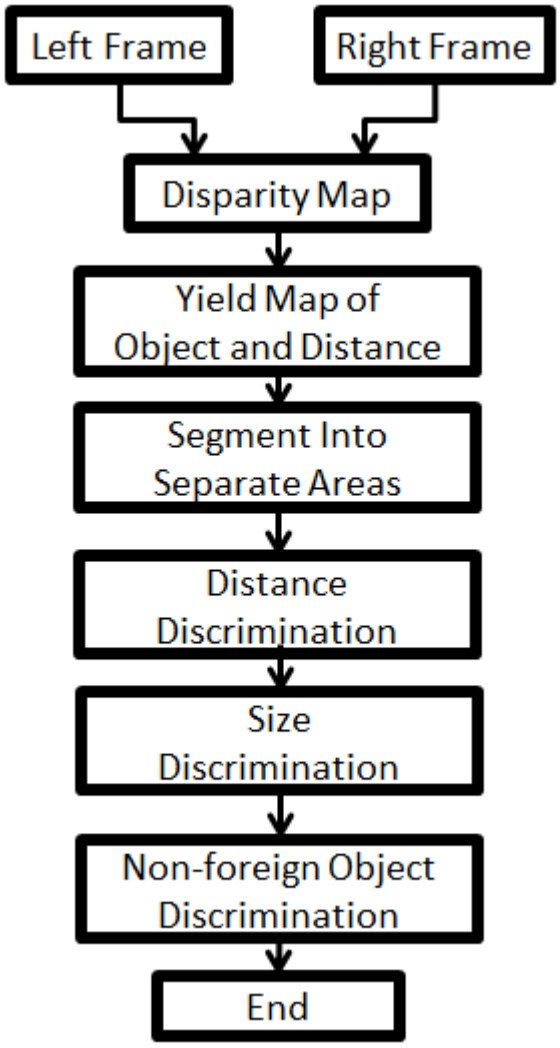

Figure 1.1 The FOD system framework.

CHAPTER 2. RELATED WORK

Object detection is a fundamental problem in computer vision. For example, pedestrian protection system Tawari et al. (2014)(implemented in the advanced driver assistant system) and intelligent robot Ćirić et al. (2014) that can follow the specified target are two good examples of such applications. Object detection has seen a great progress in both performance and detection speed in the last few years Dollar et al. (2012).

2.1 Image Features

A large variety of image features has been developed for object detection. The edge template by Gavrila (2007) is using edge detection method to get the object shape. When detecting the edge feature, the edge detection method is applied to the whole image. It is a global feature. However, most recent researches find the global features are sensitive to occlusion and light condition.

Local features are developed and they are less sensitive to occlusion. Haar-like feature Papageorgiou and Poggio (2000) considers the pixel intensity difference between adjacent rectangular regions at a specific location shown in Fig. 2.1 as a vector. Each feature is only extracted from the close pixels instead of the whole picture. Haar-like feature has been applied successfully to be efficient in human face detection problem. However, Haar-like feature cannot perform well under the condition of clustered background.

Histogram of oriented gradients (HOG) Dalal and Triggs (2005) is using edge orientation histogram to describe the shape of the objects. The technique counts occurences of gradient orientation in localized portions of an image. The basic idea behind the HOG is that local object shape can be described by the distribution of intensity gradients. Algorithm using HOG

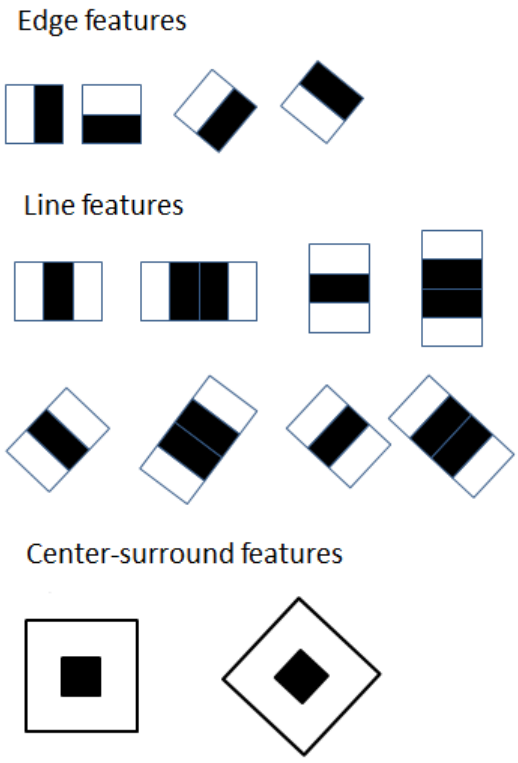

Figure 2.1 Haar-like Feature. Black Areas Have Negative and White Areas Positive Weights.

feature was developed Dalal and Triggs (2005) to boost the performance for object detection since HOG feature can tolerate the mis-position or deformation of samples.

2.2 Classification Method

After the features are computed, they are fed into a classifier. The classifier could be SVM Joachims (2002) or Real AdaBoost Schapire and Singer (1999).

Support vector machine (SVM) Joachims (2002), Wang et al. (2014) is using optimization method to find a separating hyper-plane to solve the pattern recognition problems. Boosting algorithm Zhu et al. (2009) is another powerful learning algorithm which composes several

weak classifiers into a strong classifier. Compared to SVM , boosting algorithm chooses a small amount of representative features to construct the detector rather than all features as is the case for SVM.

Real AdaBoost Schapire and Singer (1999) also has been proposed to handle multi-class multi-label problems. Vector boosting Huang et al. (2005) is proposed to extend the output of the Real AdaBoost from scalar to vector which allows one sample to be assigned into several classes.

CHAPTER 3. MULTI-CLASS CLASSIFIER WITH SINGLE CAMERA

To achieve the goal of detecting foreign objects, an idea is to use several categories of objects such as pedestrian, vehicle or animal to cover the most common foreign objects. The foreign objects detection task is converted to a multi-classification task. Our algorithm consists of the following steps. First, the clustering algorithm is applied after the feature extraction to separate the foreign objects samples into several clusters. Then the quality of clustering is defined. The clustering algorithm will be terminated if the clustering evaluation achieve the pre-defined quality of clustering. Next, The label of the samples is associated by the name of cluster. At last, the boosting classification algorithm is applied to train the multi-class classifier. The overview of our method is shown in Fig. 3.1.

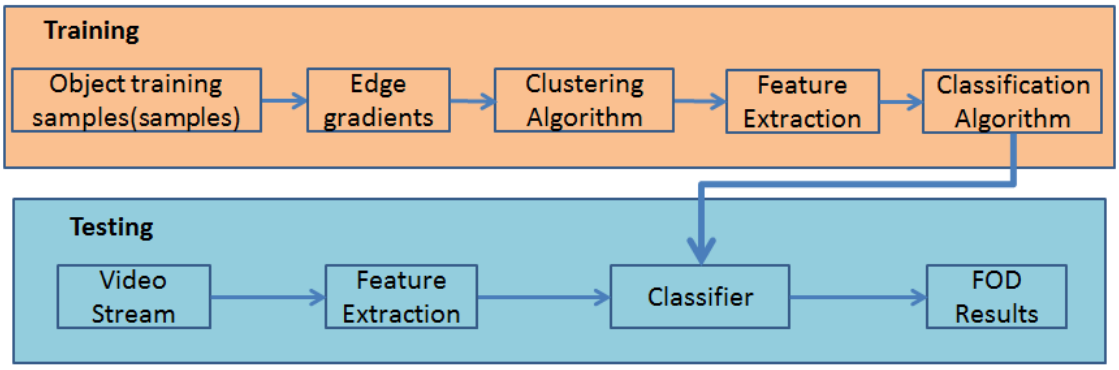

Figure 3.1 The overview of our single camera multi-class classifier method

3.1 Clustering Algorithm

In this paper, we use vector-space model to represent clustering algorithm which is an iterative process. The goal of clustering algorithm is to maximize the internal similarity within each single cluster and minimize the external similarity between any two clusters.

Assume there are k clusters. We denote i_{th} cluster as C_i $(i = 1, 2, ...k)$. Let n_i be the number of training samples in C_i. Each training sample x_i is a matrix of pixel values. The feature of each training sample x_i is considered as one vector as $\varphi(x_i)$. The function of $\varphi()$ is to map from a set of pixel values to a vector of feature values. $\varphi(x_i)$ is computed by applying feature extraction algorithm to each training sample. An example is shown in Fig. 3.2.

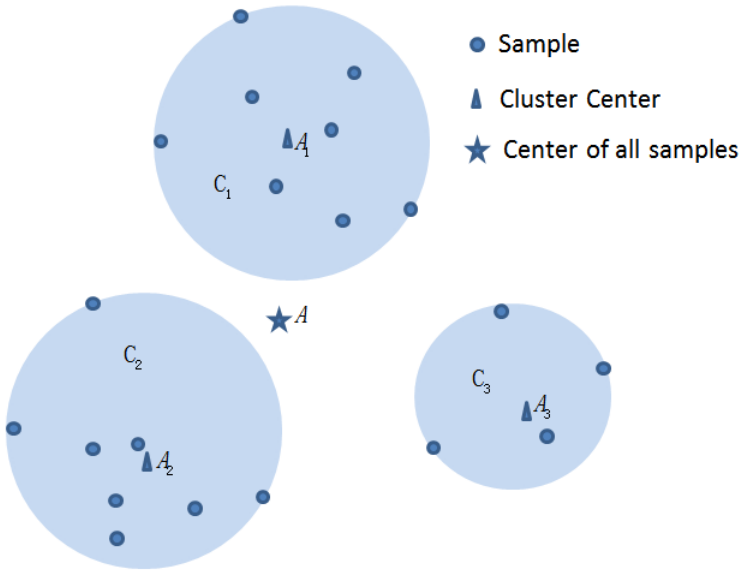

Figure 3.2 Schematic Diagram Depicting Multiple Clusters and Their Centers

The similarity between two samples x and y using Euclidean distance is defined as

$$\| \varphi(x) - \varphi(y) \|^2$$

Based on this definition, the similarity of the C_i cluster s_{1i} is defined as the average sum of

relative distances among all sample points in C_i to the center of C_i which is denoted as A_i.

$$s_{1i} = \sum_{x \in C_i} \parallel \varphi(x) - \varphi(A_i) \parallel^2$$

The similarity between any two clusters s_{2i} is defined as the sum of relative distances from the center of each cluster A_i to the center of all clusters which is denoted as A. Since the size of each cluster is not the same, the contribution of each cluster is weighted by its size as follows.

$$s_{2i} = n_i \parallel \varphi(A_i) - \varphi(A) \parallel^2$$

.

We use both internal evaluation and external evaluation to evaluate the quality of clustering, the score s is defined by

$$s = \frac{s_1}{s_2} = \frac{\sum_{i=1}^{k} s_{1i}}{\sum_{j=1}^{k} s_{2j}}$$

where s_1 denotes as internal evaluation, and s_2 denotes as external evaluation. The goal of clustering algorithm is to maximize the score s.

Not all features extracted from samples are suitable for clustering. The classification power of the feature can be measured by Z value which is defined as follows.

$$Z = 2 \sum_j \sqrt{W_+^j W_-^j}$$

We define the foreign objects as positive samples and image patches excluding foreign objects as negative samples. W_+ is the sum of weights for positive samples and W_- is the sum of weights for negative samples. The weight for all samples is normalized. The feature is more discriminating if the difference between W_+ and W_- is bigger. Therefore, the smaller value of Z implies that the classification power of the feature is more discriminating. Not all features will be useful for discriminating the objects. Boosting algorithms Schapire and Singer (1999) consist of iteratively learning weak classifiers with respect to a distribution and adding them to a final strong classifier. With the help of Z value, a threshold is set to determine if the features selected by boosting training is close to saturation and no more features could be selected. Then boosting algorithm should stop. If the selected features are not close to saturation, clustering

algorithm should be called to split the training samples into more clusters. The results of the boosting algorithm are a set of features which are selected to construct a strong classifier in the node. Meanwhile, these features are also used by clustering algorithm to prepare the samples for the next level node.

Since the background of positive training samples can be regarded as noise, some samples may not be assigned to a representative cluster due to the noise. This will produce clustering bias in the result of the clustering algorithm and will result in worse effects in the subsequent training stages. In the next step, the boosting algorithm will make biased decision based on the worse effects which are then used to select the most representative feature to train a classifier. To solve this problem, we set a safe barrier that can tolerate the bias effect from the cluster. In a addition, some examples could be assigned to multiple clusters when they are on the cluster edge or close to other clusters. This is a modification of the original clustering algorithm.

Initially clusters are assigned randomly into two sets. After calculating the center of each cluster, we use the defined distance to reassign each sample to the nearest center. Then the center will be calculated based on the new distribution of the samples. In our work, the center is the average vector value of all samples in the cluster. These cluster formation step is iterated upon with number of initial being 3, 4, 5... until the algorithm achieves the pre-defined quality of the clustering. Successively we arrive at better and better set of clusters that helps to arrive at the final configuration using the criteria defined in the previous paragraph.

Reassignment is that each sample is transferred into all other clusters and calculate the score using the pre-defined criterion. The sample will be reassigned into the cluster which has the lowest value of the score. The above iterative steps will be terminated until there are no samples are not reassigned. The detail of our clustering algorithm is shown as follows.

Algorithm 1 Clustering Algorithm

Input:

The sample set S=$\{\vec{x}_1, ..., \vec{x}_N\}$(set of samples to be clustered)

The maximum cluster number K (Initial value of K can be used only if you already know the maximum number of clusters, otherwise, K should be set to a large number)

The overlap parameter b (In our experiment, b is set to 1.2)

Output:

The cluster set W=$\{\vec{w}_1, ..., \vec{w}_k\}$(samples in each cluster)

The cluster centroids set C=$\{\vec{c}_1, ..., \vec{c}_k\}$ (set of cluster centroids)

for $k \leftarrow$2 to K **do**
  $\{\vec{\mu}_1, ..., \vec{\mu}_k\} \leftarrow RandomSeeds(\{\vec{x}_1, ..., \vec{x}_N\}, k)$
  for $i \leftarrow 1$ to N **do**
    $j \leftarrow \underset{jj}{\text{argmin}} \|\vec{x}_i - \vec{\mu}_{jj}\|$
    $\vec{w}_j \leftarrow \vec{w}_j \cup \vec{x}_i$
  end for
  $\{\vec{w}_1, ..., \vec{w}_k\} \leftarrow Reassignment(\{\vec{w}_1, ..., \vec{w}_k\})$
  for $i \leftarrow 1$ to k **do**
    $d_i \leftarrow b$ * $CalculateClusterDiameter\ (\vec{w}_i)$
    $\vec{c}_i \leftarrow CalculateClusterCenter\ (\vec{w}_i)$
  end for
  for $i \leftarrow$1 to k **do**
    $\vec{w}_i \leftarrow \{\}$
  end for
  for $i \leftarrow 1$ to N **do**
    for $j \leftarrow 1$ to k **do**
      if $\|\vec{x}_i - \vec{c}_j\| < d_j$ **then**
        $\vec{w}_j \leftarrow \vec{w}_j \cup \vec{x}_i$
      end if
    end for
  end for
end for

Algorithm 2 Reassignment

Input:

The sample set $S=\{\vec{x}_1, ..., \vec{x}_N\}$

The initial cluster set $W_1=\{\vec{w}_1, ..., \vec{w}_k\}$

Output:

The final cluster set $W_2=\{\vec{w}_1, ..., \vec{w}_k\}$

repeat
  $flag \leftarrow false$
  $score \leftarrow CalculateScore\ (\{\vec{w}_1, ..., \vec{w}_k\})$
  for i←1 to N **do**
    $p \leftarrow GetClusterId(\vec{x}_i)$
    $\vec{w}_p \leftarrow \vec{w}_p - \vec{x}_i$
    $[s, q] \leftarrow FindMaxScore\,(\{\vec{w}_1, ..., \vec{w}_k\}, \vec{x}_i)$
    if $s > score$ **then**
      $flag \leftarrow true$
      $\vec{w}_q \leftarrow \vec{w}_q \cup \vec{x}_i$
    else
      $\vec{w}_p \leftarrow \vec{w}_p \cup \vec{x}_i$
    end if
  end for
until $flag$

3.2 Classification Algorithm

The criteria for choosing the proper classification algorithm is the effectiveness as well as computational load. In Huang et al. (2005) vector boosting algorithm is proved to be a good one compared to other classification algorithms. The property of relative fast evaluation speed and the adaptive ability to the multi-class multi-label problem is suitable for our problem. In this section, we compare Real AdaBoost to vector boosting with an example, and then describe the detail of our extend vector boosting algorithm.

Real AdaBoost Schapire and Singer (1999) has been proposed to handle multi-class multi-label problems. Multi-class problem is to classify instance into more than two classes instead of only positive class and negative class. Multi-label problem is that each instance could be assigned to more than one class. The idea of Real AdaBoost method is to make each class orthogonal and consider each label independently so that this problem can be converted into original binary classification problem. One prediction is considered as correct if and only if all class labels of the instance are predicted correctly. However, this condition is not tenable since some attributes are dependent.

Vector boosting Huang et al. (2005) is proposed to extend the output of the Real AdaBoost from scalar to vector. The difference is that Real AdaBoost converts multi-label problem to several binary classification problem (the attributes are required to be independent) while vector boosting algorithm consider all labels for one instance together (the attributes are not required to be independent). Vector boosting algorithm modifies the sample weight redistribution formula using vector dot production which can avoid some independent status. For example, there are three classes horse, deer, human. The ground-true label of horse is $\{1, 0, 0\}$, the ground-true label of deer is $\{0, 1, 0\}$, and the ground-true label of human is $\{0, 0, 1\}$. One sample is clustered by the clustering algorithm as $\{1, 1, 0\}$ which means it is like horse and deer and may be any of them (the deer and horse are four-leg animals that look like similar from a certain distance). In Real AdaBoost, classifier output of $\{1, 1, 0\}$ is considered as the only right estimation. However, in vector boosting, besides $\{1, 1, 0\}$, classifier outputs $\{1, 1, 1\}$ is also considered as right estimation since the third attribute is independent of the other two. Real AdaBoost is suitable for problems in which the label of sample is absolutely with no ambiguity while vector boosting algorithm allows ambiguity to be handled. Evaluating all attributes together is the merit of vector boosting algorithm.

In our problem, we use a label denoted as either 1 or 0 to indicate if the sample belongs to the cluster or not. A label of value 0 means the sample does not belong to the cluster. A label of value 1 means the sample may belong to the cluster. Label 0 means the instance is not in this class which shares the same concept in Real AdaBoost. When the number of label 1 in one instance equals to one, the meaning is the same as in Real AdaBoost. When the number

of label 1 in one instance is more than one, different from Real AdaBoost, the instance could be in any class of these labels but cannot be in those classes where the label is 0. The label meaning is determined by our clustering algorithm. So Real AdaBoost and vector boosting algorithm cannot work for our problem.

The following example illustrates the meaning of the label in our problem. One sample is in the area safe barrier between class 1 and class 2. The sample is clustered as class 1 and class 2. The real label should be $\{1, 1, 0\}$. The correct prediction of the Real Adaboost is $\{1, 1, 0\}$ since all class label of the instance should predicted correctly. The correct predictions of the vector boosting are $\{1, 1, 0\}$, $\{1, 1, 1\}$ since the labels of the first two classes are correct and the label of last class is independent of the other two. The correct predictions of the extended vector boosting are $\{1, 1, 0\}$, $\{1, 0, 0\}$, $\{0, 1, 0\}$ since any label of the first two classes is correct and the label of the last class is 0. The difference is shown is Table 3.1.

The procedure of the classification algorithm is described as followed. First each sample will be assigned each weight. The classification algorithm will make the wrong prediction sample with higher weight and make the correct prediction sample with smaller weight. The definition of the correct prediction sample is any one label is predicted correct. Here is an example. If the real label of the sample is 1, 1, 0. The correct predictions are 1, 1, 0, 1, 0, 0, 0, 1, 0, and other predictions are all wrong. The final strong classifier is the sum of classifier trained in the iteratively process. The algorithm pseudo code is shown as follows.

Table 3.1 Correct Predictions With Different Boosting Algorithm.

Real label	$\{1, 1, 0\}$
Real Adaboost	$\{1, 1, 0\}$
Vector Boosting	$\{1, 1, 0\}\{1, 1, 1\}$
Extended Vector Boosting	$\{1, 1, 0\}\{1, 0, 0\}\{0, 1, 0\}$

Algorithm 3 Classification Algorithm

Input:

The sample set S=$\{(\vec{x}_1, \vec{v}_1), ..., (\vec{x}_N, \vec{v}_N)\}$ where $\vec{v}_i$ is got by our proposed cluster algorithm

1: Initialize the sample weight as $D_1(i) = 1/N$

2: **for** $t = 1, ..., T$ **do**

3: Under the updated sample weight, train a weak classifier $h_t(x)$

4: Update the sample weight

$D_{t+1}(i) = \frac{D_t(i) exp[v_i \otimes h_t(x_i)]}{Z_t}$

where Z_t is a normalization factor to make $D_{t+1}(i)$ a distribution

5: **end for**

6: The final strong classifier $H(x) = \sum_{t=1}^{T} h_t(x)$

3.3 Experiments and Analysis

We apply our method to the problem of FOD in a real scenario. To demonstrate the feasibility, we trained multiple categories of object detector. In this experiment, we consider objects that often appear on the road that can cause a traffic accident. We apply our algorithm on four big object categories (pedestrian, bicycle, vehicle, and four-legged animal). Reference detector is also trained for each category to compare the performance with multiple categories object detector. A big variance of shape that cannot be handled by one classifier could be seen in our setup. Our goal is to make all objects in these categories distinct without knowing the sub-category in advanced.

At the first stage, all positive samples are fed into the general Real boosting algorithm to select the feature that can be used to distinguish the positive samples against the negative samples. The value of the threshold Z is obtained by cross-validation to make sure that the training is not overfitting (Z value is too big) or underfitting (Z value is too small). When the Z value achieves the threshold, this set of features is used for clustering algorithm. A small portion of features may come from the background when Z value is close to threshold, therefore we set a tolerant ratio. This tolerant ratio is defined as the executing diameter divided by

diameter of each cluster. In this experiment, the ratio is set to 1.2. Any instance locating in the overlapping area may belong to multiple clusters. After the clustering algorithm is computed, each cluster is trained by using modified vector boosting algorithm instead of general boosting algorithm to select discriminative features to construct strong classifier. These steps are performed iteratively until the training error is under the predefined level. At the end of training, we get 58 leaf nodes in the FOD tree. In Fig. 3.3 we show parts of results in the node which contains the samples in each cluster.

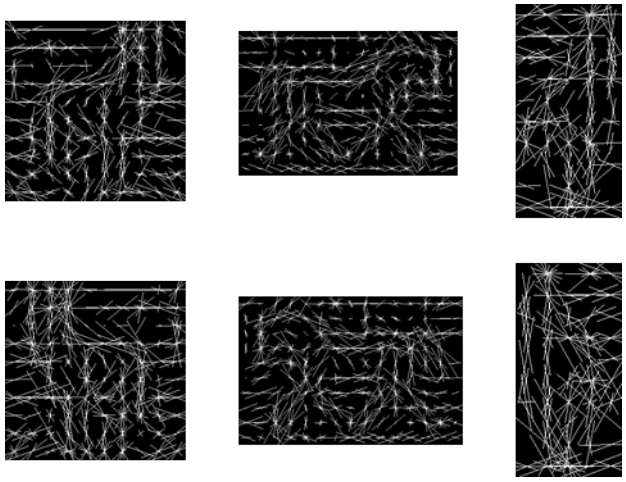

Figure 3.3 The clustered Horse Category Samples in the Node of the Foreign Detection Tree

Since the width-to-height ratio of objects is different, we propose a simple method to shrink the area that can pass all stages of the FOD classifier. The idea is based on that the discriminating features are mostly located around object outline and seldom located in background area. The operation is to squeeze the area from four directions in three steps. First, get four strips from boundary of the output box and get the number of discriminating features in the area. Second, squeeze the output box in the direction which has the least number of discriminating features. Third, calculate the total number of features that has been removed. If it is more than 10% of the total features, then stop. If not, go to first step.

In order to compare the performance of proposed method to the method composed of multiple classifiers, we build four classifiers for four objects, (i.e. pedestrian, bicycle, vehicle and four-leg animal) for each categories. We made each of the multiple classifiers learn using the samples coming from the single category belonging to the same training data set and evaluate on the same test data set. For each classifier, the training error is set to 5*10e-4 and training detection rate is set to 0.95 which is the same as the FOD. Next, we evaluate the performance of 4 separated classifiers. We calculate the number of missing and false positive is from four categories. Fig. 3.4 plots the ROC curves.

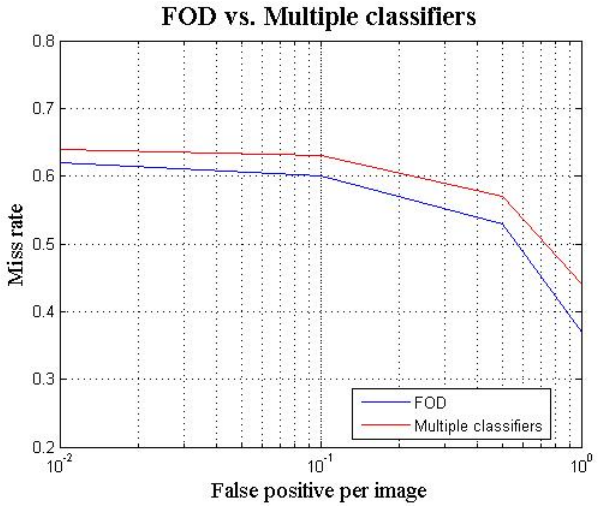

Figure 3.4 Quantitative Result of FOD vs. Multiple Classifiers

We observe that our FOD method outperforms the multiple classifier method. One main reason is that we use soft clustering algorithm which is based on the discriminating features while multiple classifier method use pre-defined category information which is hard clustering algorithm based on the domain knowledge. In this experiment, we notice that the merit of soft clustering algorithm, which is allowed to be tolerant between similar categories. Another reason is that the extended vector boosting method uses relation information of each feature among multiple classes which is totally ignored by the separated classifier method.

CHAPTER 4. DISTANCE MAP WITH STEREO CONFIGURATION

4.1 FOD System Framework

For foreign object detection with any shape, it is necessary to find a certain stable pattern for the foreign object. The basic idea behind our method is that the distance between cameras and any part of one object are much more similar than other objects in the scene. Disparity map Sanger (1988), Kauff et al. (2007) is widely used in the computer vision area to recover the 3D structure of a scene using two or more images of the 3D scene, each acquired from a different viewpoint in space. With a set of well configured cameras, disparity, which means the distance between the two corresponding points, can be calculated. The disparity is calculated by finding the corresponding points in the two frames which have a similar feature. One method to calculate the disparity is using the feature matching Izadi et al. (2011).

We use stereo vision to address the problem in hand. The two camera model is shown in Fig. 4.1. The parameters used in this paper are shown in Table 4.1.

In Fig. 4.1, based on the triangulation principle, we have $\frac{x_l}{X_l} = \frac{f}{Z}$ and $\frac{x_r}{X_r} = \frac{f}{Z}$. Manipulation of these equations give us $X_l = \frac{Z}{f}x_1$ and $X_r = \frac{Z}{f}x_r$. Also $X_l + X_r = T$. Therefore, $\frac{Z}{f}(x_l + x_r) = T$ where $x_l + x_r$ is the disparity d. Then the distance between target and camera is given by $Z = \frac{Tf}{d}$.

Table 4.1 Parameters of The Camera Model.

Distance between the two cameras	T
Focus length of the cameras	f
Distance between target and camera	Z
Position of middle point of camera film	c
Position of object in camera file	x
Disparity of the target	d
Displacement between target and camera	X

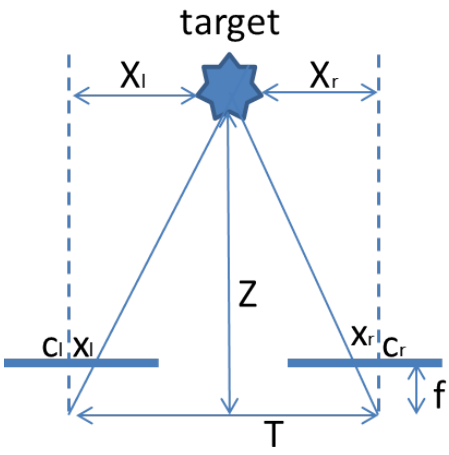

Figure 4.1 The Structure of FOD

The depth map is an important resource in the algorithm. In order to make the distance calculation robust, a block of pixels are grouped together. The distance calculation is based on the blocks instead of pixel that can eliminate single error. The calculation of the disparity also uses the feature matching algorithm Izadi et al. (2011). Edge feature within a block in the left frame can be used as the pattern to search in the right frame. For the purpose of the FOD task instead of 3D construction, the depth calculation is calculated coarsely. If the area of the block is too small, holes will be seen in the depth map. If the area of the block is too big, the object may not be detected especially when the object is far away from the camera.

In the Fig. 4.2, the number denotes the distance between background and camera in meters. Once the depth map for the initial scene is generated, the system is ready to work. The distance of the new frame is calculated and compared with the initial depth map. If the value of depth is smaller than the initial value for the block, foreign object may exist in this block and the position of the block is memorized. The distance filter can be used to filter the background object. Any feature block may be ignored beyond the range of interest. Only feature block within the range is marked. After the processing of the filter, blood fill algorithm is used to

connect the neighbor block into one integrated block. Each integrated block is represented as one object. The distance is also related to the integrated block. In our experimental setup, we use 8-neighbor rule to recognize the neighbor candidate around one block. We consider that the foreign object can be any kind of shape, therefore, we believe that all eight directions should be considered as the extension of the foreign objects.

Size discrimination is implemented using a make-up table with the distance information and object size information. When a small object is too close to the camera, the size of the pixel block may appear as a big block. By using the make-up table we store the lower-bound level of the size which has high confidence. Fig. 3 shows the result obtained from the filter using the depth map when a foreign object (a vehicle) stopped in front of the camera. The number is depth of the block in meter. The rectangle denotes the position of the foreign object.

4.2 Adaptive Depth Calculation Algorithm

The value of depth may not be a constant for one particular block if the camera is moving on the rugged ground. The initial depth map generated for the initial scene cannot be the reference for the following frames. For example, the vehicle is stand-by on an even ground. The depth map is generated based on the initial scene at this time. Once the vehicle is running on the down ramp, the distance between the camera and ground will be smaller than the initial scene since the camera is pointing down than the initial state. In this case, there will be false alarms even if there is no foreign object in the scene. The reason is that there will be areas of quite small value of the depth when the camera is pointing down to the ground. The system with high rate of false alarms is not useful at all.

An easy way is to solve this problem is to generate a depth map with the minimum depth value for each block. Only when the value of the depth is smaller than the minimum value on the depth map, the foreign object can be determined with a higher level of confidence. In order to get enough data to train the initial depth map, the vehicle with mounted cameras is set to run on the experiment site to collect the data. Then the depth map for each frame is calculated. The minimum value of depth for each block will be the value in the depth map. The advantage of this method is to reduce the false alarm sharply. However, this method will

lose a certain number of blocks which the foreign objects are located in. The reason is that the comparison is based on the minimum depth map which is a very adverse situation.

A better solution is to use information on the current frame appropriately. The method will not require the initial scene to generate the initial depth map. The area of the object shows in the frame can be part of scene if the object is in a certain distance from the vehicle or full of scene if the object is very close to the vehicle. Both cases can be handled using only the current frame without any help of the initial frame. If the area of the object takes the part of scene, the rest of the scene could be the clue to get the distance from the camera to the ground. If the area of the object takes full of scene, it is easy to handle using the depth map with the minimum depth as described above. The detail of the method is described as follows.

In order to make the distance calculation robust, the area of each frame is divided into sets of blocks instead of each pixel. The dimension of the block map is m*n. The denotations used in this algorithm are shown in Table 4.2.

Table 4.2 Correct Predictions With Different Boosting Algorithm.

The number of blocks in a row	m
The number of blocks in a column	n
The denotation for each block	b_{ij}
The depth value of block b_{ij}	d_{ij}
The maximum depth value in row i	m_i
Threshold of depth different value in row i	t_i

For each block b_{ij}, the distance d_{ij} will be calculated using the disparity. For each row, the maximum depth value is calculated. In order to eliminate the outlier value, the median value of the first three big numbers is picked as the maximum depth value m_i. If the depth value of any block in row i is smaller than $m_i - t_i$, the block will be marked as a dangerous points. Like the last section, a breadth first search algorithm will be used to group. The optimization objective is to find such a set of threshold t_i for each row i such that the error rate of detecting is high and the false alarm is low.

4.3 Experiments and Analysis

In this experiment, we use nine data sets to test the correctness of the algorithm shown in Table 4.3. The data we used in the experiment were collected by us using a real set up for applicatiion (anonymous). The first four data sets only include foreign objects without normal objects. The categories of the foreign objects are human and vehicle. The last five data sets includes both foreign objects and normal objects. The category of the normal objects is bale. For each data set, the # frames means the number of frames in the video clip and the # object means the times the object shows up in the video clip. The following is the definition of the columns of the table. D is the number of times the foreign objects are correctly detected, M is the number of times an object was not detected in the frames considered and FA is the number of times that there is no object but the algorithm outputs one result at that position. We make the experimental cases changeable and control the moving direction of the object for the purpose of all corner cases. The foreign object is designed to move on the designated path. The object could move from the far site to the near site. The object could move from the near site to the far site and move from left to right with the same distance. The pose of the objects can be arbitrary shape.

The multi-class classifier approach is based on the training method. The classifier will detect any object with the similar shape that are fed into the training data set. The stereo-based algorithm is based on the physical scenario relationship. The area and physical size will be the feature to discriminate the object from the background.

We compare our stereo-based algorithm to the previous work using the multi-classifier method. From the results, we observe that the detecting rate increased more than shape-based method. This can be contributed to the reason that the distance can be tolerated when the object is with shape that is not included in the model of the shape-based method. Besides that, the false alarm is also decreased than the shape-based method. The background may be clustered in most case, so the foreground and background may mix together to make the frame area much more like a target object. However, the distance is always not in the range. The area can be eliminated by the depth information.

Table 4.3 Experiment Result Comparison Between Stereo-based and Shape-based Method.

Datasets	Stereo-based			Shape-based		
	D	M	FA	D	M	FA
Dataset1 (42 objects/116 frames)	40	2	0	39	3	2
Dataset2 (15 objects/34 frames)	14	1	0	13	2	0
Dataset3 (12 objects/56 frames)	11	1	0	10	2	0
Dataset4 (22 objects/60 frames)	21	1	1	21	1	2
Dataset5 (12 objects/46 frames)	11	1	0	8	4	2
Dataset6 (17 objects/51 frames)	16	1	0	10	7	1
Dataset7 (7 objects/35 frames)	7	0	0	4	3	0
Dataset8 (18 objects/36 frames)	11	7	0	7	11	1
Dataset9 (14 objects/51 frames)	7	7	0	7	7	0

From the detecting results, there is no false alarms in all datasets except one false alarm in dataset 4 using stereo-based algorithm. From the machine operating point of view, the more false alarms the algorithm generates, the more improper stop operation will be made. The miss is one or two frames in dataset 1 to dataset 7 and seven in dataset 8 and dataset 9 each. In data set 8 and 9, the foreign object happens to be behind an object of interest and moves away from it. That is when we have more difficulty in separating the object of interest and foreign object. We collected data at the rate of 2 to 3 frames per second That means the detection is delayed by 3-4 seconds. For some application this may be acceptable. The reason is that the foreign objects and normal objects are close. We use shape detection for the normal objects and the bound is a little wider than the exact objects. The shape method is using the different between the foreground and background, so the background has to be included to get the contrast. When the foreign objects are not close to the normal objects, the detection rate is very high.

4.4 Future Work

More research could be done in following directions.

1. The disparity does not work very robust beyond 10 meters. More research could be worked on how to improve the performance of the disparity beyond 10 meters. The possible direction is to make new set-up of two cameras or new technology.

2. The tracking module could be the second research point. In this work, we use linear formula to do the prediction. It is not perfect when the number of the detection samples is not enough. Other prediction algorithms can be not linear to make the prediction more accurate.

3. The calculation of the distance can be more optimal. In this work, we use 50*50 block to calculate the robust distance to reduce the noise from disparity. The bigger block is used, the effect of the noise could be reduced. However, the distance calculation is not accurate when using bigger block. The size of the block is chosen considering the trade-off.

Figure 4.2 The Result After The Blood Fill Algorithm When One Foreign Object (vehicle) Is in Front of The Camera

CHAPTER 5. SUMMARY AND CONCLUSION

We describe a method to learn a foreign object detector which can detect many categories objects simultaneously, without knowing the label information in each instance. We divide the positive sample space by unsupervised clustering algorithm with tolerance ratio with the features learned by our proposed extended vector boosting algorithm. The extended vector boosting algorithm considers the relation of multiple classes instead of individual classes. In the research on detection of any foreign objects with no pre-set shape, we use stereo cameras configuration to generate depth map. The goal of our research is to design a FOD framework using a depth map to find block with certain area. Our fundamental approach is to use stereo matching algorithm to get the disparity information based on intensity images from stereo cameras and using the camera model to retrieve the distance information. From the result of our experiments, the proposed framework has a better performance with higher detection rate with lower false alarm. The processing speed is boosted significantly.

BIBLIOGRAPHY

Administration, N. H. T. S. et al. (2012). Visual-manual nhtsa driver distraction guidelines for in-vehicle electronic devices. *Washington, DC: National Highway Traffic Safety Administration (NHTSA), Department of Transportation (DOT)*.

Bertozzi, M., Broggi, A., Felisa, M., Vezzoni, G., and Del Rose, M. (2006). Low-level pedestrian detection by means of visible and far infra-red tetra-vision. In *Intelligent Vehicles Symposium, 2006 IEEE*, pages 231–236. IEEE.

Ćirić, I. T., Ćojbašić, Ž. M., Nikolić, V. D., Igić, T. S., and Turšnek, B. A. (2014). Intelligent optimal control of thermal vision-based person-following robot platform. *Thermal Science*, 18(3):957–966.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE.

Ding, Y. and Xiao, J. (2012). Contextual boost for pedestrian detection. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2895–2902. IEEE.

Dollar, P., Wojek, C., Schiele, B., and Perona, P. (2012). Pedestrian detection: An evaluation of the state of the art. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(4):743–761.

Gavrila, D. M. (2007). A bayesian, exemplar-based approach to hierarchical shape matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(8):1408–1421.

Huang, C., Ai, H., Li, Y., and Lao, S. (2005). Vector boosting for rotation invariant multi-view face detection. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 446–453. IEEE.

Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A., et al. (2011). Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM.

Joachims, T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

Kauff, P., Atzpadin, N., Fehn, C., Müller, M., Schreer, O., Smolic, A., and Tanger, R. (2007). Depth map creation and image-based rendering for advanced 3dtv services providing interoperability and scalability. *Signal Processing: Image Communication*, 22(2):217–234.

Lee, Y., Kim, T. G., and Choi, H.-T. (2014). A new approach of detection and recognition for artificial landmarks from noisy acoustic images. In *Robot Intelligence Technology and Applications 2*, pages 851–858. Springer.

Mertz, C., Navarro-Serment, L. E., MacLachlan, R., Rybski, P., Steinfeld, A., Suppe, A., Urmson, C., Vandapel, N., Hebert, M., Thorpe, C., et al. (2013). Moving object detection with laser scanners. *Journal of Field Robotics*, 30(1):17–43.

Mohammadpoor, M., Abdullah, R. R., Saleh, A. A., and Al-Dabbagh, M. D. (2013). A bistatic linear frequency modulated radar for on-the-ground object detection. *Electromagnetics*, 33(2):153–177.

Papageorgiou, C. and Poggio, T. (2000). A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33.

Sanger, T. D. (1988). Stereo disparity computation using gabor filters. *Biological cybernetics*, 59(6):405–418.

Schapire, R. E. and Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37(3):297–336.

Spinello, L. and Arras, K. O. (2012). Leveraging rgb-d data: Adaptive fusion and domain adaptation for object detection. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 4469–4474. IEEE.

Tang, S., Andriluka, M., and Schiele, B. (2012). Detection and tracking of occluded people. *International Journal of Computer Vision*, pages 1–12.

Tawari, A., Sivaraman, S., Trivedi, M. M., Shannon, T., and Tippelhofer, M. (2014). Looking-in and looking-out vision for urban intelligent assistance: Estimation of driver attentive state and dynamic surround for safe merging and braking. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 115–120. IEEE.

Wang, Z., Yoon, S., Hong, C., and Park, D. S. (2014). A novel svm based pedestrian detection algorithm via locality sensitive histograms.

Zhu, J., Zou, H., Rosset, S., and Hastie, T. (2009). Multi-class adaboost. *Statistics and its Interface*, 2(3):349–360.