Reducing False Triggers In Surveillance Systems Using Sensor Fusion

Madeleine Boström
dat11mbo@student.lu.se

Tobias Claesson
ada10tcl@student.lu.se

March 9, 2017

Master's thesis work carried out at Axis Communications AB.

Supervisors: Kalle Åström, kalle@maths.lth.se
Magnus Oskarsson, magnuso@maths.lth.se
Mazdak Farzone, mazdak.farzone@axis.com

Examiner: Carl Olsson, calle@maths.lth.se

Abstract

Sensor fusion has been widely adopted in the last couple of years, especially in the automobile industry. Their main goal is to gain a more robust system and increase security by e.g. predicting and preventing collisions.

Surveillance systems, based on video motion detection, face similar issues by having numerous problems with false triggers, particularly when there are big variations in the lighting of the scene, e.g. shadows or light beams. To address this issue, the effect of adding a radar sensor, whilst the video system is used as a black box, is investigated.

There exists a presentiment that a number of detections that are identified should not decrease noteworthy, as the two different systems complement each other. The validation is not necessarily identical with reality, however, it is a clear indication that sensor fusion is more reliable than using only video motion detection.

Keywords: Sensor fusion, Information fusion, Radar, Video, Motion detection, Surveillance system

Acknowledgements

We would like to express our gratitude to Axis Communication AB and the PCNI team for giving us the opportunity to do this master's thesis and making us feel as a part of the team and deepened our understanding of radar.

Special thanks to our supervisors Kalle Åström and Magnus Oskarsson for their guidance and useful comments throughout this thesis and to our supervisor at Axis, Mazdak Farzone, for introducing us to the topic as well as for all the support and help on every step of the way.

Furthermore, we would also want to thank team Analytics and especially Martin Ljungqvist for the help throughout the project. And to Aras Papadelis for helping us interpret radar data.

Last but definitely not least, we would like to thank the other master's thesis students for making our time at Axis a highly enjoyable time.

Glossary

CW Continuous Wave.

FFT Fast Fourier Transform.

FM Frequency Modulation.

FMCW Frequency Modulated Continues Wave.

FOV Field of View.

FPS frames per second.

MOTE Motion Object Tracking Engine.

PDF Probability Density Function.

RCS Radar Cross Section.

VMD Video Motion Detection.

Contents

Chapter 1

Introduction

In surveillance systems, the video motion detection software plays a significant role. Modern surveillance cameras with motion detection are widely used for monitoring break-ins and to prevent crime, where the system notifies if any unexpected activity is detected. However, there are times when the surveillance system has difficulties to distinguish objects i.e. under poor light or weather conditions. There are also problems that the system notifies when there is no human activity, e.g. dynamic backgrounds such as wind beams in trees. If this occurs too often the security of this system is impaired.

1.1 Main Objective

The primary focus of this master's thesis is to investigate how combining radar with classic motion detection could reduce false triggers, the ideal would be to only find human activity and nothing else. Particularly this thesis will focus on the fusion process after video and radar analysis has been calculated. In other words, there should not be any need to modify current systems, only add extra functionality, see Figure 1.1. The scenes this master's thesis will analyze are outdoors and called sterile, i.e. scenes where there are only one or a few people at the same time. Furthermore, the camera setup is assumed to be installed in a fixed position and will not be moving around.

In order to combine the two sensors in a way that will reduce the number of false triggers, the individual sensors' advantages and disadvantages need to be found. Some of the sensor characteristics are documented, however, for a complete comparison, some test scenarios need to be set up. Furthermore, there are some questions that need to be investigated, for instance: Which sensor should the system rely on? Are there external factors that can affect the sensors' reliability? Is it possible to keep the true positives at

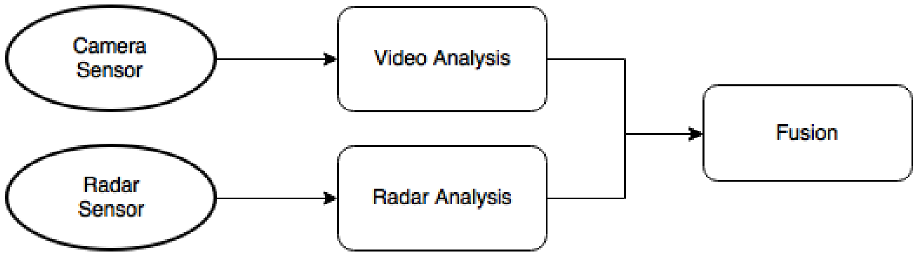


Figure 1.1: The flowchart describes the system's pipeline

the same level while reducing the false positives?

1.2 Problem Description

This thesis will be done with a video motion detection system used as a black box, meaning that it can only be viewed in terms of inputs and outputs, the internal logic is unknown. The video motion system is normally installed on the camera and works directly with the video stream as an input. Therefore there is no possibility for the radar sensor to communicate and affect the video motion system.

The video motion system has known problems including false alarms, missed alarms, and incorrect tracking behavior, the primary focus will be on improving the false alarms but it is important that the number of missed alarms does not increase.

The video motion system outputs tracked sources, in order to evaluate the two systems on an equal level and fuse the results, the radar information needs to be clustered and tracked. These algorithms need to be implemented since there is not any clustering or tracking implemented as of yet.

Even though the result could be greatly affected by different implementations of these steps it is not in focus of this thesis. As stated in Section 1.1 the focus is to examine if the result can be improved by adding a radar sensor.

It is possible to look at the problem of reducing false triggers in different ways, either that the correct frames trigger or that the trigger is located correctly in the frames. These different approaches will both be investigated to some extent. Because the tracking behavior is not in focus, the first approach will be of greater importance.

1.3 Utilities

This master's thesis will be using an Axis Communications AB (referred to as Axis) camera setup and motion detection software, also known as, Motion Object Tracking Engine (MOTE), as a base setup.

1.3.1 Hardware

The camera setup consists of an Axis F41 main unit [1] together with an Axis F1005-E sensor unit [2]. A radar system with a 24 GHz radar sensor was attached. During approximately half of this thesis time, the radar sensor was located above and slightly to the right of the camera sensor and openly displayed, as can be seen in Figure 1.2. Due to the importance of test scenes with diversified weather conditions, the radar sensor was placed inside a plastic box together with the camera sensor.

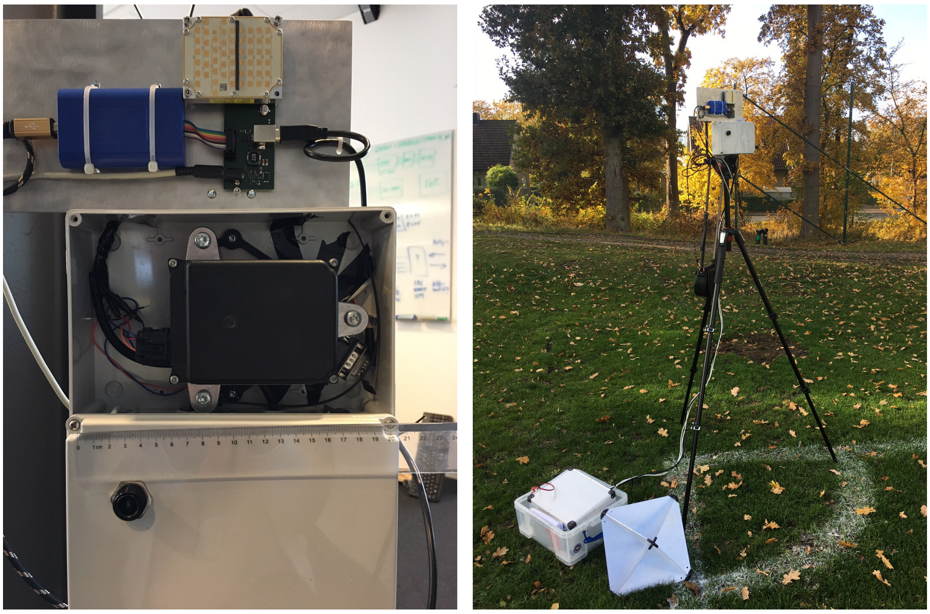


Figure 1.2: Camera setup used while radar sensor is exposed. Left image is a closeup of the sensors while the right image shows the entire camera setup.

The radar system can detect an object in a 2-dimensional environment by detecting and separating objects according to their speed, range and azimuth angle. The radar system has a documented $\pm 55°$ angle measurement range and $\pm 1°$ angle accuracy. Furthermore, it has a detectable radial speed from 0 km/h up to 34 km/h with a speed resolution of 0.15 m/s and detects a person (Radar Cross Section (RCS) = 0.75m^2, see Section 2.1.1) up to a distance of 50 m from the radar unit, with a distance resolution of 0.9 m.

1.3.2 Software

The image analysis will be performed by Axis' own software called MOTE, that normally is available within the camera's firmware. This thesis will simulate MOTE offline on a computer which will give additional information to the resulting tracked object masks, in the form of a motion and a foreground mask for each frame. These masks can be made available as outputs from the system in a future revision of MOTE if necessary.

The parameters used when running MOTE offline will be kept the same as the system running on cameras today. Specifically an analysis resolution of 480×270, an analysis frame rate at 10 frames per second (FPS) (whilst the output is kept at 5 FPS) and grayscale images. All the implementation will be done in MATLAB.

Axis has a program called Video Motion Detection (VMD), which lets the user filter results from MOTE by the individual needs e.g. include/exclude areas and minimum object duration to trigger an alarm. There may arise some confusion because of the name but this thesis will not include Axis' VMD but assume some similar software will be able to filter the output generated from this thesis.

1.4 Limitations and Delimitations

At the time of this thesis there is no available radar system running on a camera at Axis, therefore the radar and video stream will be saved and later used offline on a desktop computer. Hence the real-time performance will not be taken into account. Unfortunately, the radar and camera stream is recorded separately without timestamps, resulting in a manually synchronization. There are no guarantees that the data is correctly synced in time, which might impact the accuracy of the fusion.

The radar operates slightly slower than 5 FPS, whilst, MOTE has a specified output of 5 FPS. This increases the difficulty of manually synchronizing the two streams, since there is no direct mapping between a radar frame and a camera frame. This leads to that the fusion could to be more difficult when there exists a synchronization error especially in scenes where two objects are close to each other.

The motion detection software does not have a limitation in distance only a minimum object detection size of $1.3\% \times 2.2\%$ of the image with the used camera chip ARTPEC-5. The maximum velocity of a moving object to be detected by the same software is 1/2 FOV/s (Field of View per second). In comparison, the radar system has a documented distance limitation of 50 meters, although it may happen that objects further away are detected.

The radar sensor used has a documented Field of View (FOV) 110° while the practical FOV is quite larger, see Figure 1.3. Since the camera sensor's FOV is 113° the same FOV can be used for radar as well and still maintain a good accuracy.

The radar sensor used during this thesis is not calibrated, which results in some larger angle and distance errors than documented. The magnitude measurement has also been affected, therefore harder to use and the time spent on the RCS values has been limited. Since the angle measurement is of greater importance when combining the two sensors, the calibration graph in Figure 1.3 has been taken into account and the angle measurements from the radar have been adapted to counteract the angle errors.

What should trigger the alarm is often well defined, however, there are gray areas where the alarm might need to trigger. For example, humans or cars should trigger the alarm while birds and rabbits should not. One might argue that bigger or dangerous animals such as a bear might need to trigger the alarm for safety reasons. This topic is out of the scope for this thesis and will therefore not be discussed further, the main focus is to trigger on human activity and not trigger on light rays, shadows or trees.

The radar information will give a new perspective which could be used for improving

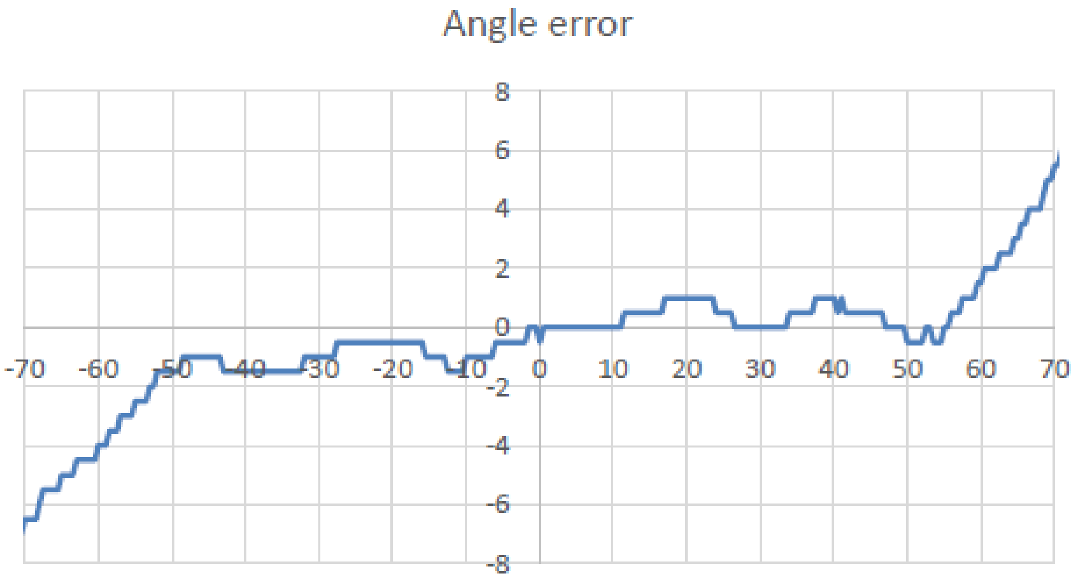


Figure 1.3: Calibration graph for angle measurements

the tracking behavior, but will not be prioritized due to the limited time of this thesis.

In the beginning of the thesis, there were hopes of being able to have the camera setup filming during a longer period of time, in order to capture more realistic scenarios. Due to reasons out of the hands of the authors, this was not possible resulting in only stage-managed scenarios, which may induce bias in the dataset.

1.5 Outline

The outline for the rest of this thesis is as follows:

Chapter 2 Relevant background theory will be introduced, along with concepts that are of importance for this report.

Chapter 3 The gathering process along with a description of the retrieved data is described in detail in this chapter.

Chapter 4 In this chapter the different steps and methods are described.

Chapter 5 The results obtained will be presented in this chapter.

Chapter 6 Have the purpose been fulfilled and what possibilities exist for future works.

Chapter 7 Conclusion

Chapter 2

Background

In this chapter, some of the theoretical backgrounds that this thesis is built upon or uses is described, along with concepts that are of importance for this report.

2.1 Radar Technology

RAdio **D**etection **A**nd **R**anging (RADAR) is an object-detection system that has been used in the military since World War II. Radio waves are used to determine the range, angle, or velocity of objects [3]. The modern uses of radar are highly diverse including anti-missile systems, public health surveillance and it has recently become widely used in the automobile industry, partly to detect people or other vehicles for collision warning and emergency braking [4].

The simplest form of a radar system consists of a radio transmitter and a receiver. The transmitter radiates electromagnetic energy which is reradiated in many directions. When an object intercepts the wave, some of the reradiated (echo) energy is collected by the receiver [5]. Since electromagnetic energy travels at the speed of light, the range or distance to a target can be obtained by measuring the time it takes for the radiated energy to travel to the target and back [3]. The distance R [m] is given by

$$R = \frac{c_0 \cdot \Delta t}{2} \tag{2.1}$$

where c_0 is the speed of light in free space ($3 \cdot 10^8$ m/s) and Δt is the round-trip time [s].

There are two types of radar signals; pulse, which transmits a sequence of pulses, and Continuous Wave (CW), which transmits a continuous signal. The later one usually uses separate transmit and receive antennas. Because it is hard to receive with full sensitivity through an antenna while it transmits a high power signal. Pulsed radar on the other hand only needs one antenna, the pulsed radar emits short powerful pulses during transmit time, waits for echo, and then sends out the next pulse. In contrast to the CW radar, the transmitter is switched off until the measurement is finished. [3, 6]

There is no way to determine target range with a continuous wave radar without Frequency Modulation (FM). Because of the lack of necessary time marks to accurately calculate the round trip time Δt in Equation 2.1. Such a time reference can be obtained by using Frequency Modulated Continues Wave (FMCW) radar, which can modify the operating carrier frequency periodically. The basic idea is to use a linear frequency ramp as a transmit signal. When an echo signal is received, a delay Δt is obtained by the change of frequency. The Equation 2.1 can be rewritten as

$$R = \frac{c_0 \cdot \Delta t}{2} = \frac{c_0 \cdot |\Delta f|}{2 \cdot (df/dt)} \tag{2.2}$$

where Δf is the measured frequency difference [Hz] and df/dt is the frequency shift per unit of time. [7]

For CW-radars the velocity can be given as a Doppler shift since the transmitter of the radar emits a signal at a constant frequency. Therefore, the velocity can be calculated by the frequency shift in the returning echo. When an object moves directly towards the radar, the receiver frequency increases at a faster rate compared to an object that moves in another direction. This is called the Cosine Effect and explains why the Doppler speed can be slightly slower than the actual speed. [8]

In order to determine both range and velocity with FMCW-radars, the information from one frequency ramp is not sufficient, because it is ambiguous. To separate the frequency shifts generated by range from the Doppler frequency, several subsequent ramps are needed. To do this Fast Fourier Transform (FFT) can be used. For more detailed information the reader is encouraged to read *"Range Doppler Detection for automotive FMCW Radars"* by Volker Winkler. [9]

2.1.1 Radar Cross Section

Radars can sometime classify detected objects depending on the returned echo signal. The Radar Cross Section (RCS), denoted as σ, is a measurement of the target's ability to reflect radar signals in the direction of the radar receiver. RCS depends mainly on the size, geometry, and material of the object, where a high value of RCS indicates a high visibility. [10]

The RCS value can be solved by first modifying the radar equation:

$$P_r = \frac{P_t G_t}{4\pi r^2} \sigma \frac{1}{4\pi r^2} A_{eff} \tag{2.3}$$

such that the RCS value σ is given by

$$\sigma = \frac{P_r r^4}{c} \tag{2.4}$$

where P_r is power received back from the target by the radar, r is the distance from the radar to the target and c is a constant depending on the radar sensors abilities see Equation 2.5, where P_t is power transmitted by the radar, G_t is gain of the radar transmit antenna and A_{eff} is the effective area of the radar receiving antenna. [11]

$$c = \frac{P_t G_t A_{eff}}{16\pi^2} \tag{2.5}$$

The RCS value of the target can be seen as a comparison between, the strength of the reflected signal from the target and the reflected signal from a perfectly smooth sphere, with a cross-sectional area of 1 m^2. The echo of a sphere is independent of the observation angle, compared to the echo of a human that is dependent on the observation angle and varies significantly. [10, 12, 13]

In theory, the RCS does not vary with the distance. However, due to the FOV of the radar and the compensation for the free space path loss, the measured RCS varies significantly over distance. Furthermore, at near distances, it is not certain that the target is scanned over its complete height. [13, 14]

2.2 Video Motion Detection

There are many approaches for detecting motion in a video stream. All of them are based on comparing the current video frame with the previous one or with a background model. One of the most common and the simplest form of a video motion detection system is to compare the current video frame in grayscale with the previous one and calculate the difference for each pixel. If the pixel difference is greater than a selected threshold there is motion in this area. It is also fairly common to set a threshold for the least amount of connective pixels with motion detected that should be allowed. [15]

Another common approach is to save the first video frame as a background model and then calculate the pixel differences from the current frame to the background model instead [15]. Drawbacks that can occur with this is when the background is changed i.e. a parked car drives away. This means that the system sees the absence of the car as a new object that is stationed where the car was before, a so-called ghost object.

The simple motion detection described in the first paragraph has some problem with slow moving objects, while foreground detection has some other problems as mentioned and therefore it is good to combine these two methods. MOTE uses a combination of the two but with an adaptive background model, with a decision matrix, that is based on the advantages and disadvantages of motion and foreground detection.

2.3 Clustering

The goal of clustering is to divide data into groups based on some measure of similarity or characteristics. As a result, objects in the same cluster are similar and objects in different clusters are distinct. There are various clustering algorithms and they differ significantly in their notation of what constitutes a cluster. There are two broad groups of clustering algorithms, hard and soft clustering. Hard clustering, where each data point only belongs to one cluster. Soft clustering, where each data point belongs to each cluster to a certain degree. [16, 17, 18]

There are lots of different clustering algorithms and models. One of the most common is the K-means clustering algorithm, which belongs to the centroid model techniques, each cluster is represented by a single mean vector. K-means aim to divide n observations into K clusters, each observation belongs to the cluster with the nearest mean. The K-means is simple and the method is described in Algorithm 1, where K is user-specified, namely, the number of clusters desired. However, it is sensitive to outliers, therefore it is a good idea to discover and eliminate them beforehand. [16, 18, 19]

Algorithm 1: Basic K-means algorithm

```
Select K points as initial centroids;
repeat
    Form K clusters by assigning each point to its closest centroid;
    Recompute the centroid of each cluster;
until Convergence has been reached;
```

Another widely used clustering algorithm is the hierarchical clustering, which belongs to the connectivity models, and builds on distance connectivity. There are two basic approaches for generating a hierarchical clustering. Agglomerative (bottom-up) approach, which starts with all points as individual clusters, is by far the most common hierarchical approach. For each step the closest cluster pair merge, a notation of proximity is required, see Algorithm 2. The divisive (top-down) approach, starts with one cluster (including all points). For each step, a cluster is divided into new clusters, until only clusters of individual points remain. Hierarchical clustering can be displayed graphically with a dendrogram, which is a tree-like diagram. The dendrogram shows both clusters, sub-clusters, their relationship and the order of merge/split. [16, 18]

Algorithm 2: Basic agglomerative hierarchical clustering algorithm

```
Compute the proximity matrix, if necessary;
repeat
    Merge the closest two clusters;
    Update the proximity matrix to reflect the proximity between the new cluster
      and the original clusters;
until Only one cluster remains;
```

In agglomerative hierarchical clustering, see Algorithm 2, there are different ways to calculate the proximity between the clusters, this can be done using linkage methods. The following three linkage methods differ in how the distance between two clusters is measured: [16, 18]

Single linkage (or nearest neighbor) calculates the distance as the shortest distance between a point in each cluster.

Complete linkage calculates the distance as the longest distance between a point in each of the clusters.

Average linkage the distance is calculated as the average distance between each point in one cluster to every point in the other cluster.

2.4 Tracking

Tracking is the process of associating and locating objects over consecutive time frames. There can be detection at consecutive frames, without a guarantee that they are relevant. For instance, some of them could come from the object of interest and some could come from other objects or noise. Therefore there is a need for a model over how an object change over time. Objects that are moving fast in relation to the frame rate can be aggravating for this process, as well as frequent orientation changes. [20]

Two common ways of tracking objects are tracking by detection and tracking by matching. The first method requires distinct characteristics between the objects, such that the algorithm can separate and associate the objects, for example, a red and a blue ball on a green background. The algorithm only needs to search for red or blue pixels in the image and match that position with previous frames. [20]

The second method utilizes how the object moves. That is, if an object moves from (x, y) in frame i to $(x + d_1, y + d_2)$ in frame $i + 1$, were d_1 and d_2 are chosen based on the frame rate and the distance that an object could have moved to during the time period. Due to missing data points, for each track, there can be frames where there are not any measurements. Thus, there is a need to estimate where the object could have been and use that point for the next frame to see if the track is still alive. How long a track should live without a new measurement is based on how reliable the system is to have a measurement of existing objects. Furthermore, there exists a possibility in which two measurements could be assigned to two tracks, in that case, one could implement a method to find which one is more likely to be assigned to each track, a simple example being the shortest distance. [20]

2.5 Information Fusion

In some cases, the term information fusion is used to describe a fusion where the data has been preprocessed. Whereas, data fusion is used to describe a fusion where the fusion is based on raw data; usually the terms are used as synonyms. [21] In this thesis information fusion will be used and referred to as the fusion of preprocessed data. Another commonly used term is sensor fusion which describes a fusion of different sensors regardless if the data has been preprocessed or not.

When utilizing sensor fusion, the goal is to achieve lower detection error rate and increase the reliability of the system. [21] There is also a possibility to decide which level of uncertainty is accepted, such as when a car decides if it can drive in a specific lane, there must be a high level of certainty that e.g. there is not a pedestrian walking there. To achieve this, a requirement could be that both sensors must decide that it is safe to drive there. This leads to some miss-classifications, yet the goal is to not have a collision. [22]

2.6 Parameter Estimation

There might be times when it is desirable to estimate points between two points in order to describe a discrete system as if it was continuous. For instance, assume a set of timestamps $S = s_1, ..., s_N$ where one or many measurements x_k are given at the timestamps $t_k \in S$, with $k = 1, ..., n$. For each timestamp s there can be zero, one or many measurements. Consequently, an algorithm to generate a smooth curve f in the interval $[t_1, t_n]$ is needed. This can be achieved by minimizing the minus log likelihood of the measurements (t_k, x_k) assuming that $x_k = f(t_k) + e_k$ where e are random Gaussian variables with mean 0 and variance σ. Furthermore, there is a prior e.g. $f_{i+1} = f_i + e_i$ where e are random Gaussian variables with a mean of zero and a variance σ. Denote $z = \begin{pmatrix} f_1 \\ \vdots \\ f_N \end{pmatrix}$ and find z that minimizes

$$g(z) = \sum_{k=1}^{n} |x_k - f(t_k)|^2 + \sum_{i=1}^{N-1} |f_{i+1} - f_i|^2$$

which is a linear least squares problem and can be solved using linear algebra.

2.7 Probability Theory

Probability theory is the mathematical study of random or uncertain events, including various methods to describe and calculate random events. Events and outcomes are often studied in form of probability distributions, describing how likely an outcome is.

Probability is the ratio of how many times an outcome can occur compared to all possible outcomes. The probability is denoted by a number in the closed interval from 0 to 1, where an event's occurrence or failure to occur is random. The intervals extremes represent impossible and certain respectively.

2.7.1 Normal Distribution

A normal distribution with expected value μ and variation σ^2 has the Probability Density Function (PDF):

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

The curve is symmetrical around the mean of the distribution and the probability of a value to be within two standard deviations from the mean value is roughly 95%. Data which has; strong tendency to be close to the expected value, equally probable deviations on both sides and decreasing occurrence frequency for values further away, can be well described with normal distribution. [23]

2.7.2 Logistic Distribution

A logistic distribution with expected value μ and a scale parameter proportional to the standard deviation s has the PDF:

$$P(x) = \frac{e^{-\frac{x-\mu}{s}}}{s\left(1 + e^{-\frac{x-\mu}{s}}\right)^2}$$

It is similar to normal distribution in shape but with longer tails and higher kurtosis. Hence, logistic distribution is more tolerant when there are more big misses but still most values are focused close to the expected value. [23]

2.7.3 Bayes' Theorem

Bayes' Theorem is a mathematical formula used for calculating conditional probabilities, and is written as:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)} \tag{2.6}$$

Where $P(A \mid B)$ is the conditioned probability that A occur given that B is true and $P(B \mid A)$ is the reverse. $P(A)$ and $P(B)$ are the probabilities of A and B individually without regard to each outer. [24]

2.8 Classification Tree

A decision tree can be used to classify different problems to given class labels. When used for classification problems it is more fitting to refer to the tree as a classification tree. One way to answer a question, such as *"What kind of animal is this?"*, is to use a classification tree. The procedure is to ask a new question such as how many legs the animal has or if it has a tail. For each question answered a new question should be asked until a conclusion is reached of what animal it was. The sequence of questions and the possible answers can be represented as a classification tree, which is a hierarchical structure with different nodes. Namely, one root node symbolizing the starting point, internal nodes which can be in multiple layers, each node with multiple outputs and lastly, the leafs nodes which indicate the classification. [25, 26]

2.9 Related Work

During the last couple of years, there has been a lot of work on combining both video and radar information, especially in the automobile industry. These two technologies complement each other well as radar is excellent for measuring range and Doppler speed, while cameras recognize objects and measures lateral movement well. [27]

In the automobile industry, it is important to predict where cars and pedestrians will be in the near future to avoid a collision. This is crucial in achieving safety in traffic. This area has been researched a lot with, but not limited to, radar and video fusion. [28, 29] This is not implemented in this project, but it could be used if there is a part of the area covered by the system that is a restricted area. For this problem, the system could trigger the alarm if a pedestrian is predicted to go inside the restricted zone.

There has also been work done for surveillance systems using sensor fusion. Particular one paper focused on data fusion where the measurements from different sensors, that monitors the same scene, are fused together to obtain a more accurate estimation of the tracked object. The system can also monitor a bigger area with multiple sensors, consequently, a track can still be continuous when an object moves outside the FOV of a sensor if another sensor covers that area. [30]

Furthermore, there is another master's thesis work at Axis that is closely related, which combines radar and video using data fusion, with the focus on tracking objects in world coordinates.

Chapter 3

Data

In this chapter, the data from the different sensors are described. Data retrieved from radar and camera are in different dimensions, which means that fusion between them is not a trivial problem. Additionally, the dataset used during this master's thesis is described.

3.1 Sensor Data

The output information retrieved by MOTE is in the form of polygons representing the tracked objects, with x and y coordinates for pixel position. The output representing objects' position from the radar sensor are the distance in meters and angle in azimuth degrees. The angle is converted to image coordinate x using the camera lens correction matrix. Due to the lack of the object's elevation knowledge, some errors occur in addition to the angle scattering.

In addition to the object position, the radar outputs two more parameters, velocity and magnitude. The velocity is the object's Doppler speed in meters per second and the magnitude is the loss of signal strength in decibel (dB). It is possible to calculate the RCS for the objects, using Equation 2.4, where P_r represent the magnitude.

The output from the radar sensor can consist of one or many hits per detected object depending on the range, material, velocity and size of the object. In Figure 3.1 the radar data is visualized from a camera perspective and radar perspective. Unfortunately, the radar data is preprocessed in partly unknown ways, what is known is that some clustering with discrete steps is performed.

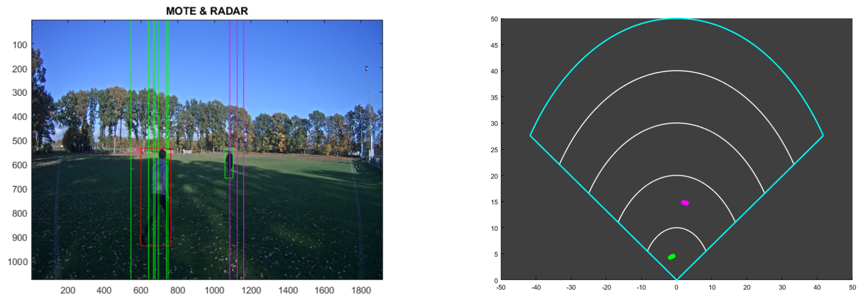


Figure 3.1: Representation of radar data with and without the camera's perspective. The blue lines in the right image represents ±57.5° FOV and 50 m from the radar, each intermediate line is an increment of 10 m.

3.2 Sensor Fusion

The camera and radar sensors are heterogeneous; it is not trivial to combine them. However, there are benefits to gain, a summery of the sensors advantages and disadvantages can be seen in Table 5.1. Clearly radar seems like a good complementary sensor to video.

Table 3.1: Radar and video characteristics

Characteristic	**Radar**	**Video**
Good azimuth detectability	-	✓
Background model	-	✓
Good Doppler detectability	✓	-
Only physical objects	✓	-
Distance	✓	-

Surveillance systems have different uses of radar than e.g. the automotive industry. For instance, a car needs to know the exact location in both distance and angle in order to know when to use the breaks. Whilst, the exact distance or angle is not as crucial for a surveillance system.

Furthermore, the radar can detect objects just as clear at night time as in day time. Therefore, if an area that is not sufficiently illuminated it can be monitored both day and night with sensor fusion. The goal with adding another sensor dimension is to achieve a higher level of certainty when decisions are made.

3.3 Datasets

All data used for analyzing and testing has been produced throughout the project since there was not any material available created with the correct combination of hardware. The data was gathered at multiple locations and dates to get a broader variety of data. However, as mentioned in Section 1.4 it was not possible to perform any long time recording, which probably would have resulted in more realistic empty scenes with some false triggers.

The film scenarios consist of mostly sterile scenes where one or two persons moves in a variety of ways to simulate how a typical surveillance scene could look like. Many of the filmed scenarios are based on parts of the i-LIDS sterile zone monitoring datasets, which is the UK government benchmark for video analytics systems [31]. In contrast, the scenes used during this thesis are shorter and contains mostly the parts with actual moving objects.

In addition, film scenarios that are known to be problematic for MOTE have been attempted to be recreated. Including scenes with light variations, reflections, and dynamic backgrounds. Furthermore, film scenarios for easier analyzing of radar data were created, using environments which avoid generating numerous radar reflections.

Finally, a workshop was held, around the question ”What are the most critical test scenarios that should be evaluated?”, to increase the likelihood of a dataset containing the most critical test scenarios. The workshop was performed using the KJ-Method, which allows groups to quickly reach a consensus on priorities of subjective, qualitative data [32].

For a complete list of film scenarios see Appendix A.1. The films have been combined into different smaller datasets, to test different characteristics and to reduce the skewness of the complete dataset. Eight different datasets have been used which can be seen in Appendix A.2. Focus will be on four of them *Mixed dataset*, *Mixed without dark dataset*, *Filtered dataset* and *Dark dataset*. The results for the remaining datasets will be presented in Appendix B without a detailed discussion.

Chapter 4

Methodology

This chapter describes the methods used in order to combine the sensors with data described in Chapter 3, and how the evaluation was done. All implementation has been done using MATLAB on a desktop computer.

4.1 Synchronization of Data

The radar and video are recorded separately, as described in Section 1.4, in consequence, the data need to be manually synchronized. This is not a trivial problem, hence there are multiple factors that affect the dispersion of radar hits. Influencing factors are; uncalibrated radar sensor, angle accuracy, image skewness due to the fisheye camera, frame rate disparities between sensors and also the uncertainty about which part of the object the radar hit represents (e.g. left or right arm).

Initially, an assumption has been made, that radar hits are normally distributed around the object's center of mass. With this in mind, an estimated density function of the radar hits' dispersion has been calculated. Based on the expected value and the assumption that it should be zero, readjustments over time was made, until the expected value was sufficiently close to zero.

All the synchronization has also been verified visually by the authors to avoid corrupt data. The difficulties with synchronization increases if there are multiple objects in the scene. In scenes where objects are moving in opposite direction to each other, visual synchronization has almost exclusively been used. Because their error caused by faulty synchronization in time, cancel each other out.

4.2 Radar Filter

Only radar hits within 50 m are kept and the rest are filtered out, see Section 1.4. Additionally, all objects with a speed slower than ± 16 cm/s are filtered out as well. Because the used FMCW radar antenna have problems to distinguish between moving and stationary objects in this area. This has been done since detections with these velocities was found in all scenes and nearly all frames, even empty ones.

4.3 Radar Cluster

Even though some clustering is done in the preprocessing phase an additional clustering is needed. The goal is to associate each real world object in the frame to one cluster. The clustering method is inspired by hierarchical clustering using average linkage, the criteria for a detection to be assigned to a cluster depends on the distance and angle measurements. When a new measurement has been added to a cluster the mean of the cluster is recalculated. The distance and angle differences are weighted differently depending on the distance, this was mainly done because the radar gives more hits on body parts on a shorter range.

There are some challenges with clustering the radar detections so that one cluster represents one object. For instance, to cluster a car as one object could be very similar to cluster two persons that are standing close to each other, as one object, see Figure 4.1.

4.4 Radar Tracker

The main focus for this thesis has been on sterile environments, see Section 1.1, therefore a simple tracker was implemented due to time limitations. Tracking by matching was best suited for this problem, considering the lack of a strong model for identifying objects.

The flowchart is shown in Figure 4.2 describes the process for a radar detection, until it becomes a track or classified as noise. When there are missing data, it needs to be estimated, this has been done linearly using the two last positions. The estimations are not included in the resulting track, it is only used to associate the track with new data points.

To limit the number of false triggers, a condition needs to be fulfilled before the radar detections are counted as a real object and a track is started. On the contrary, a hard condition will reduce the number of true detections and potentially miss objects before they are tracked. The condition was set to that at least two out of five frames need to have a detection. In order to limit the number of misses in the beginning of each object, the whole large practical FOV of the radar is used.

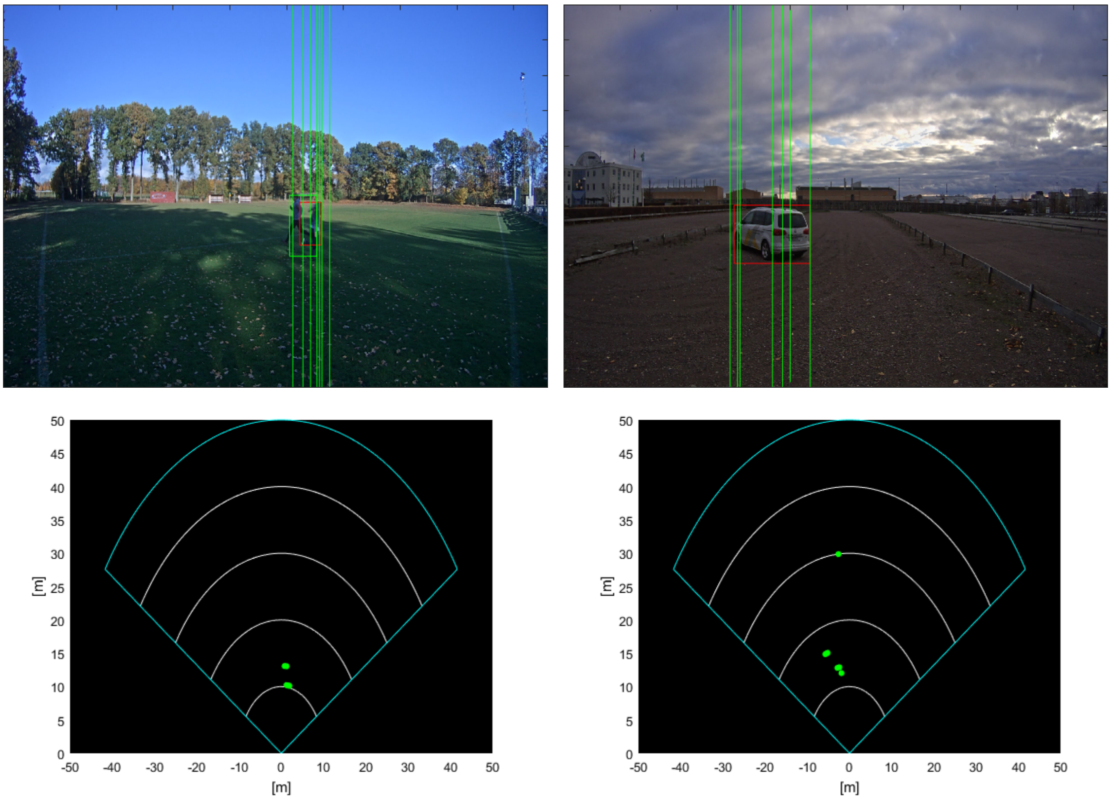


Figure 4.1: Visualization of a clustering problem. The left figures shows radar hits for two persons and the right ones shows radar hits for one car.

4.5 Ground Truth

Comparing the fusion system and MOTE with real world objects, also referred to as ground truth, is not a trivial task. The process of creating ground truth usually consist of at least some part manual creation which could lead to errors caused by human error. Ground truth was used both to analyze and validate data, there was a need for several types of ground truth, in order to evaluate different perspectives. The different types evaluate; radar tracker, how accurate the objects are located in each frame, how many objects are detected and the number of correctly alarmed frames.

Ground truth for radar tracker was generated by manually selecting all measurements associated with an object. This was done by dividing the data into different steps, where the measurements are represented as points from the different features; distance, angle, velocity and magnitude each over time and finally distance over angle. These were iterated through one at a time and only the chosen points from a feature were passed along to the next feature. In Figure 4.3, some of the different features described can be seen where there are distinct patterns for each object.

Radar can have multiple measurements per object and frame and sometimes it can fail to detect an object. To compensate for this, interpolation using parameter estimation with a smoothening was used. The evaluation is performed visually to see how the tracker

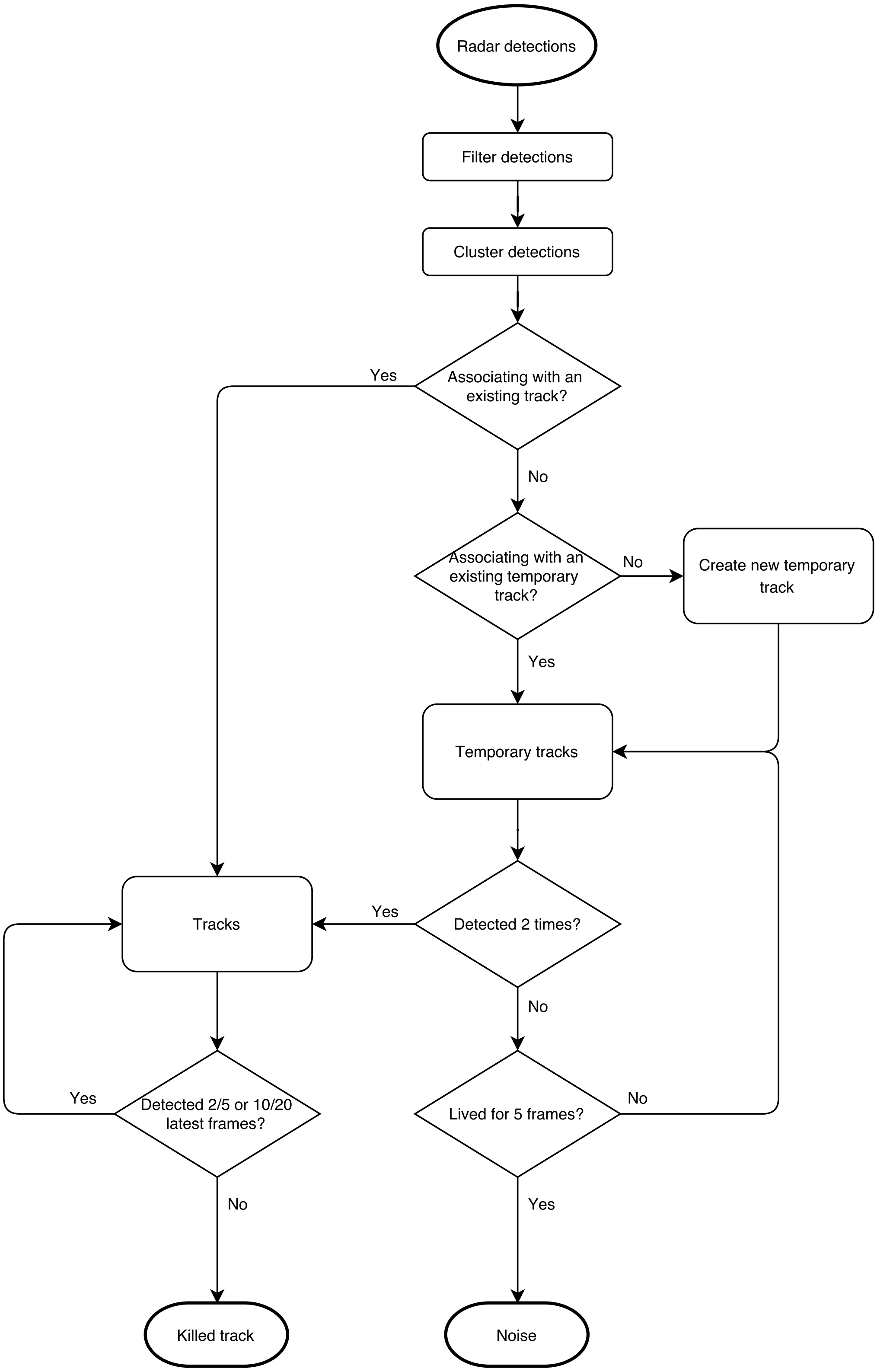


Figure 4.2: A flow chart that displays the different steps in the tracking algorithm and when a track will be killed or which clustered detections that will be seen as noise.

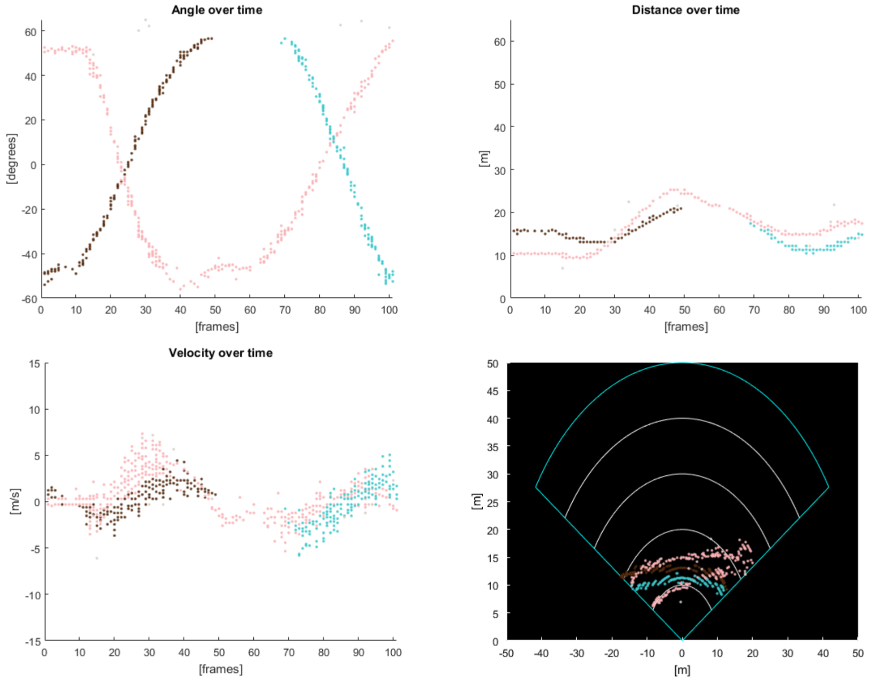


Figure 4.3: Images to show how the ground truth was generated based on patterns. The purple color is a person that starts from the left side and runs to the right side and then back again. Yellow and red color is a person that starts from the right side and runs to the left side and is outside of FOV for a couple of frames and then runs back to the right side. This example is from *Victoriastation 7* see Appendix A.1.2.

appears versus how the ground truth appears.

Ground truth for how accurate the objects is located was created similar to how the ground truth for the radar tracker was created. But instead of radar data, it was generated from the video sequence where the features are maximum and minimum x, y coordinates of each object for both foreground and motion masks. Next, the two ground truths are visually analyzed and the mask that matches the real object the best for different intervals throughout the film is selected for the respective interval. Parameter estimation is used here as well. Finally, if needed the position or size of the ground truth can be modified.

To test how many objects that were detected, a list of frame intervals was created for each film that each represents one object. Finally, to test how many correctly alarmed frames that were triggered, a similar list was created, but a frame can only belong to one interval since it is not of interest to know if there are one or five objects that trigger the alarm.

4.6 Data Analysis

In order to analyze possibilities of classifying objects using the measured magnitude from radar detections Equation 2.4 was used. Because of the uncalibrated radar sensor and lack of documentation about the radars internal values and underlying functionalities, the constant c given by Equation 2.5 needed to be estimated. This was done by assuming that the RCS for a human would be approximately 0.75m^2. In this thesis RCS is referred to as this calculated value and does not correspond to the actual RCS.

RCS values for human, car and dog was analyzed, from some selected scenes, over the distance. Lower and upper boundaries for the different objects' RCS values was created in order to see if it was possible to distinguish between them.

In order to analyze which sensor is best suited for different type of scenes, some kind of probability estimation needed to be calculated. Bayes' theorem was used to calculate the probability of an object given the information about sensor detections. First, two matrices was constructed P_{obj} and $P_{\neg\text{obj}}$ representing the number of sensor detections when there is an object and when there is not. The first matrix was designed as

$$P_{\text{obj}} = \begin{bmatrix} x_{0,0} & x_{0,1} \\ x_{1,0} & x_{1,1} \end{bmatrix}$$

where the value of i in $x_{i,j}$ represents if MOTE has detected the object correctly and value j represents the same for radar. The second matrix was similarly designed as

$$P_{\neg\text{obj}} = \begin{bmatrix} x_{0,0} & x_{0,1} \\ x_{1,0} & x_{1,1} \end{bmatrix}$$

where i and j represents which sensor that detected occurrences of faulty objects, and $x_{0,0}$ is the number of frames that did not have any faulty detections.

Using the matrices, the probability for object occurrences is given by

$$P(\text{obj}) = \frac{\sum P_{\text{obj}}}{\sum P_{\text{obj}} + \sum P_{\neg\text{obj}}}$$

where $\mathbf{1}^\intercal P_{\text{obj}} \mathbf{1}$ is denoted as $\sum P_{\text{obj}}$. Furthermore, the probability for a sensor combination, (i, j), to yield detections is given by

$$P(i,j) = \frac{P_{\text{obj}}(i,j) + P_{\neg\text{obj}}(i,j)}{\sum P_{\text{obj}} + \sum P_{\neg\text{obj}}}$$

and the probability for the sensor combination, (i, j), given the information that it is an object is given by

$$P(i,j \mid \text{obj}) = \frac{P_{\text{obj}}(i,j)}{\sum P_{\text{obj}}}.$$

Using Bayes' theorem the probability of an object given that the sensor combination, (i, j), is calculated by:

$$P(\text{obj} \mid i,j) = \frac{P(i,j \mid \text{obj})P(\text{obj})}{P(i,j)} = \frac{P_{\text{obj}}(i,j)}{P_{\text{obj}}(i,j) + P_{\neg\text{obj}}(i,j)} \tag{4.1}$$

Similar calculation yields the probabilities of a faulty detection.

These calculations have been performed on the different datasets described in Section 3.3. With the goal of finding which sensor that is more likely to be correct, for various characteristics of scenarios. These calculations have only been done on films in the datasets, due to the time consumption of generating ground truth for objects' position.

4.7 Decision Making

As stated in Section 3.2, video and radar has different strengths and weaknesses and complement each other in a very efficient way. However, different scenarios were analyzed to find how the sensors performed in various situations.

Due to, time and data limitations as well as the consistency of both sensors, as can be seen in Figure 4.4, the only special treated situation was when scenes were dark.

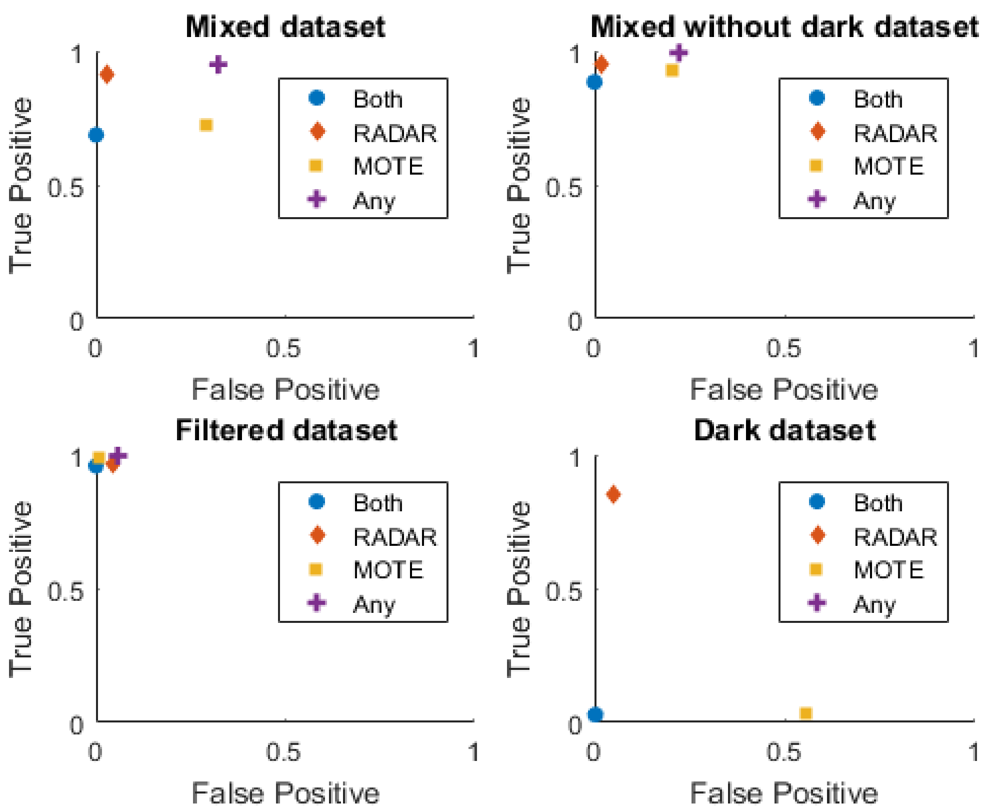


Figure 4.4: The performance of the sensors in various datasets.

As long as the scene is sufficiently illuminated, both sensors need to have detected the object in order to trigger. For how many frames the object should continue to trigger, is for the radar decided, in a similar fashion as to its' tracker. This is due to the fact that the radar can fail to detect objects in a couple of isolated frames, e.g., on larger distances or when the object has low Doppler speed. On the contrary, if this is needed for the image analysis, then it probably already has been implemented in MOTE.

4.8 Validation methods

Only films that were divided into data sets was tested based on where objects are located in each frame. This is a consequence of that it is very time-consuming to generate ground truth. The validation process was based on how big part of the detected objects' intersected with the ground truth when there was a MOTE object and if there were only a radar object the x position was compared with a specified threshold that was based on the dispersion of the radar hits.

Almost all scenes are used to test the amount of correctly alarmed frames and most of them are used to test how many correct objects trigger the alarm in each frame. The evaluation is performed on both MOTE and fusion to be able to analyze how the resulting system performs compared to MOTE. A list of frames the system should trigger in is compared to the actual frames that the system trigger in to count the correct number of trigger frames. Similarly, a list with the number of objects in each frame is used to count the number of objects that correctly trigger.

For different reasons, some of the films were cut. For example, in some films, there were objects already standing in the scene when the recording started, but MOTE utilizes the first four seconds with gathering information about how the background looks like and therefore the first few seconds will perform worse. Another example is when an object stands still for a longer period of time, it is not specified how the system should handle these situations and therefore not used when validating. Due to the limitation of 50 m for the radar module all scenes with objects further away than this is cut. This is done since otherwise MOTE would get false triggers when it triggers on an object further away, even though it is a human activity. The main reason why some scenes were cut is to make sure that fusion and MOTE is validated based on the same premises for a fair judgment.

Chapter 5

Result

In this chapter the different results obtained using the methods specified in Chapter 4 will be presented. Additional results for the performance of MOTE and fusion for individual scenes and datasets will be included in Appendix B.

5.1 Radar

The resulting radar tracker obtained by the simple method described in Section 4.4 was only visually validated, therefore, these results are not presented. However, its result will influence the fusion's forthcoming result. The resulting radar tracker is good enough since it is not important that each object belongs to only one track, since the fusion works even when the tracker id changes over time. However, the reliability of the fusion is slightly higher when the tracker id is constant. In Figure 5.1 an example of a resulting track over 125 frames can be seen.

The RCS values calculated can be seen in Figure 5.2 and the resulting upper and lower boundaries for human, car and dog can be seen in Figure 5.3.

5.2 Sensor and Fusion Validation

In this section the results obtained by the fusion will be presented, the validation will be performed in different ways as described in Section 4.8, this was done because reducing false trigger may be ambiguous. The goal was to present results for, correct frames, the correct number of objects in frames and also for correctly locating the objects in the frames.

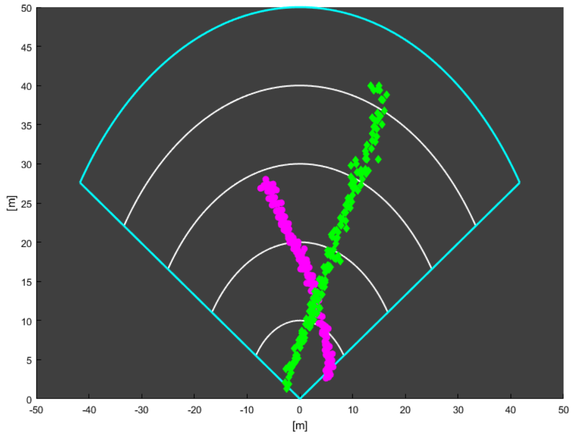
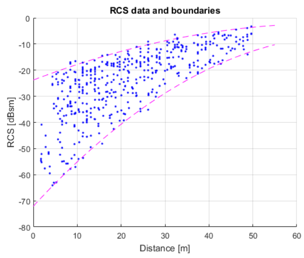
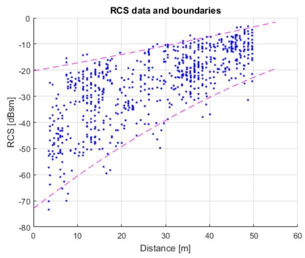


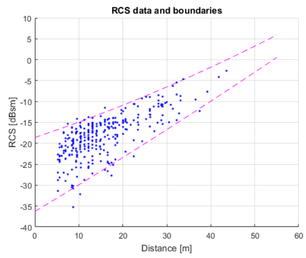
Figure 5.1: The output of a track of two persons which is later used in the fusion process. From the film *Victoriastadion 9* where one person walking from the right side diagonally away from the camera and another person running from the left side diagonally to the right side.



(a) Human RCS over distance.



(b) Car RCS over distance.



(c) Dog RCS over distance.

Figure 5.2: Plot of how RCS values change over distance for human, car and dog.

The resulting probabilities for an object, given which sensors that had the detection is displayed in Table 5.1, see Equation 4.1. The values given in columns MOTE and Radar are representing the probabilities for an object when only that sensor triggered and without a detection of the other sensor. In addition, the probability for an object given fusion trigger is presented in the same table.

5.2.1 Localization

The results of how the different sensors and fusion correctly located objects in frames can be seen in Figure 5.4 and 5.5. The first shows the amount of correctly located objects, while the second shows the number of false detections (observe the scale on the x-axis). The same result is displayed in Figure 5.6 where the relation between true and false triggers is easier to read. For the results of additional datasets see Appendix B.1.

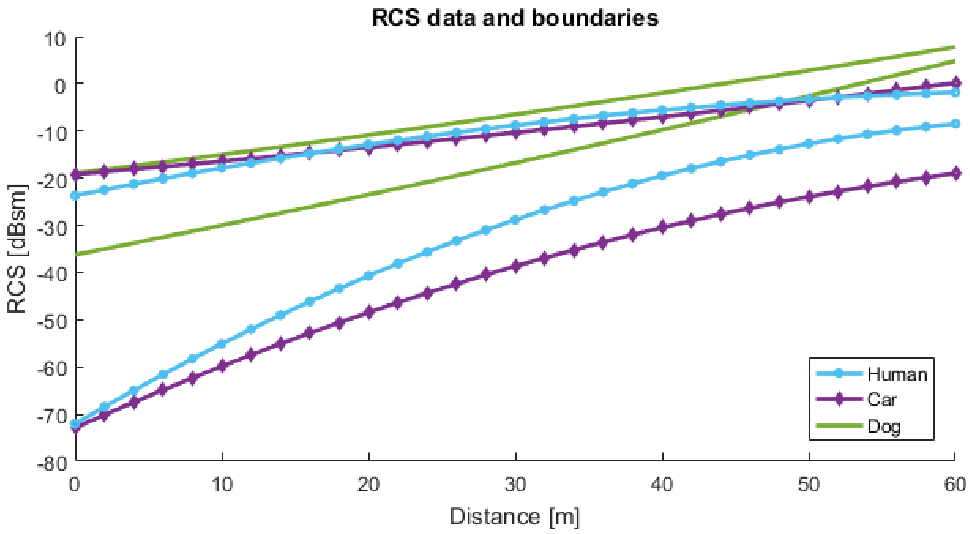


Figure 5.3: Plot of how RCS boundaries over distance for human, dog and car.

Table 5.1: The probability that it is an object given which sensors that detected.

Dataset	Both	MOTE	Radar	Fusion
Mixed	0.99925	0.089147	0.86693	0.96387
Mixed without dark	0.99924	0.1675	0.77236	0.98062
Filtered	1	0.7	0.1875	0.98851
Dark	0.86364	0.0035907	0.91724	0.9182

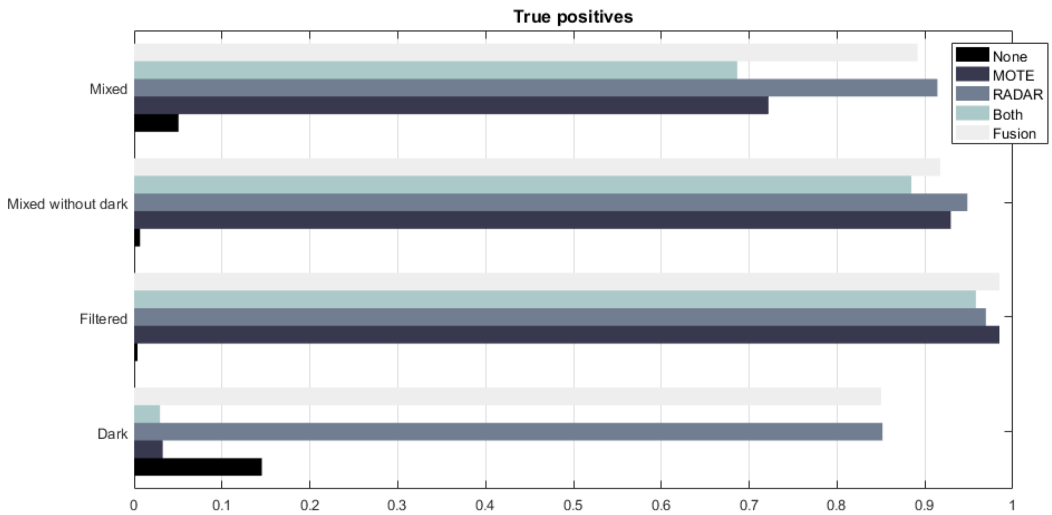


Figure 5.4: The graph shows how the true positives is distributed between the sensors for the selected datasets, as well as how the fusion performs.

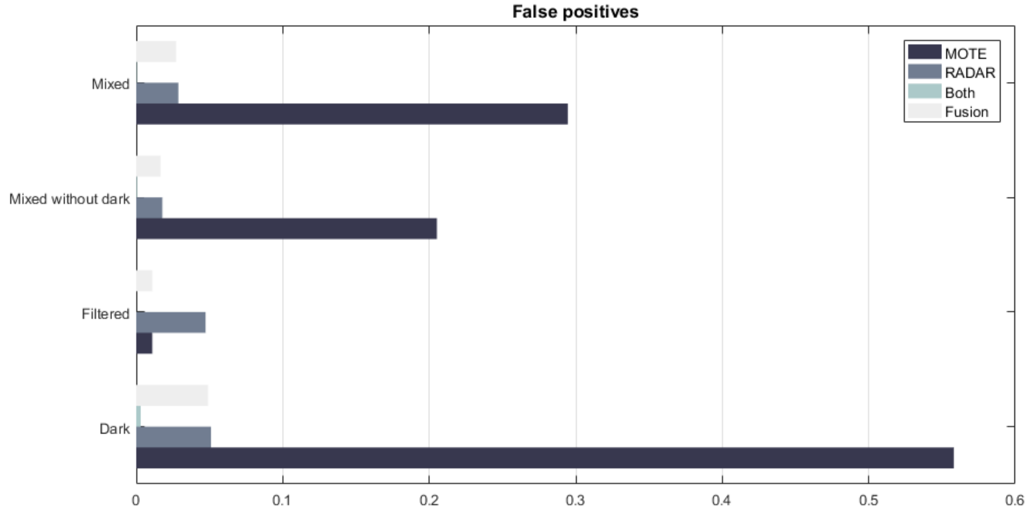


Figure 5.5: The graph show how the false positives is distributed between the sensors and fusion for the selected datasets.

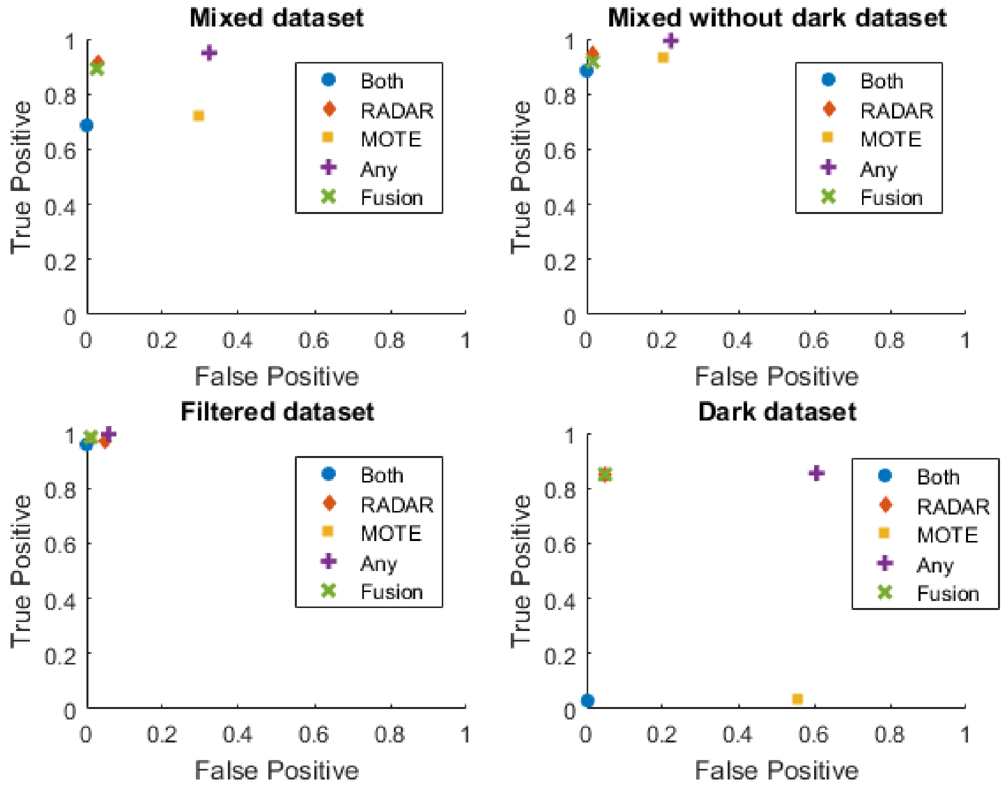


Figure 5.6: The performance of the fusion and sensors in various datasets.

5.2.2 Number of Objects

The results for the correct number of detected objects in each frame are presented in Table 5.2, where the true positives are presented as the ratio between detected true objects and total number of true objects. Whilst the false positives are presented as the number of faulty detected objects per frame. Additional results for individual films are presented in Appendix B.2.

Table 5.2: Validation of number of objects in frames.

	Fusion	**MOTE**
True Positives	0.8997	0.9222
False Positives	0.2499	0.6351

5.2.3 Correctly Triggered Frames

The results for triggering at the correct frames are presented in Table 5.3. The true positives are presented as the ratio between correct detected frames and the true number of trigger frames, whilst, false positives is presented as the ratio between false detected frames and empty frames. The results for individual films can be found in Appendix B.3.

Table 5.3: Validation of correctly triggered frames.

	Fusion	**MOTE**
True Positives	0.9926	0.9576
False Positives	0.1231	0.3818

Chapter 6

Discussion

In this chapter, the results obtained are discussed. Additionally, important comments, potential drawbacks, and skewed test data are commented. Finally, potential future work is presented.

6.1 Performance of the Fusion

When evaluating the performance it is important to understand how the system will be used. When an alarm is triggered in a frame it is sent to another system, which is also running software to filter out false alarms. If the detection is still classified as a threat it notifies an operator, that often monitors multiple scenes simultaneously. The operator needs to visually analyze if the detection is a threat or not. A problem with false triggers is that the workload of an operator can be too much to handle. Therefore it is important to minimize the number of false triggers, but also that the alarms are located correctly in the frame so that an operator quickly can see if the threat is real or not.

All in all, the performance of the fusion implemented imply that the number of false triggers is reduced. Neither MOTE or fusion miss an entire object over its lifetime in any of the films. Yet MOTE picks up the objects faster, as expected since the condition used needs MOTE to trigger in order for the fusion to do so (whilst the scene is sufficient illuminated).

6.1.1 Localization

The results for correctly localizing objects seen in Figure 5.6 indicates that the fusion and radar perform similarly. Worth mentioning is that the condition for MOTE/fusion to be correctly localized is harder. For the radar, it is enough that the detection is within 6% of the image width from the ground truth, while the others need to have a common area with the ground truth of at least 40%. The tougher requirement also results in a greater impact on any errors in the ground truth, which may arise out of the human factor.

The focus of this master's thesis was to investigate the possibility of reducing the number of false triggers in today's camera software and not to replace the camera sensor. There may be several reasons for additional visual confirmation received by the camera, e.g. additional software with object detection/face detection or similar.

When comparing the results for MOTE and fusion, seen in Figure 5.6, the loss of true positives are restricted even without dark scenes, while the decrees of false positives are distinct. Additionally, the fusion performs remarkably better under poor lighting conditions.

6.1.2 Number of Objects

Counting the number of correctly triggered objects for each frame is a good indication of how well the system can handle multiple objects in a scene. This measurement is not necessarily the most important validation in restricted sterile environments. However, when evaluating how well the algorithms work it is still important information.

These results may be partly faulty, the intention when generating the ground truth was that an object is present even if it is passing behind a solid object which covers the visibility of the moving object. This was done because the sensors have different abilities to detect objects. Since the implemented fusion needs a MOTE object to keep the fusion object, these objects are never detected by the system, but there are times when MOTE triggers e.g. on a flagpole at the same time resulting in a faulty true positive.

Even though there might be partly skewed results as mentioned, the fusion will never benefit in respect to MOTE of this, since fusion can't false trigger if MOTE does not (except for dark scenes). The number of false triggers has greatly decreased, both overall and for each individual film. The true positives are slightly lower but still keeps a high level, there are some films where there are larger drops, e.g. *Victoriastadion 8* which was caused by an extremely slow moving object and *Axis Parking Lot 2* which was affected by what was discussed in the previous paragraph.

6.1.3 Correctly Triggered Frames

When monitoring a restricted area, the information about how frequent a false trigger occurs is a good indication of how often manual work is required by an operator. It is good to note that the false triggers obtained from this method are actual false triggers where there is no human activity, whereas the other methods do not give any indication of if there is an actual object in the scene that should trigger an alarm. The results obtained using this method indicates that using sensor fusion can greatly reduce the number of false triggers while still maintaining a good amount of true positives as can be seen in Table 5.3. In this table, the fusion is triggering on even more frames than MOTE, this is due to the fact that there are dark scenes in the data set.

6.2 Classification with Radar

The possibility to classify objects with the additional information was lightly investigated using the RCS values. The RCS boundaries for human, car and dog was quite overlapping were only the highest values for a car could be separated from humans, as well as the lower values for the dog. The variety of data for dogs and cars were strongly limited to only consisting of one dog and one car. In addition, the dog ran with high speed which resulted in less radar detection, especially on longer distances. This could imply that the boundaries for these object types should be wider. Even though the used films for analyzing human RCS values were larger, with up to seven different persons, the boundaries did not get significantly wider compared to using only one human to calculate RCS. There may still be a wider variety in the RCS value, all the people that were tested on was between 20-30 years of age with normal height and physique.

6.3 Future Work

Even though the results proved to be good, there could still be improvements to the system to make it even better. In this section, the authors will try to list and discuss all ideas that they came up with during this thesis but due to some limitation, e.g. lack of time, could not test.

When an object only was detected by radar but still classified as a human activity, the output is only a vertical line on the x position of the object. This could be improved by using the available foreground and motion masks to see if the radar detection could be mapped to an object at either of the two masks to produce an output of a polygon.

The masks described could also be used in a classification tree. One of the masks could detect an object even if MOTE did not detect it. In that case, if radar detected an object and there is some motion detected in the same area the system could detect the object. This could decrease missed alarms, however, it could also increase the number of false alarms.

The algorithms for radar clustering and tracking objects could be improved and generate better results. This could make the system more stable on more difficult scenes and not be limited to sterile environments.

Throughout the thesis, there were some experiments with RCS, unfortunately with no greater success. However, the authors believe that it could be possible to combine RCS with information about the radar hits for the object, e.g. how many hits, to reduce even more false triggers from trees. There need to be more data collected from scenes where there are only trees that are detected. Machine learning is also worth investigating to see if it can make the system learn how the response from trees looks like.

Chapter 7

Conclusion

The aim of this master's thesis was to explore the possibility of reducing the number of false triggers in a motion detection system without changing the current system while aiming for remaining the number of true positives.

It was a clear indication that the information fusion was able to keep the number of true object detections around the same level, at the same time as reducing false triggers drastically. The methods used are good groundwork, but there are many parameters, and different combinations thereof, that needs to be taken into consideration if worked upon.

The results gave an indication that it could be enough to only use a radar sensor and receive results as good as the ones for fusion or potentially even better. For these kinds of scenes, it could be sufficient to use only radar if a visual validation is not required. Some problems with this could be that it could be difficult to distinguish between a deer and a burglar. Looking at the history of radar data after a burglary can't identify the burglar so the legal actions that can be taken are limited. However, scenes that weren't specified for this thesis have not been sufficiently tested upon to draw a conclusion.

There are external factors that affect the reliability of the sensors. It is mostly the visual factors that affect MOTE, such as night time but also if objects are moving towards the camera; this results in both true positives and false positives. Radar has more problems with lateral movements or scenes with a lot of metallic objects which causes reflections; this results in false triggers. If object movement and scenes are correctly identified then sensor fusion could be greatly improved if basing decisions on the probability that each sensor is correct.

Bibliography

[1] Axis Communication AB, *F41 Main Unit*, 2015.

[2] Axis Communication AB, *AXIS F1005-E Sensor Unit*, 2015.

[3] M. C. B. Jr. and S. R. German, *Basic Radar Analysis (Artech House Radar)*, ch. 1. Artech House Publishers, 9 2015.

[4] C. Stiller, F. P. León, and M. Kruse, "Information fusion for automotive applications – an overview," *Information Fusion*, vol. 12, no. 4, pp. 244 – 252, 2011. Special Issue on Information Fusion for Cognitive Automobiles.

[5] M. Skolnik, *Radar Handbook, Third Edition*, ch. 1. McGraw-Hill Education, 3 ed., 2 2008.

[6] C. Wolff, "Pulse radar." www.radartutorial.eu Radar Basics, `http://www.radartutorial.eu/02.basics/Pulse%20Radar.en.html`. Retrieved: 2017-02-12.

[7] C. Wolff, "Frequency-modulated continuous-wave radar (fmcw radar)." www.radartutorial.eu Radar Basics, `http://www.radartutorial.eu/02.basics/Frequency%20Modulated%20Continuous%20Wave%20Radar.en.html`. Retrieved: 2017-01-16.

[8] D. S. Sawicki, "Doppler principles (police traffic radar handbook)." CopRadar.com – subsidiary of Sawicki Enterprises, `http://www.copradar.com/preview/chapt2/ch2d1.html`, 1999–2000. Retrieved: 2017-01-16.

[9] V. Winkler, "Range doppler detection for automotive fmcw radars," in *2007 European Microwave Conference*, pp. 1445–1448, Oct 2007.

[10] C. Wolff, "Radar cross-section." www.radartutorial.eu Radar Basics, `http://www.radartutorial.eu/01.basics/Radar%20Cross%20Section.en.html`. Retrieved: 2017-01-26.

[11] D. R. M. O'Donnell, "Introduction to radar systems." http://www.ll.mit.edu WORKSHOPS/EDUCATION, http://www.ll.mit.edu/workshops/education/videocourses/introradar/. Retrieved: 2017-02-13.

[12] M. Skolnik, *Radar Handbook, Third Edition*, ch. 14. McGraw-Hill Education, 3 ed., 2 2008.

[13] F. B. Thomas Wimmer, Martin Fritz, "Articulated pedestrian target specifications." European Automobile Manufacturers Association https://www.acea.be/uploads/publications/Articulated_Pedestrian_Target_Specifications_-_Version_1.0.pdf. Retrieved: 2017-02-13.

[14] C. Wolff, "Free-space path loss (fspl)." www.radartutorial.eu Radar Basics, https://www.cis.upenn.edu/~mkearns/papers/em-kmeans.pdf. Retrieved: 2017-02-13.

[15] A. Kirillov, "Motion detection algorithms." http://www.codeproject.com/Articles/10248/Motion-Detection-Algorithms. Accessed: 2016-11-20.

[16] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, ch. 8. Pearson, 1 ed., 5 2005.

[17] A. Y. N. Michael Kearns, Yishay Mansour, "An information-theoretic analysis of hard and soft assignment methods for clustering." https://www.cis.upenn.edu/, http://www.radartutorial.eu/01.basics/Free-Space%20Path%20Loss.en.html. Retrieved: 2017-02-13.

[18] D. S. Sayad, "An introduction to data mining - clustering." SaedSayad.com http://www.saedsayad.com/clustering.htm. Accessed: 2017-02-14.

[19] J. P. Ortega, M. Del, R. B. Rojas, and M. J. Somodevilla, "Research issues on k-means algorithm: An experimental trial using matlab," in *CEUR Workshop Proceedings: Semantic Web and New Technologies*.

[20] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, ch. 11. Pearson Education, 2 ed., 2 2012.

[21] F. Castanedo, "A review of data fusion techniques," *The Scientific World Journal*, vol. 2013, 2013.

[22] M. Guo, D. Li, G. Chen, Y. Xu, W. He, T. Zhang, L. An, and M. Lv, "Multi-sensor information fusion for unmanned cars using radar map," in *2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems*, vol. 03, pp. 1166–1170, Oct 2012.

[23] A. Damodaran, "Statistical distributions." people.stern.nyu.edu Statistical Distributions, http://people.stern.nyu.edu/adamodar/New_Home_Page/StatFile/statdistns.htm. Retrieved: 2017-02-21.

[24] "Bayes' theorem." Investopedia, http://www.investopedia.com/terms/b/bayes-theorem.asp. Retrieved: 2017-02-22.

[25] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, ch. 1. Pearson, 1 ed., 5 2005.

[26] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, ch. 4. Pearson, 1 ed., 5 2005.

[27] A. Safety, "Why automakers choose radar and camera sensor fusion." http://safety.trw.com/why-automakers-choose-radar-and-camera-sensor-fusion/. Accessed: 2016-11-08.

[28] A. Polychronopoulos, M. Tsogas, A. J. Amditis, and L. Andreone, "Sensor fusion for predicting vehicles' path for collision avoidance systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, pp. 549–562, Sept 2007.

[29] P. Lytrivis, G. Thomaidis, M. Tsogas, and A. Amditis, "An advanced cooperative path prediction algorithm for safety applications in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, pp. 669–679, Sept 2011.

[30] L. Snidaro, G. L. Foresti, R. Niu, and P. K. Varshney, "Sensor fusion for video surveillance," 2004.

[31] i LIDS team, "Imagery library for intelligent detection systems." https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/143870/ilids-brochure.pdf. Retrieved: 2016-09-29.

[32] J. M. Spool, "The kj-technique: A group process for establishing prioritie." UIE, User Interface Engineering, https://articles.uie.com/kj_technique/, 11 2004.

Appendices

Appendix A

Datasets

Presentation of the complete list of film scenarios and description about datasets that has ground truth used for testing.

A.1 Film scenarios

Short summaries of the filmed scenarios, sorted by the location.

A.1.1 Park (Stadsparken)

Stadsparken 1 Lots of people, one camouflaged in the bushes.

Stadsparken 2 Several single persons, bikers.

Stadsparken 3 Single persons, stopping for a long time, wind beams in trees.

Stadsparken 4 A lot of noise; birds, bushes, trees. There are also single persons and groups.

Stadsparken 5 Big trees directly above camera, single persons, and groups.

Stadsparken 6 Big trees directly above camera, single persons, and groups.

Stadsparken 7 Fountain with several single persons in complex paths.

A.1.2 Open field, football stadium (Victoriastadion)

Victoriastadion 1 Sunny with dog and person.

Victoriastadion 2 Sunny with a single person walking straight from the camera, rabbit runs by.

Victoriastadion 3 Sunny, a person with a metal object and a single person walking across FOV on 10 m distance.

Victoriastadion 4 Sunny, a person with a metal object and a single person walking across FOV on 25 m distance.

Victoriastadion 5 Sunny, a person with a metal object and a single person walking across FOV on 45 m distance.

Victoriastadion 6 Sunny with two people crouching across.

Victoriastadion 7 Sunny with two people running across.

Victoriastadion 8 Sunny with two people walking real slow across.

Victoriastadion 9 Sunny with a person walking and running diagonally.

Victoriastadion 10 Sunny with the single person running diagonally towards the camera.

Victoriastadion 11 Person walking holding the metal object.

Victoriastadion 12 Single person walking straight away from the camera.

Victoriastadion 13 Single person walking straight away from the camera.

Victoriastadion 14 Single person walking straight away from the camera.

Victoriastadion 15 Single person walking straight away from the camera.

Victoriastadion 16 Single person walking straight away from the camera.

Victoriastadion 17 Single person walking straight away from the camera.

Victoriastadion 18 Single person walking straight away from the camera.

Victoriastadion 19 Groups of people walking, sensors shaking.

Victoriastadion 20 Groups walking randomly, high loaded scene, dog at the end.

Victoriastadion 21 Dog running around.

Victoriastadion 22 Group walking away from camera and back.

Victoriastadion 23 Snow on the ground, a single person walking away and back, ice blocking radar.

Victoriastadion 24 Snow on the ground, a single person walking away and back tossing snow.

Victoriastadion 25 Ice blocking camera, a person running.

Victoriastadion 26 Snow on the ground, two persons throwing snowballs.

Victoriastadion 27 Snow on the ground single person running around.

Victoriastadion 28 Snow, three people walking randomly.

Victoriastadion 29 Snow, three people throwing snowballs, and push one around.

Victoriastadion 30 Snow, three people building an ugly snowman.

A.1.3 Empty parking lot (Near Axis)

Parking Lot 1 Empty scene one parked car.

Parking Lot 2 Car driving away, a single person walking with a metal object.

Parking Lot 3 Car drive straight.

Parking Lot 4 Car drive zig zag.

Parking Lot 5 Bike drive straight.

Parking Lot 6 Bike drive zig zag.

Parking Lot 7 Snow, a single person walking close in front of the camera.

Parking Lot 8 Snow, a car driving zig zag.

Parking Lot 9 Snow, car driving fast.

A.1.4 Busy parking lot (Axis)

Axis Parking Lot 1 Lots of cars parked, cars drive by, single people walking, a metal object thrown, a car comes in and park, swaying flags.

Axis Parking Lot 2 Single people walking around between lots of parked cars, swaying flags, car drive by.

Axis Parking Lot 3 Lots of cars parked, car drives by, single persons walking around, swaying flags.

Axis Parking Lot 4 Lots of cars parked, car drive by, two person does jumping jacks in front of the camera, swaying flags.

A.1.5 Indoor

Indoor 1 Single person walking around and down in stairs.

Indoor 2 Single person walking up from stairs.

Indoor 3 At an office, a single person walking back and forth, a single person with a metal object.

Indoor 4 At office, single person walking back and forth.

A.1.6 Dark open field (Värpinge)

Värpinge 1 Dark night scene, a single person walking.

Värpinge 2 Empty dark night scene with a light beam.

Värpinge 3 Dark night scene, two persons walking on different distances.

Värpinge 4 Dark night scene, a single person walking around, light beam sweep in.

Värpinge 5 Dark night scene, a single person walking with a flashlight.

Värpinge 6 Dark night scene, a single person walking and begins to dazzle the camera.

Värpinge 7 Dark night scene, one person standing in still and dazzle the camera while other person walking.

Värpinge 8 Dark night scene, four persons walking around.

Värpinge 9 Dark night scene, light beams from car sweep by.

Värpinge 10 Dark night scene, light beams from car sweep by.

A.1.7 In front of a tree (Kemicentrum)

Kemicentrum 1 Directly in front of a tree, with small wind beams.

Kemicentrum 2 Directly in front of a tree, with small wind beams.

Kemicentrum 3 Directly in front of a tree, with small wind beams, with big branch flapping in front of the camera.

Kemicentrum 4 Directly in front of a tree, with small wind beams and a branch in the tree that flaps.

A.1.8 Open field (Lophtet)

Lophtet 1 Two plastic bags moving around.

Lophtet 2 One person walking in from the side and drops a plastic bag and run away. Two plastic bags moving around.

Lophtet 3 Plastic bags moving around.

Lophtet 4 Cardboard dragged towards the camera.

Lophtet 5 Sundown, Single person walking from left to right at approximately 5 meters from the camera.

Lophtet 6 Sundown, single person walking from right to left at approximately 5 meters from the camera.

Lophtet 7 Single person walking from left to right at approximately 10 meters from the camera.

Lophtet 8 Single person walking from right to left at approximately 10 meters from the camera.

Lophtet 9 Single person running from left to right at approximately 10 meters from the camera.

Lophtet 10 Single person running from right to left at approximately 10 meters from the camera.

Lophtet 11 Single person walking from left to right at approximately 30 meters from the camera.

Lophtet 12 Single person walking from right to left at approximately 30 meters from the camera.

Lophtet 13 Single person walking directly towards the camera.

A.1.9 Other

Other Near office, between buildings with parked cars, several people, one of which sneaks around the cars.

A.2 Divided subsets of data

The complete dataset of film scenarios in Appendix A.1 have been divided into different smaller subsets to test different characteristics.

A.2.1 Mixed dataset

This dataset have a broad mix of scene types, the individual films used are;

Victoriastadion 2, 4, 5, 6, 7, 9, 10 and 12

Lophtet 1, 3, 4, 5, 8, 10, 11 and 13

Parking Lot 5

Värpinge 2, 3, 4 and 10

A.2.2 Mixed without dark dataset

This dataset have a broad mix of scene types and contains the same films as the Mixed dataset, except that it does not contain any of the dark ones, the individual films used are;

Victoriastadion 2, 4, 5, 6, 7, 9, 10 and 12

Lophtet 1, 3, 4, 5, 8, 10, 11 and 13

Parking Lot 5

A.2.3 Filtered dataset

This dataset contains just a few film scenarios which should not be of any problem for either of the sensors, the individual films used are;

Lophtet 5, 6, 7, 8, 9 and 10

A.2.4 Dark dataset

This dataset contains only dark scenes, some with light beams, flash lights and some completely dark, the individual films used are;

Värpinge 3, 4, 5, 6, 9 and 10

A.2.5 Dark with lights dataset

This dataset contains only dark scenes with lights e.g. light beams and flash lights, the individual films used are;

Värpinge 2, 4, 5, 6, 7, 9 and 10

A.2.6 Dark without light dataset

This dataset contains only completely dark scenes, the individual films used are;

Värpinge 3 and 8

A.2.7 Non-human dataset

This dataset only contains objects that should not trigger an alarm, e.g. plastic bag and cardboard, the individual films used are;

Lophtet 1, 3 and 4

A.2.8 Azimuth dataset

This dataset only contains objects that moves along the horizontal FOV, the individual films used are;

Lophtet 5, 6, 7, 8, 9, 10 and 11

Victoriastadion 3, 4 and 5

Appendix B

Results

Presentation of additional validation results, for the datasets *Dark with lights*, *Dark without light*, *Non-human* and *Azimuth dataset*, when validating of localization. As well as, the validation of number of objects in frames and correctly trigged frames.

B.1 Localization

Results for true positives for each validated dataset is presented in Table B.1, where the numbers are calculated by Equation 4.1. The false positive rates for the same datasets is presented in Table B.2, and the numbers are calculated in a similar way.

Table B.1: True positives rate

Dataset	**True Positives**				
	Both	Radar	MOTE	Any	Fusion
Mixed	0.6864	0.9141	0.7219	0.9496	0.8915
Mixed without dark	0.8845	0.9483	0.9295	0.9933	0.9174
Filtered	0.9580	0.9695	0.9847	0.9962	0.9847
Dark	0.0294	0.8516	0.0325	0.8547	0.8501
Dark with lights	0.0503	0.8783	0.0556	0.8836	0.8624
Dark without light	0	0.8521	0	0.8521	0.8535
Non-human	-	-	-	-	-
Azimuth	0.8603	0.9051	0.9249	0.9698	0.9208

Table B.2: False positives rate

Dataset	False Positives				
	Both	Radar	MOTE	Any	Fusion
Mixed	0.0004	0.0288	0.2945	0.3229	0.0271
Mixed without dark	0.0006	0.0178	0.2052	0.2224	0.0166
Filtered	0	0.0473	0.0109	0.0582	0.0109
Dark	0.0030	0.0510	0.5580	0.6060	0.0490
Dark with lights	0.0029	0.0485	0.5808	0.6264	0.0466
Dark without light	0	0.0738	0.0123	0.0862	0.0277
Non-human	0	0.0046	0.6598	0.6644	0.0183
Azimuth	0	0.0200	0.0173	0.0372	0.0145

B.2 Number of Objects

Results for each validated scenario is presented in Table B.3. For each frame, the number of objects that should trigger is defined and if it is equal to the number of detected objects then the frame gives 100% true positives. However, if the detected number is larger it will still give 100% true positives but the additional detections will be counted as false positives. Furthermore, if the detected number is lower the percentage for the frame will decrease. For this validation method, the percentage of true positives is calculated by the number of true detected divided by the number of real objects.

The column Objects in Table B.3 under True Positives indicates the maximum number of true positives, e.g. a film with 3 objects that is present under 10 frames each will be displayed as 30.

False positives are represented as the number of false detections in total for each system, an additional column is presented in Table B.3 indicating total validated frames for the scenario.

Table B.3: Results of validation on number of objects in each frame for individual scenarios.

Scenarios	True Positives			False Positives		
	MOTE	Fusion	Objects	MOTE	Fusion	Frames
Stadsparken 1	100%	99.23%	391	1024	685	405
Stadsparken 2	99.73%	99.20%	375	660	264	440
Stadsparken 5	99.68%	99.68%	308	4766	2103	405
Stadsparken 6	-	-	-	376	119	66
Victoriastadion 1	-	-	-	1	0	61
Victoriastadion 2	95.71%	95.71%	280	6	4	489
Victoriastadion 3	92.31%	89.74%	117	9	2	99
Victoriastadion 4	100%	100%	196	0	0	196
Victoriastadion 5	100%	100%	141	1	0	141

Continued on next page

Table B.3 – *Continued from previous page*

Scenarios	True Positives			False Positives		
	MOTE	Fusion	Objects	MOTE	Fusion	Frames
Victoriastadion 6	98.33%	92.50%	240	0	0	120
Victoriastadion 7	97.83%	95.65%	138	0	0	82
Victoriastadion 8	100%	78.30%	507	20	0	259
Victoriastadion 9	96.28%	91.63%	215	3	0	158
Victoriastadion 10	100%	100%	119	51	4	136
Victoriastadion 11	97.50%	97.50%	80	37	10	154
Victoriastadion 12	100%	99.34%	152	48	10	237
Victoriastadion 13	96.18%	96.18%	157	34	0	225
Victoriastadion 14	100%	99.43%	174	123	0	208
Victoriastadion 15	100%	100%	107	10	9	107
Victoriastadion 16	94.92%	94.92%	118	0	0	118
Victoriastadion 17	99.31%	99.31%	145	20	0	194
Victoriastadion 18	100%	100%	106	57	0	106
Victoriastadion 23	100%	100%	139	89	0	139
Victoriastadion 24	99.75%	99.75%	394	6	6	466
Victoriastadion 26	100%	96.41%	1756	0	0	981
Victoriastadion 27	99.71%	99.71%	341	0	0	454
Victoriastadion 28	85.98%	77.89%	1248	9	2	626
Victoriastadion 29	89.64%	79.57%	2394	16	0	798
Victoriastadion 30	88.20%	84.06%	4128	16	3	1376
Parking Lot 1	-	-	-	0	0	44
Parking Lot 2	98.95%	97.90%	95	39	20	77
Parking Lot 3	100%	99.03%	103	11	0	161
Parking Lot 4	99.53%	99.05%	211	40	0	230
Parking Lot 5	99.04%	99.04%	104	0	0	187
Parking Lot 6	99.62%	99.62%	261	15	0	301
Parking Lot 8	100%	98.85%	174	0	0	600
Parking Lot 9	99.20%	99.20%	125	0	0	245
Axis Parking Lot 1	98.36%	96.72%	427	282	132	328
Axis Parking Lot 2	99.43%	90.17%	529	449	176	370
Axis Parking Lot 3	97.45%	94.90%	392	13	3	244
Axis Parking Lot 4	100%	99.11%	112	61	14	107
Värpinge 1	-	-	-	0	0	48
Värpinge 2	-	-	-	57	0	198
Värpinge 3	1.63%	88.16%	245	0	6	169
Värpinge 4	36.49%	98.65%	148	56	2	183
Värpinge 5	96.00%	100%	100	45	2	175
Värpinge 6	15.00%	97.00%	100	4	1	157
Värpinge 9	-	-	-	129	0	139
Värpinge 10	-	-	-	224	0	155
Kemicentrum 1	-	-	-	22	16	123

Continued on next page

Table B.3 – *Continued from previous page*

Scenarios	True Positives			False Positives		
	MOTE	Fusion	Objects	MOTE	Fusion	Frames
Kemicentrum 2	-	-	-	277	100	106
Kemicentrum 3	-	-	-	147	77	58
Kemicentrum 4	-	-	-	59	27	88
Lophtet 1	-	-	-	86	0	143
Lophtet 2	100%	99.12%	113	59	5	235
Lophtet 3	-	-	-	164	0	247
Lophtet 4	-	-	-	39	8	133
Lophtet 5	100%	100%	37	0	0	60
Lophtet 6	100%	100%	49	11	1	55
Lophtet 7	100%	100%	61	0	0	91
Lophtet 8	94.87%	94.87%	78	0	0	105
Lophtet 9	96.67%	96.67%	30	0	0	52
Lophtet 10	85.29%	85.29%	34	0	0	56
Lophtet 11	95.94%	95.94%	123	0	0	140
Lophtet 12	100%	99.20%	125	0	0	125
Lophtet 13	100%	100%	31	0	0	31
Other	100%	92.93%	99	7	0	125

B.3 Correctly Trigged Frames

The results for each of the validated scenarios can be seen in Table B.4. True positives mean that the frame contains at least one real object that should trigger the alarm and that the sensor has found at least one object. There is no guarantee that the found object is the one that should trigger the alarm, e.g. there can be a person on the right side of the image that should trigger the alarm while the sensor finds a flagpole on the left part of the image, this would count as a true positive. False positives are represented as the ratio between the number of trigger frames that should not trigger and the number of frames that should not trigger, resulting in a percentage.

Both true and false positives are calculated for MOTE and fusion. In addition to the percentage of the different systems, the number of frames, which is representing 100%, is presented in Table B.4.

As mentioned in Section 4.8, all frames for each scenario is not validated. For instance, if an object starts in the image the start is cut, because of MOTE include the object as a part of the background model and detection may be delayed.

Table B.4: Results of validation on correct triggered frame for individual scenarios.

Scenarios	True Positives			False Positives		
	MOTE	Fusion	Frames	MOTE	Fusion	Frames
Stadsparken 1	100%	99.23%	391	21.43%	0%	14
Stadsparken 2	100%	100%	363	42.47%	10.96%	73
Stadsparken 5	100%	100%	474	100%	88.09%	235
Stadsparken 6	-	-	-	100%	54.55%	66
Stadsparken 7	100%	100%	543	-	-	-
Victoriastadion 1	100%	100%	125	1.64%	0%	61
Victoriastadion 2	99.63%	99.63%	267	0%	0%	233
Victoriastadion 3	100%	100%	99	-	-	-
Victoriastadion 3	100%	100%	196	-	-	-
Victoriastadion 5	100%	100%	141	-	-	-
Victoriastadion 6	100%	100%	128	-	-	-
Victoriastadion 7	100%	100%	82	-	-	-
Victoriastadion 8	100%	100%	259	-	-	-
Victoriastadion 9	95.81%	95.81%	191	0%	0%	8
Victoriastadion 10	100%	100%	119	41.18%	11.77%	17
Victoriastadion 11	97.50%	97.50%	80	50.00%	13.51%	74
Victoriastadion 12	100%	99.34%	152	56.47%	11.77%	85
Victoriastadion 13	96.18%	96.18%	157	48.53%	0%	68
Victoriastadion 14	100%	99.43%	174	100%	0%	34
Victoriastadion 15	100%	100%	107	-	-	-
Victoriastadion 16	94.92%	94.92%	118	-	-	-
Victoriastadion 17	99.31%	99.31%	145	34.69%	0%	49
Victoriastadion 18	100%	100%	106	-	-	-
Victoriastadion 19	100%	100%	237	74.42%	26.74%	86
Victoriastadion 20	100%	100%	379	-	-	-
Victoriastadion 21	100%	100%	27	-	-	-
Victoriastadion 22	97.57%	97.57%	371	25.68%	0%	74
Victoriastadion 23	100%	100%	139	-	-	-
Victoriastadion 24	99.75%	99.75%	394	0%	0%	72
Victoriastadion 26	100%	99.89%	883	0%	0%	98
Victoriastadion 27	99.71%	99.71%	341	0%	0%	113
Victoriastadion 28	100%	100%	429	0.51%	0.51%	197
Victoriastadion 29	99.89%	99.67%	896	0%	0%	55
Victoriastadion 30	99.72%	99.72%	1425	0%	0%	197
Parking Lot 1	-	-	-	0%	0%	44
Parking Lot 2	100%	100%	109	0%	0%	26
Parking Lot 3	100%	99.03%	103	15.52%	0%	58
Parking Lot 4	99.53%	99.05%	211	0%	0%	19
Parking Lot 5	99.27%	99.27%	137	0%	0%	83
Parking Lot 6	99.63%	99.25%	268	0%	0%	40

Continued on next page

Table B.4 – *Continued from previous page*

Scenarios	True Positives			False Positives		
	MOTE	Fusion	Frames	MOTE	Fusion	Frames
Parking Lot 7	100%	100%	73	54.47%	0.81%	123
Parking Lot 8	100%	98.85%	174	0%	0%	426
Parking Lot 9	99.40%	98.80%	166	0%	0%	120
Axis Parking Lot 1	100%	100%	257	49.30%	16.90%	71
Axis Parking Lot 2	100%	100%	265	81.91%	27.62%	105
Axis Parking Lot 3	100%	100%	221	1.15%	1.15%	87
Axis Parking Lot 4	100%	100%	105	19.15%	0%	47
Indoor 1	90.08%	89.31%	131	0%	0%	8
Indoor 2	98.44%	98.44%	64	0%	0%	17
Värpinge 1	-	-	-	0%	0%	48
Värpinge 2	-	-	-	27.27%	0%	198
Värpinge 3	2.61%	98.69%	153	0%	9.62%	52
Värpinge 4	36.49%	98.65%	148	0%	2.86%	35
Värpinge 5	96.00%	100%	100	26.67%	2.67%	75
Värpinge 6	15.00%	97.00%	100	1.75%	1.75%	57
Värpinge 7	0%	93.10%	29	0%	28.57%	28
Värpinge 8	0%	97.42%	155	-	-	-
Värpinge 9	-	-	-	84.17%	0%	139
Värpinge 10	-	-	-	66.45%	0%	155
Kemicentrum 1	-	-	-	11.38%	10.57%	123
Kemicentrum 2	-	-	-	69.81%	52.83%	106
Kemicentrum 3	-	-	-	68.97%	58.62%	58
Kemicentrum 4	-	-	-	47.73%	27.27%	88
Lophtet 1	-	-	-	36.36%	0%	143
Lophtet 2	100%	99.12%	113	18.03%	4.10%	122
Lophtet 3	-	-	-	66.40%	0%	247
Lophtet 4	-	-	-	28.57%	6.02%	133
Lophtet 5	100%	100%	37	0%	0%	23
Lophtet 6	100%	100%	49	0%	0%	6
Lophtet 7	100%	100%	61	0%	0%	30
Lophtet 8	94.87%	94.87%	78	0%	0%	27
Lophtet 9	96.67%	96.67%	30	0%	0%	22
Lophtet 10	85.29%	85.29%	34	0%	0%	22
Lophtet 11	95.94%	95.94%	123	0%	0%	17
Lophtet 12	100%	99.20%	125	-	-	-
Lophtet 13	100%	100%	31	-	-	-
Other	100%	100%	289	36.11%	12.50%	72