Winter 2011

# Sub pixel analysis and processing of sensor data for mobile target intelligence information and verification

Theresa Allen Williams
*Louisiana Tech University*

# SUB PIXEL ANALYSIS AND PROCESSING OF SENSOR DATA FOR MOBILE TARGET INTELLIGENCE INFORMATION AND VERIFICATION

by

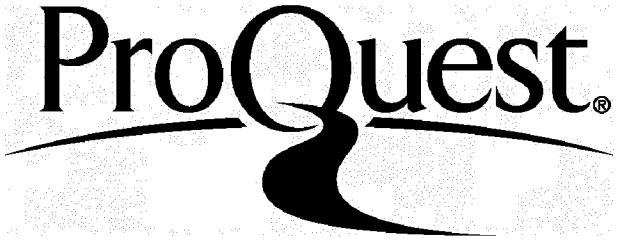
Theresa Allen Williams, M.E.

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Engineering

COLLEGE OF ENGINEERING AND SCIENCE

LOUISIANA TECH UNIVERSITY

February 2011



LOUISIANA TECH UNIVERSITY

**THE GRADUATE SCHOOL**

20 August 2010

Date

We hereby recommend that the dissertation prepared under our supervision by Theresa Allen Williams

entitled SUB PIXEL ANALYSIS AND PROCESSING OF SENSOR DATA FOR MOBILE TARGET INTELLIGENCE INFORMATION AND VERIFICATION

be accepted in partial fulfillment of the requirements for the Degree of Doctor of Engineering

Supervisor of Dissertation Research

Head of Department

Department

Recommendation concurred in:

Advisory Committee

**Approved:**

Director of Graduate Studies

**Approved:**

Dean of the Graduate School

Dean of the College

GS Form 13a
(6/07)

# ABSTRACT

This dissertation introduces a novel process to study and analyze sensor data in order to obtain information pertaining to mobile targets at the sub-pixel level. The process design is modular in nature and utilizes a set of algorithmic tools for change detection, target extraction and analysis, super-pixel processing and target refinement. The scope of this investigation is confined to a staring sensor that records data of sub-pixel vehicles traveling horizontally across the ground. Statistical models of the targets and background are developed with noise and jitter effects. Threshold Change Detection, Duration Change Detection and Fast Adaptive Power Iteration (FAPI) Detection techniques are the three methods used for target detection. The PolyFit and FermiFit are two tools developed and employed for target analysis, which allows for flexible processing. Tunable parameters in the detection methods, along with filters for false alarms, show the adaptability of the procedures. Super-pixel processing tools are designed, and Refinement Through Tracking (RTT) techniques are investigated as post-processing refinement options. The process is tested on simulated datasets, and validated with sensor datasets obtained from RP Flight Systems, Inc.

# APPROVAL FOR SCHOLARLY DISSEMINATION

The author grants to the Prescott Memorial Library of Louisiana Tech University the right to reproduce, by appropriate methods, upon request, any or all portions of this Dissertation. It is understood that "proper request" consists of the agreement, on the part of the requesting party, that said reproduction is for his personal use and that subsequent reproduction will not occur without written approval of the author of this Dissertation. Further, any portions of the Dissertation used in books, papers, and other works must be appropriately referenced to this Dissertation.

Finally, the author of this Dissertation reserves the right to publish freely, in the literature, at any time, any or all portions of this Dissertation.

Author Theresa A Williams

Date 10/21/2010

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# CHAPTER ONE

# INTRODUCTION

With the nation gearing toward a persistent surveillance environment [1], the ability to convert an abundance of raw sensor data into quality information is imperative. One aspiration in this arena is to obtain intelligent information from the data with efficiency and accuracy in order to provide the means to make timely strategic, operational, and tactical decisions concerning targets of interest. This dissertation presents a sub-pixel processing technique that will detect and analyze sensor data and provide moving target information.

Although computational methods currently exist for target detection and analysis, many of these processing and analysis techniques rely on multiple pixels or whole images. Problems with these methods arise when the target is smaller than a pixel. This issue is solved by temporally focusing on single pixels. By capturing the intensity variations of a pixel over time and applying knowledge of a pixel's geometry, it is feasible to detect a mobile target and acquire its characteristics.

The sub-pixel process assumes a staring sensor observing individual targets smaller than a pixel. Using standard detection algorithms, a target is detected and extracted from a frame sequence. Then, the FermiFit and PolyFit methods are utilized to analyze the target's data; yielding the moving target's characteristics of velocity and

length. The process is tested on simulated datasets, using the statistical models, and validated with sensor datasets obtained from RP Flight Systems, Inc.

## 1.1. Research Need

### 1.1.1. Meeting the Needs of High Altitude Sensors

The distance between the sensor and the target plays an important role in determining the level of image detail and the amount of area coverage the sensor can provide. Sensors on high altitude platforms, such as satellites, high altitude aircraft or Remotely Piloted Aircraft (RPAs), typically view large areas, but are limited in detail [2].

The level of image detail depends on the spatial resolution found in the instantaneous field of view. This field of view and the altitude of the sensor determine the size of the resolution cell. In order to detect a homogeneous feature in an image, the object has to be equal to or larger than the resolution cell [2]. Smaller targets are not easily discerned in a single image as the sensor detects the intensity combination of all of the features in the cell (Figure 1.1).

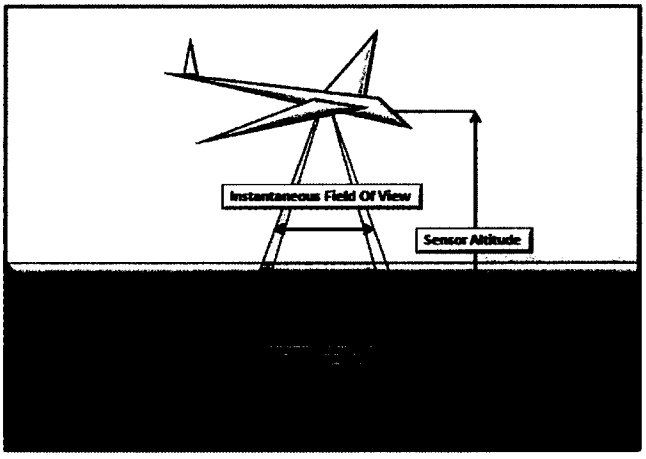



Figure 1.1 Graphical depiction of the relationship between sensor altitude, instantaneous field of view and the resolution cell

Novel methods of detecting these smaller targets to provide target characteristics are useful in military, law enforcement and homeland security applications.

### 1.1.2. Applications

There are numerous applications for which sub-pixel processing is applicable. Some notable applications are for military Intelligence, Surveillance and Reconnaissance (ISR), border patrol and drug interdiction.

A longer duration staring capability is needed for persistence surveillance, but given ephemeris data, persistent surveillance with high resolution may not be possible. Low Earth Orbiting (LEO) satellites have high resolution but a short dwell-time over the target area, which is dictated by ephemeris data. Obtaining a balance between dwell-time and resolution often means missing target opportunities. Geostationary Earth Orbiting (GEO) satellites, however, provide persistent surveillance, but given the distance from Earth, it has limited resolution. With sub-pixel processing, the low resolutions experienced by geostationary satellites are mitigated, providing new opportunities for the Intelligence, Surveillance, and Reconnaissance (ISR) community.

The U.S. border patrol has a vast amount of area to safeguard, with nearly 6,000 miles of Mexican and Canadian international land borders and over 2,000 miles of coastal waters in its jurisdiction. The Department of Homeland Security does not have the resources or assets to continuously monitor every mile [3]. The sub-pixel process can be useful in detecting the illegal entrance of people or vehicles into the country, especially in remote locations. The algorithm can utilize sensor data from the patrol's existing unmanned aerial vehicles inventory by raising its altitude. This methodology can be applied to the resulting lower resolution of the sensor data to provide analysis of

sub-pixel targets; potentially capturing previously undetected illegal entries or smuggling routes.

The Joint Inter-Agency Task Force South (JIATF-South) performs drug interdiction operations that involve 'actionable intelligence on the locations of suspect drug trafficking vessels' [4]. As with the border patrol, JIATF-south has a large area to cover on land and sea. Illegal 'drugs coming to the United States from South America pass through a 42 million square-mile transit zone, roughly twice the size of the continental U.S.' [4]. To transport the drug supply, traffickers typically use fishing, high-speed 'go-fast', or low-profile semi-submersible vessels [4]. The go-fast and semi-submersible vessels are small and difficult to detect. Stealth is also a consideration as law enforcement officials need to monitor activity without being detected lest they give away their position or opportunity to apprehend the suspect. Small RPAs at high altitudes would provide the opportunity to constantly monitor traffic while in turn, being difficult to detect. This platform is ideal for the sub-pixel process, which has the potential to identify this small, drug trafficking watercraft traveling over a large, relatively static aquatic background.

### 1.1.3. Overabundance of Data

Data by itself is meaningless. If knowledge is added to this data, it becomes information, which adds value to what is collected. This knowledge can change a previously fruitless search into a productive, efficient, and focused investigation, ultimately providing the end user the capability to make more informed judgments and decisions (Figure 1.2).

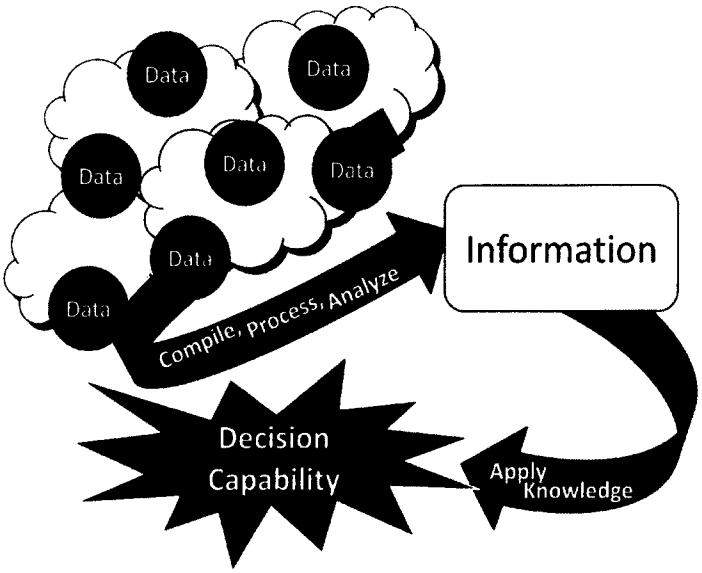



Figure 1.2 Applying Information to Gain Knowledge

A main issue is that the amount of data far outnumbers the trained analysts available to process it into information. Within the intelligence community, the need for applications to help analysts is in the area of image processing used to detect, track and identify targets [5].

There exists a myriad of ways to process images; some of which are discussed in Chapter Four. The desired result is to gain actionable information. There will not always be the proper resolution, the necessary equipment, nor the manpower required to filter through the copious amounts of data. The need to identify novel methods to mitigate these circumstances is paramount.

In low resolution instances, visually analyzing the image to infer information is a daunting task. If a target size is less than a pixel, whole-image and static processing will not be effective. The sub-pixel process described in this dissertation is a concept able to pull actionable information from streams of raw data.

### 1.1.4. Limited Resources

As former Secretary of Defense Donald Rumsfeld advised, 'go to war with the army you have, not the army you might want' [6]. Resources have limitations, such as cost, equipment, and capabilities.

From a sensor based perspective, specifically with satellites, targets of opportunity are limited based on ephemeris data, weather, and target availability. It is not practical or cost effective to go into space and repair a satellite or change out its sensors. It is also not cheap or quick to put up another satellite with a different sensor sweep when the need arises and to have it operational in time.

As discussed earlier, by widening the area of view, thereby decreasing the focus or resolution, one can observe more area at once. The sub-pixel methodology in this dissertation ascribes to this principle. Instead of increasing the number of sensors or platforms in an area, sensors can encompass a larger field of view and still characterize moving targets from the resulting degradation in resolution.

### 1.1.5. Sensor Validation/Verification via Sub Pixel Process

Sub-pixel processing can also revalidate sensor suites by analyzing the performance capabilities and limits of a given platform and sensor for each of its current applications or by identifying new functions and capabilities of existing sensors.

Current sensors applying the sub-pixel process can re-certify their capability through this analysis. The analysis would provide the new parameters for the sensor altitude and resolution and the subsequent ranges of the detectable target's size, velocity and percent pixel radiometry.

Used in conjunction with other sensors, the sub-pixel algorithm can compile comparisons to include altitude versus resolution trade-offs as well as sensor and target constraints for detection and characterization capability.

## 1.2. Dissertation Objectives

The overall objective of this dissertation is to provide an innovative process for imagery data for the purpose of identifying characteristics of sub-pixel mobile target.

The specific objectives for this research involve the development, testing and analysis of the sub-pixel algorithm. The process will study sensor data, identify moving targets within a sensor pixel, and estimate target information.

The follow-on objective after the completion of the dissertation is to incorporate the super-pixel refinement and Refinement Through Tracking (RTT) methods developed in this paper into the sub-pixel process algorithm.

## 1.3. Organization of Dissertation

The second chapter recaptures the problem and defines the assumptions and their handling, while the third chapter develops the database models for the pixel and target used throughout the dissertation.

The fourth chapter discusses the current techniques in the change detection field, investigates three detection algorithms and develops two false alarm refinement algorithm tools. At the end of each section, examples of each process are provided for illustration purposes.

The fifth chapter provides a brief literature search of target analysis methods, and creates two novel processes for analyzing sub-pixel targets. Illustrative examples are also included in this chapter.

Chapter Six describes the tools produced for super-pixel processing and Chapter Seven does a theoretical investigation on improving target analysis via historical target tracking.

The eighth chapter presents the testing scenarios and Chapter Nine provides the results and discussion portion of the dissertation. Chapter Ten focuses on further work and project exploration. The dissertation concludes with Chapter Eleven.

# CHAPTER TWO

# PROBLEM CONCEPT

## 2.1 Problem Statement

This dissertation develops a set of tools to process sensor data and analyze sub-pixel mobile targets.

The basic target concept is that an object moving through the pixel will create a trapezoid intensity profile when viewed temporally (Figure 2.1).

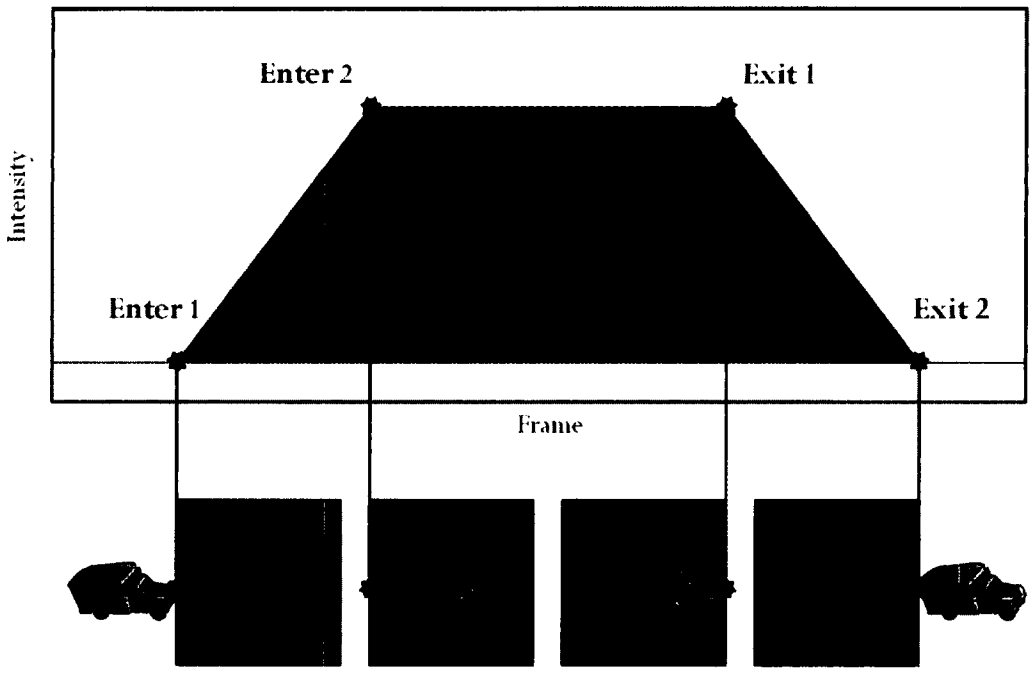



Figure 2.1 Pictoral Description of Critical Target Points

*Enter*$_1$ is the point where the target first starts to enter the pixel, *Enter*$_2$ is the point where the target is fully within the pixel, *Exit*$_1$ is the point where the target first starts to exit the pixel, and *Exit*$_2$ is the point where the target has fully exited the pixel.

By studying this problem geometrically and temporally, it is feasible to discover target characteristics, such as velocity and length.

## 2.2. Assumptions and Handling

### 2.2.1. Sensors

The sub-pixel algorithm is generic in design such that it will perform for a variety of sensors and platforms. However, some basic assumptions concerning the sensors are necessary for algorithm development. Once completely developed and as these assumptions change, adjustments in the affected areas are possible.

The sensor assumptions include that the data from the sensor is a framing system, which gathers instantaneous measurements at a constant sample rate from the entire scene at once. The pixel geometry is rectangular, with target intensities centered on the pixel and having a point spread function contained within the pixel. The sensor bias is considered constant and the measured noise has a Gaussian profile.

The algorithm processes data as a single intensity value, normalized between zero and one. If other types of data are received, such as RGB data, it is converted to a grayscale dataset and normalized.

### 2.2.2. Target

The sub-pixel algorithm detects and analyzes mobile targets smaller than a pixel. The development of an appropriate target model is imperative for algorithm design and assumptions are made to bound the problem.

The basic target shape is rectangular and emits measurements uniformly across its entire surface area. The target velocity is constant and greater than one frame per pixel and the ground slope is negligible in comparison to the altitude of the sensor.

The initial target trajectories travel in the horizontal plane with respect to the sensor array. Studies later in the dissertation involve relaxing this assumption to refine the algorithm performance during target analysis. However, the assumption remains that the trajectory has a short term linear motion such that within a given pixel sequence; the target will maintain a straight line.

# CHAPTER THREE

# DATABASE MODELING

Identifying the existence of a moving target is basically a change detection problem. Whether done temporally or spatially, all detection models 'look' for a difference between the target and the quasi-static background. When finding objects in the sub-pixel realm, the target rarely presents visually identifying features, such as contours and textures. This limits the ability to detect a target using spatial variations. Temporal processing, therefore, becomes the feasible option for the sub-pixel detection algorithm.

The development and testing of the detection algorithm uses statistical models simulating the background and target profiles. Many researchers, such as Zhang and Barniv, focused primarily on target detection, treat small, moving targets as a point source [7] [8]. Others, [9] [10] [11] [12] [13] [14], rely on background suppression techniques to isolate the target and do not use a target model. Caefer et. al classifies temporal profiles of the static object/sky and a moving target and uses the distinctions between these profiles to segment IR images for target detection via a Triple Temporal Filtering process [15]. Tzannes et. al. furthers this work and performs a series of experimental tests which validates the use of a Fermi derivative equation as a model of IR targets [16]. The purpose of this dissertation is to analyze target characteristics of moving vehicles, so a modeled target is essential for testing an evaluation. Tzannes

Fermi derivative as well as a trapezoid depiction, which is novel to this dissertation, provides the target models for the sub-pixel algorithm development.

A generally accepted background model uses a constant signal with additive Gaussian noise [7] [9] [10] [15] [17] [18] [19] [20] [21]. More complex models use an adaptive approach to accommodate shifts in the background scenario [13] [14] [22] [23]. The sub-pixel algorithm employs both the generic background and an adaptive Fast Approximated Power Iteration background model [13] [14] for a comparative analysis.

For the modeled database, the simplest scenario is created, tested and analyzed. The simplest scenario is a static, uniform background with a single, uniform, well-defined target moving at a constant speed in the horizontal direction from left to right. It assumes no effects from noise, jitter, percent radiometry, and change of target trajectory. The scenario is then complicated by removing these assumptions. Once analysis of the simplest scenario occurs, the scenario complicates itself by removing the assumptions and adds in noise and jitter, varying percent radiometry and different target trajectories.

This analysis uses input datasets with grayscale intensity levels. The single intensity scale values keep the computation complexity minimized and works well for a multitude of environments, such as IR. The input intensity values are stored in $Pixel(r, c, f)$ arrays, where $r$ is the pixel row, $c$ is the pixel column and $f$ is the number of frames. For computational purposes and for differing future datasets, all intensity values are converted into normalized values between zero and one.

It is important to note that the sub-pixel algorithm processes data in terms of frame count vs. time. The correlation between the two is expressed by

$$t = \frac{frame}{SampleRate},$$

where $t$ is the passage of time during the dataset, *frame* is the total number of frames in the dataset, and *SampleRate* is the rate that the data samples are taken in frames per second.

### 3.1. Basic Background Model

Assuming a quasi-static background with all the other assumptions in place, the background can be considered a constant signal.

$$I_{back}(r, c, f) = C,$$

where $I_{Back}$ is the background intensity matrix with indices of row, $r$, column, $c$, and current frame, $f$. $C$ is the constant background signal.

### 3.2. Basic Target Model

Unless the target has an identical intensity to the background, a uniform target moving across the pixel will experience either a positive intensity change throughout the transit or a negative intensity change. The temporal profile, assuming a constant speed across the pixel, displays a trapezoid geometric configuration. When the target enters the pixel, the intensity increases or decreases linearly until the target is fully within the pixel. With the target fully in the frame, the intensity steadies out. Once the target starts to exit the pixel, the intensity decreases or increases at the same rate, but of opposite slope of the entrance. Once the target fully exits the pixel, the pixel will reflect the background signal (Figure 3.1).

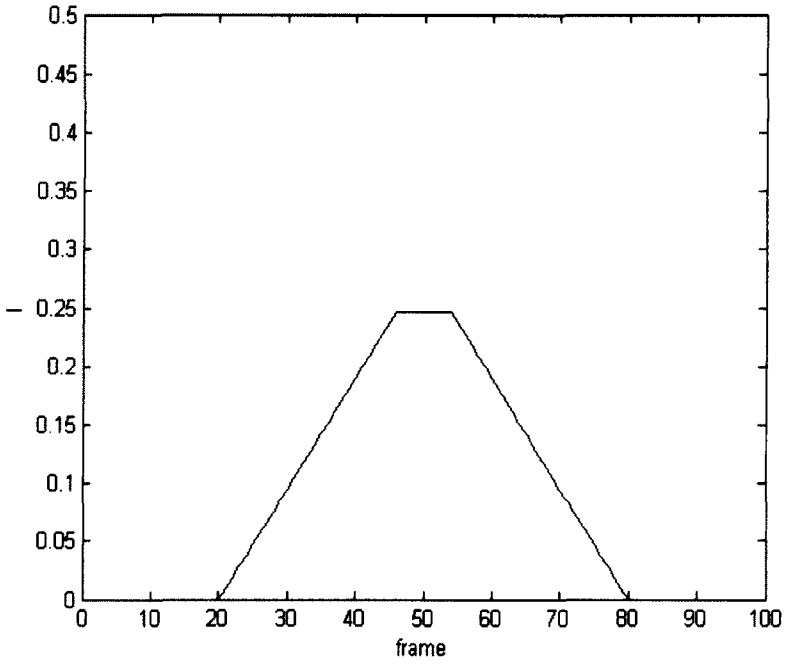



Figure 3.1 Trapezoid figure of target within the pixel

The trapezoidial model equation is given by

$$I_{Target}(r,c,f) = \begin{cases} \dfrac{(q)}{(Enter_2 - Enter_1)}(f - Enter_1), & if\ Enter_1 \le f \le Enter_2 \\ q, & if\ Enter_2 < f < Exit_1, \\ \dfrac{(-q)}{(Exit_2 - Exit_1)}(f - Exit_1) + q, & if\ Exit_1 \le f \le Exit_2 \end{cases}$$

where $I_{Target}(r,c,f)$ is the temporal target intensity matrix with indices of row, $r$, column, $c$, and current frame, $f$. $Enter_1$ is the frame at which the target enters the pixel, $Enter_2$ is the frame at which the target is fully in the pixel, $Exit_1$ is the frame at which the target starts to leave the pixel, $Exit_2$ is the frame at which the target has fully exited the pixel, and $q$ is the uniform intensity of the target. Due to the constant velocity assumption, the target entry and exit frames are equivalent,

$$(Enter_2 - Enter_1 = Exit_2 - Exit_1).$$

### 3.3. Quantized Fermi Target Model

The trapezoidal model is not continuous and an alternative model for comparison purposes is useful. Previous studies by Tzannes use a Fermi derivative to generate the pulse-like shape of IR targets [16] using the equation

$$f_{df}(k;a,b,c,d) = \frac{ae^{\frac{k-b}{c}}}{(e^{\frac{k-b}{c}}+1)^2} + d, \tag{3.1}$$

where $k$ is the frame number, $a$ is proportional to the target intensity, $b$ is the related to the frame where the target is fully within the pixel, $c$ controls the width of the function or the velocity parameter, and $d$ is the background level. This equation uses the normalized intensity vs. the grayscale intensity.

A trapezoid shape is desired because the edges of the max intensities are critical values for target analysis, which is discussed in Chapter Five. Applying a threshold quantization forces the Fermi pulse-like shape into a quasi trapezoidal signal (Figure 3.2).

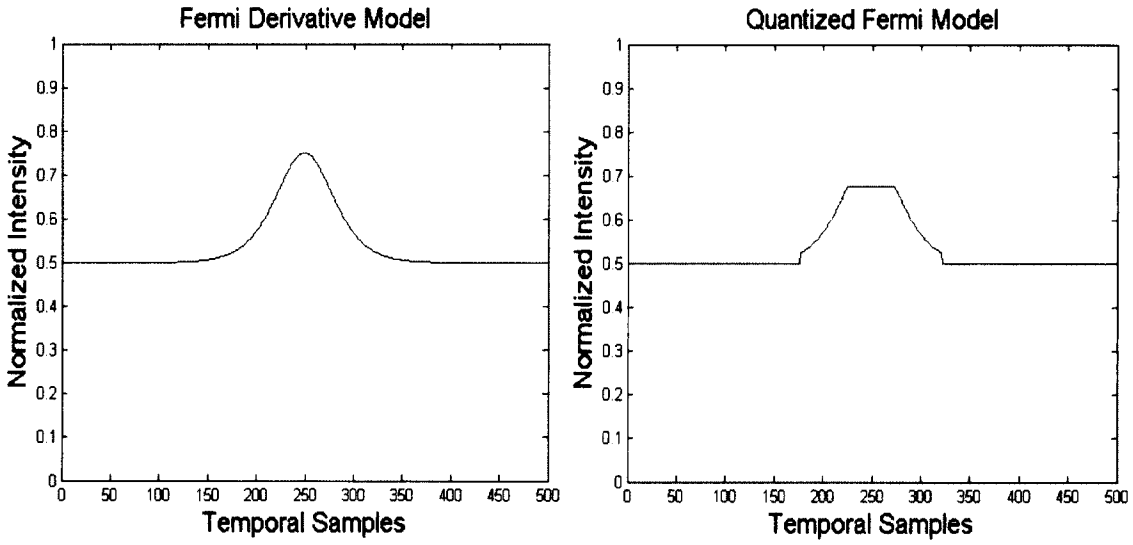



Figure 3.2 (A) Fermi Derivative Model and (B) Quantized Derivative Model Output

## 3.4. Incorporating Background and Target

### 3.4.1. Trapezoid Background and Target Model

Unlike a multi-pixel target, where the entire pixel is replaced upon passing, when a sub-pixel target moves into the pixel, it replaces only a percentage of the pixel background (Figure 3.3).

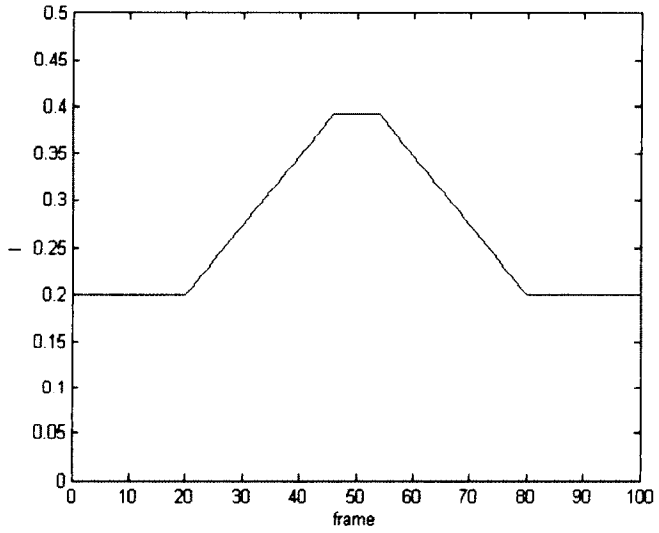



Figure 3.3 Temporal Profile of a Target Moving Through a Pixel

The critical target points from Chapter Two are used to develop the trapezoid model of the mobile target, which is given by

$$I_{Pixel}(r,c,f) = \begin{cases} I_{Back}(r,c,f), & if\ f < Enter_1 \\ \dfrac{q}{(Enter_2 - Enter_1)}(f - Enter_1) + I_{Back}(r,c,f), & if\ Enter_1 \le f \le Enter_2 \\ (1 - RTP)I_{Back}(r,c,f) + RTP \cdot I_{Target}(r,c,f), & if\ Enter_2 < f < Exit_1, \\ \dfrac{-q}{(Exit_2 - Exit_1)}(t - Exit_1) + q, & if\ Exit_1 \le f \le Exit_2 \\ I_{Back}(r,c,f), & if\ f < Exit_2 \end{cases}$$

where $I_{Pixel}(r, c, f)$ is the pixel intensity matrix, $I_{Target}$ is the temporal target intensity matrix, and $I_{Back}$ is the background intensity matrix. All of the matrices use indices of row, $r$, column, $c$, and current data frame, $f$. $RTP$ is the ratio of the target size to the pixel size, $q$ is the uniform target intensity, and $Enter_1$, $Enter_2$, $Exit_1$, and $Exit_2$ are previously defined.

The main issue with incorporating the background is having enough intensity variation between the background and the target to be able to detect the target. The intensity variation is directly proportional to $RTP$. As $RTP$ decreases, the success of the detection algorithm decreases.

### 3.4.2. Quantized Fermi Derivative Background and Target Model

The Fermi Derivative Model (FDM) was not originally designed for sub-pixel targets and the model assumes the target fully covers the pixel ($RTP \geq 1$). However, through analysis, the FDM can work on sub-pixel targets by making a few adaptations, such as correlating the trapezoid model and the FDM to gain knowledge of the target's parameters (Figure 3.4).

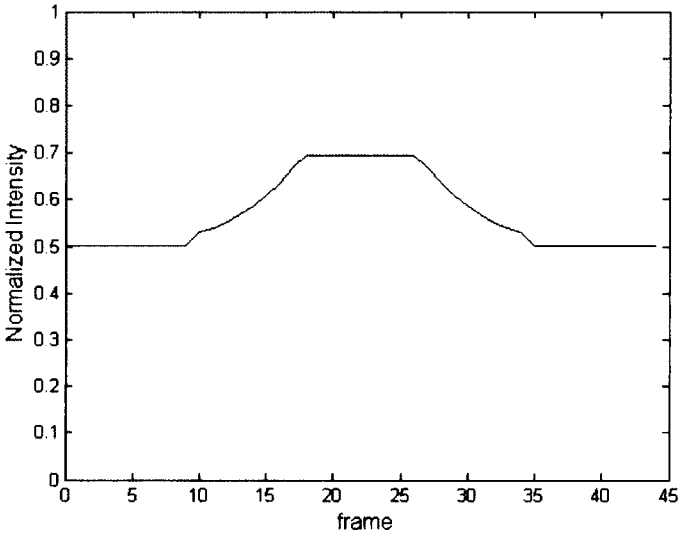



Figure 3.4 Illustrates the FDM as applied to a sub-pixel environment

To relate the FDM into a sub-pixel environment, each parameter within the FDM (Eq. 3.1) defines and correlates the target and sub-pixel environment. The parameters provide the relationship to the trapezoid model needed for ground truth knowledge during testing.

*3.4.2.1. Finding d parameter*

From Section 3.3, $k$ is the frame number and $a$, $b$, $c$, and $d$ are constants. As $k$ approaches infinity,

$$\lim_{k\to\infty} \frac{ae^{\frac{k-b}{c}}}{(e^{\frac{k-b}{c}}+1)^2} + d = d, \tag{3.2}$$

and the pixel intensity reduces to $d$,

$$d = I_{Back},$$

which for a staring sensor would be the average background intensity.

*3.4.2.2. Finding b parameter*

The $b$ parameter is related to the midpoint of the Fermi derivative, which correlates to the midpoint frame of the target signature. The target transit frames, $f_{transit}$, are found by

$$f_{transit} = Exit_2 - Enter_1,$$

where $Enter_1$ is the first frame of target entry and $Exit_2$ is the frame where the target exits the pixel. The transit frames are halved and added to the frame of entry to find the target signature midpoint, $b$,

$$b = \frac{f_{transit}}{2} + Enter_1.$$

*3.4.2.3. Finding a parameter*

At the midpoint frame ($k = b$), Eq. 3.2 reduces to:

$$f_{df}(k = b; a, b, c, d) = \frac{a}{2^2} + d,$$

Referencing the trapezoid model in Section 3.4.1., when $k = b$, the pixel intensity is the intensity value of the target fully within the pixel frame such that

$$f_{df}(k = b) = (1 - RTP)I_{Back} + RTP \cdot I_{Target},$$

where $RTP$ is the ratio of the target size to the pixel size, $I_{Target}$ is the target intensity, and $I_{Back}$ is the background intensity. These equations are combined to solve for parameter $a$,

$$a = 4(\left[(1 - RTP)I_{Back} + RTP \cdot I_{Target}\right] - d).$$

*3.4.2.4. Finding c parameter*

The $c$ parameter relates to the width of the Fermi derivative function and is related to the trapezoid profile via the equation:

$$c = N(Exit_1 - Enter_2),$$

where $Enter_1$ is the first frame of target entry, $Exit_2$ is the frame where the target exits the pixel, and $N$ is an adjustable parameter used to control the width estimation. $N$ values between one and two effectively model the targets used in this dissertation.

## 3.5. Utilizing Realistic Inputs for Modeled Database

Army vehicle parameters are used for the database modeling because they are known values that can be tested and verified.

Table 3.1 lists the length, width and maximum speed of four common Army vehicles. Throughout the testing scenarios, these Army vehicles will be used as the modeled targets [24].

Table 3.1 Army Wheeled Vehicles to Test

| | Length | Width | Max Speed |
|---|---|---|---|
| HUMMWV | 4.57 m | 2.16 m | 29.06 m/s |
| HEMTT (M977) | 10.19 m | 2.44 m | 25.48 m/s |
| M1070 | 24.89 m | 3.68 m | 20.12 m/s |
| - Tractor | 9.09 m | 2.59 m | 20.12 m/s |
| - Trailer | 15.80 m | 3.68 m | 20.12 m/s |
| - w/ payload | | | 13.41 m/s |
| PLS | | | |
| - Truck | 10.95 m | 2.44m | 25.48 m/s |
| - Trailer | 8.32 m | 2.43 m | 25.48 m/s |
| - w/ payload | | | 15.64 m/s |

The target's information (length, width and velocity), the pixel size and the sensor sample rate are related to the trapezoid model parameters for programming and signal insertion (Figure 3.5). The equations in this section and throughout the testing scenarios are based on an assumed horizontal trajectory. Alternate target trajectories and their change effects are discussed in Chapter Seven.

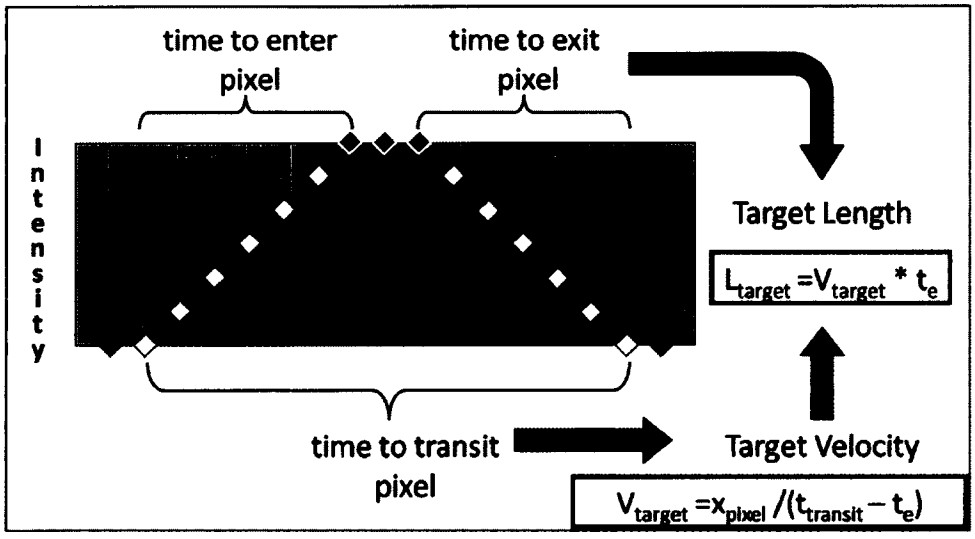



Figure 3.5 Derivation of basic parameters using pixel intensity values

The basic equation for velocity,

$$Velocity = \frac{\Delta d}{\Delta t} = \frac{x_{pixel}}{Exit_1 - Enter_1} = \frac{x_{pixel}}{Exit_2 - Enter_2},$$

can be related to a vehicle crossing a pixel, where the change in distance, $\Delta d$, is the horizontal length of the pixel, $x_{pixel}$ and the change in time, $\Delta t$, is the amount of frames it takes for one point of the target to fully transverse the pixel. $Enter_1$ is the first frame of target entry, $Enter_2$ is when the target fully enters the pixel, $Exit_1$ is when the target starts to exit the pixel, and $Exit_2$ is the frame where the target exits the pixel.

The front end of the target, $f_{entry}$, correlates to

$$f_{entry} = Enter_2 - Enter_1,$$

while the back end of the target, $f_{exit}$, correlates to

$$f_{exit} = Exit_2 - Exit_1.$$

Note that at this point, the term frame is interchanged with time, knowing that there is a sample rate conversion factor between frames/s and m/s, shown by

$$t = \frac{frame}{SampleRate},$$

where $t$ is the passage of time during the dataset, *frame* is the total number of frames in the dataset, and *SampleRate* is the rate that the data samples are taken in frames per second.

The velocity can also be related to the length of the target, $L_{target}$ by

$$V_{target} = \frac{L_{target}}{f_{entry/exit}},$$

where $f_{entry/exit}$ is the number of frames it takes the target to either enter or exit the pixel.

The target transit time can be calculated by

$$f_{transit} = \frac{x_{pixel}}{V_{target}} + f_{entry},$$

where $f_{transit}$ is the number of frames it takes the target to completely transverse the pixel.

From these equations, and knowing the frame number where the target starts to enter the pixel, the critical target points can be found by:

$$Enter_2 = f_{entry} + Enter_1,$$

$$Exit_1 = Exit_2 - f_{entry},$$

and

$$Exit_2 = Enter_1 + f_{transit}.$$

From the vehicle and the pixel area, the ratio of target to pixel, *RTP* can be found via

$$RTP = \frac{L_{target} W_{target}}{x_{pixel} y_{pixel}},$$

where $L_{target}$ is the length of the target, $W_{target}$ is the width of the target, $x_{pixel}$ is the pixel size in the x-direction, and $y_{pixel}$ is the pixel size in the y-direction.

### 3.6. Modeling Multiple Pixels

The sub-pixel algorithm is designed to find target characteristics for a single pixel. However, recording and comparing the target's information across multiple arrays allows for jitter effects, target refinement and target tracking.

To model the target across multiple pixels, an nxm array of background pixels is created, such as in Figure 3.6, although the figure shows more than just the background. The background is initially uniform across all pixels, but later testing will allow for background pixel variation to further test jitter effects.

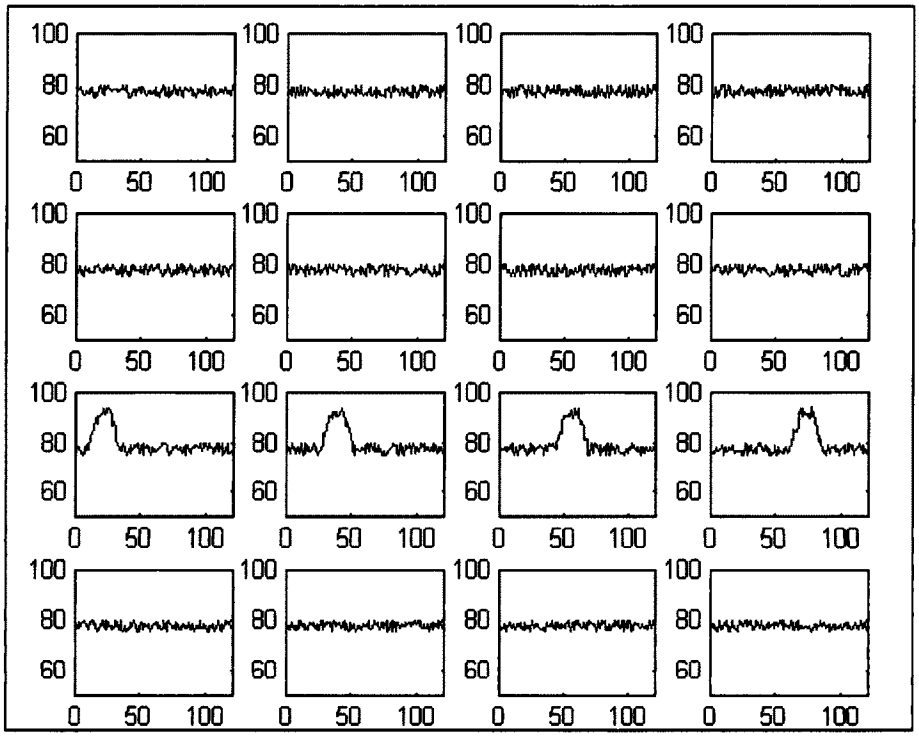



Figure 3.6 Modeled target moving horizontally in a single row

The first multiple-pixel dataset uses a target model that moves in the horizontal direction from left to right and only traverses in a single row. The development of the database is created in the same manner of the single pixel, with the exception of the time of entry in each pixel,

$$Entry_{1(i+1)} = Exit_{1i},$$

where the frame of target entry in the next pixel, $Entry_{1(i+1)}$, is equivalent to the frame the target starts to exit the previous pixel, $Exit_{1i}$, and $i$ is the next column in the row.

The second multiple-pixel database uses the same target model with the exception that the single target traverses in two rows, splitting the pixel, so that half the target is in one row and the other half is in the row immediately below it. To model this scenario, the target's width was reduced by a factor of two (Figure 3.7).

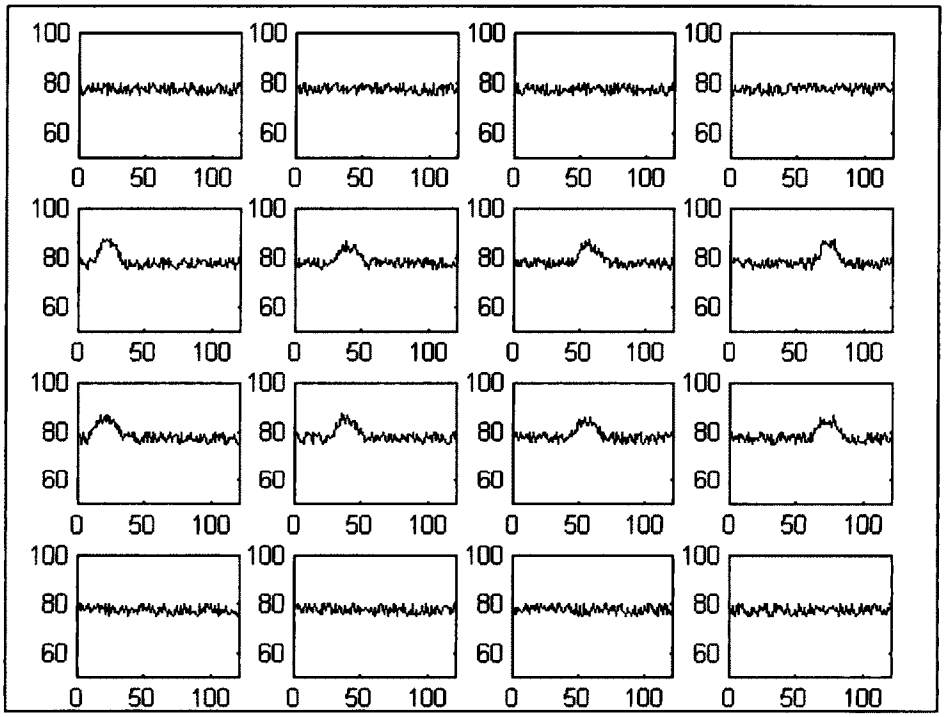



Figure 3.7 Single target with horizontal motion split among two rows

Variations in the datasets have differing background intensities between pixels and the ability to choose the split ratio of the target's width. Due to the sub-pixel nature of the targets and the horizontal direction of travel, the split will only happen between two rows. Changes in this assumption create a much more complex insertion method, which will be discussed theoretically in Chapter Seven.

## 3.7. Removing Assumptions

Noise and jitter is sensor specific. After the target and the background model are complete, noise is added to model the sensor's surroundings and jitter is added over the noise to ensure frame uniformity, i.e. to make sure the target, background and noise all jitter together. This method and order of removing assumptions better models reality.

### 3.7.1. Adding Noise

Noise is assumed to be a random process, modeling an independent Gaussian function,

$$I_{Total}(t) = I_{Pixel}(t) + N(t),$$

with zero mean and standard deviation, $\sigma$, found via

$$N(t) \sim N(0, \sigma),$$

where $I_{Total}$ is the total pixel value, $I_{Pixel}$ is the original pixel, and $N$ is the noise. A commonly accepted noise profile, with widely accepted parameters, used with detection algorithms is the independent Gaussian function [7]. The modeled noise is shown in Figure 3.8.

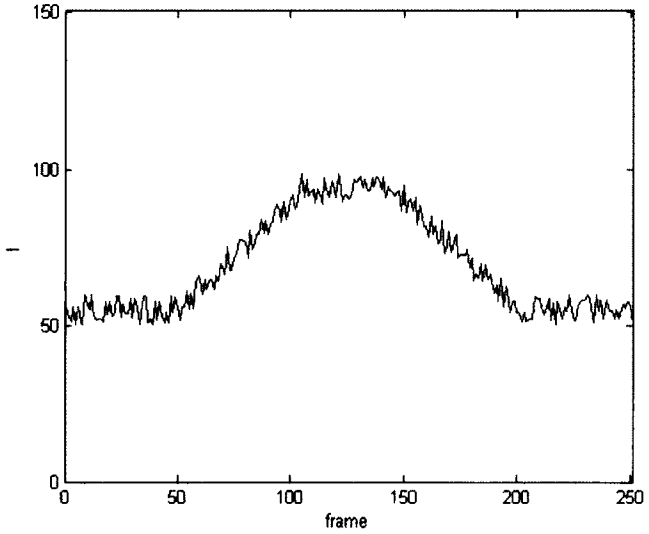



Figure 3.8 Noise added after the target insertion algorithms

### 3.7.2. Adding Jitter

Sensor jitter is a major concern in digital imaging [25]. If the jitter between pixels is significant, the background intensity no longer remains quasi-static. Jitter is specific to each individual sensor and its environment. Over time, jitter can be predicted and removed. However, the techniques employed for jitter removal could smooth out or completely erase the sub-pixel target. For this dissertation, an analysis of jitter effects will be performed in order to find the sub-pixel target algorithm's performance and limitations. This dissertation investigates two types of jitter, translational jitter and random jitter.

#### *3.7.2.1. Translational Jitter*

As a generalization, translational jitter assumes a horizontal linear motion, with the sensor behaving as a simple harmonic oscillator [26] such that

$$Shift(t) = A\sin\left(\frac{2\pi}{T}t + \varphi\right),$$

where *Shift(t),* is the current sensor shift position, $A$ is the total jitter percent shift, $T$ is the period of the jitter cycle, and $\varphi$, is the phase, which is set to zero for this model. Figure 3.10.A shows the jitter from the hardware perspective and Figure 3.10.B shows the jitter from the signal perspective.

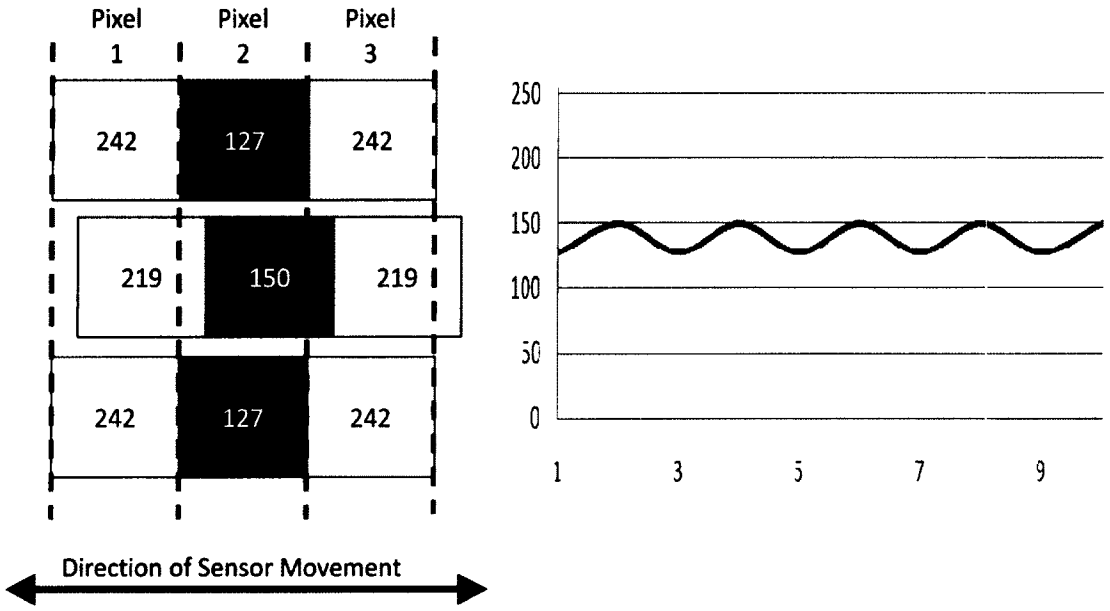



Figure 3.9 (A) Pictorial Description of Translational Jitter (B) Intensity variation due to translational jitter

The pixel intensity variations due to sensor jitter can be expressed as a linear combination of the pixels related to the percent jitter shift by

$$Pixel_{2_{new}}(t) = \begin{cases} \big(1 + Shift(t)\big)Pixel_2 - Shift(t)Pixel_1, & for\ Shift(t) < 0 \\ Pixel_2, & for\ Shift(t) = 0, \\ \big(1 - Shift(t)\big)Pixel_2 + Shift(t)Pixel_3, & for\ Shift(t) > 0 \end{cases}$$

where $Pixel_{2new}$ is the jittered pixel and $Pixel_1$, $Pixel_2$, and $Pixel_3$ are the original pixels as shown in Figure 3.9.A.

The pixels on the frame edges will substitute the $Pixel_2$ value for the missing parameter. Jitter will be modeled at different percents in order to analyze its effect on the algorithm's ability to detect and characterize targets.

*3.7.2.2.* *Random Jitter*

Random jitter allows for the entire pixel array to vary with a uniform distribution for a specified jitter parameter, *J*. The jitter parameter can be as large as a multiple pixel shift or restricted within a pixel fraction. The jitter equations are calculated via:

$$J_{row} = J(1 - 2[1]),$$

and

$$J_{col} = J(1 - 2[1]),$$

where $J_{row}$ is the random row shift and $J_{col}$ is the random column shift. Both shift factors are randomly updated per frame with a uniform distribution on the interval [-J, J]. The two shift factors allow for multidirectional shifting along the frame (Figure 3.10).

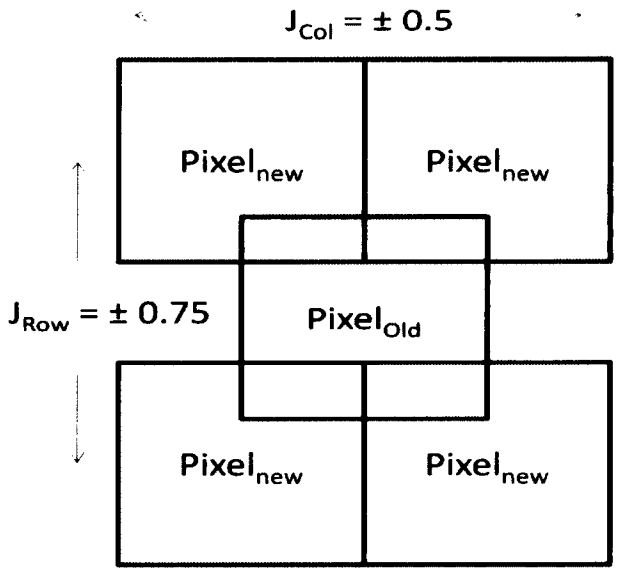



Figure 3.10 Multidirectional shifting along a frame caused by jitter

The pixel intensity variations due to sensor jitter can be expressed as a linear combination of the pixels related to the percent of jitter shift using

$$I_{Pixel_{Jitter}}(r,c,f) =$$

$$I_{Pixel}(r + \lfloor J_{row}(f) \rfloor, c + \lfloor J_{col}(f) \rfloor, f) \cdot (1 - J_{row}(f))(1 - J_{col}(f))$$

$$+ I_{Pixel}(r + \lfloor J_{row}(f) \rfloor, c + \lceil J_{col}(f) \rceil, f) \cdot (1 - J_{row}(f)) J_{col}(f)$$

$$+ I_{Pixel}(r + \lceil J_{row}(f) \rceil, c + \lfloor J_{col}(f) \rfloor, f) \cdot J_{row}(f)(1 - J_{col}(f))$$

$$+ I_{Pixel}(r + \lceil J_{row}(f) \rceil, c + \lceil J_{col}(f) \rceil, f) \cdot J_{row}(f) J_{col}(f),$$

where $I_{PixelJitter}$ is the resulting jittered pixel matrix, $I_{Pixel}$ is the original pixel matrix, and the indices of the matrices are row, $r$, column, $c$, and current frame, $f$. $J_{row}$ is the random row shift and $J_{col}$ is the random column shift. If the row or column indices are exceeded during the calculation, in which case the jitter shift is past the pixel array frame, the last row or column index is used as a substitution.

# CHAPTER FOUR

# CHANGE DETECTION

Target detection is essentially a change detection problem and can be done in the spatial, temporal or spatio-temporal domain. Spatial-image processing analyzes the intensity feature variation in a single frame and works well for multi-pixel or high-contrast targets. Some of these processing techniques include Matched Filtering (MF) [27], Dynamic Programming Algorithm (DPA) [28], and Multistage Hypothesis Testing (MAHT) [9]. Temporal-image processing capitalizes on the intensity changes over a sequence of frame data and works well for small, dim, point-like moving targets. Spatio-temporal processing does a combination of both.

The main goal is to differentiate the target from the background signal. As the target moves through the pixel, a pulse-like signal is generated, interrupting the background signal. This scenario is perfect for temporal analysis.

A straightforward change detection method uses a simple differencing method,

$$D(f) = I_2(f) - I_1(f),$$

where $I$ represents two images, $D$ is the differences between the images, and $f$ is the frame sequence. For this method, the decision factor on detection is determined by an empirical threshold [29]. Rosin did an extensive survey on different methods to choose the proper threshold [30] [31]. Instead of automating the threshold process, Simonson allows for flexibility by keeping the threshold variable a tunable parameter [13]. This

simple differencing threshold change detection method is computationally inexpensive and provides a quick check method for target detection. However, this method is sensitive to noise and variations in background intensity changes [29]. To mitigate these problems, adaptive background models and thresholds are effective [32].

Extensive research in background modeling and suppression has been done and is especially useful for temporal problems with either historical data or multiple frame sequences, where changes happen over a short span of time [13]. Some background segmentation techniques involve neural network with edge constraints [33], fuzzy threshold and edge detection [34], contrast boxes, double gated filter, spoke filter [35], morphological techniques: top hat transform [36], and Gabor filtering functions [36]. One form of background suppression involves wavelet transformations [11] [21] [23]. An advantage of wavelet transforms is that in target detection techniques they can operate as matching filter, multi-resolution image analyzer, a multi-dimensional image analyzer, a singularity detector, and an orthogonal extractor [37]. Problems with this method are its intrinsic complexity [14]. Fast subspace tracking methods utilize a power iteration technique, which can also be computationally expensive [38] [39] [40]. Badeau et al. uses a low complexity update to the background estimation called a Fast Approximated Power Iteration (FAPI) subspace tracking [14]. Simonson shows success with this method in noisy and highly jittered environments [13].

This summary of change detection methods just scratches the surface of the abundant information found in the academic communities. The target detection is a necessity in order to proceed to the target analysis and is part of the overall sub-pixel process. However, a particular detection mechanism is not a necessity as the modularity of the design allows for interchangeability. For initial algorithm design, two generic and

computationally inexpensive methods, and one more robust, adaptive background estimation method comprise the change detection methodology. The temporal change detection methods described in this section are the Threshold Change Detection, the Duration Change Detection, and the Fast Adaptive Power Iteration (FAPI) Detection methods.

The Threshold and Duration methods are straightforward comparison methods, are computationally efficient and work well in low noise and jitter environments. These methods are good tools for quick and easy target detection which is especially useful in the initial testing of the sub-pixel target analysis. When noise and jitter effects are introduced into the testing scenarios, a more robust detection scheme, like FAPI, is a necessary tool to have in the analysis arsenal.

When working with the detection algorithms, it is important to keep as much of the signal as possible, especially in a low Signal-to-Noise Ratio (SNR) environment. For this reason, pre-processing smoothing techniques are not considered due to the high probability that the target signal will be lost or corrupted. In all the algorithms described in this section, the original pixel sequence is preserved globally so that it can be utilized in its original state during the analysis section.

A major product of the change detection algorithms is the Binary Target Matrix. This matrix stores the target detection results as either a one for a target or a zero for no target. This matrix helps differentiate between actual targets and false alarms and helps discover the target's entry and exit frames for a particular pixel.

For this dissertation, the sensor platform is assumed to be a constant staring camera, where the background is quasi-static. Although over time, it may be possible to characterize the background, noise and jitter, for this algorithm, it is assumed that this *a priori* knowledge does not exist.

## 4.1 Threshold Change Detection

The image sequences of the sensor, $I_{Pixel}$, are stored in 3D arrays,

$$I_{Pixel}(NRow, NCol, NFrames),$$

where *NRow* is the total number of rows in the sensor frame, *NCol* is the total number of columns in the sensor frame, and *NFrames* is the total number of frames in the data sequence.

The total number of pixels per frame, *NPixels*, is found by

$$NPixels = NRow \cdot NCol,$$

and a single pixel is referenced as

$$I_{Pixel}(r, c, f),$$

where $r$ is the current row, $c$ is the current column, and $f$ is the current frame.

At each discrete time frame, a new image is stored into the $I_{Pixel}$ array. The objective at each frame is to determine whether the intensity of the pixel of the new frame is consistent with the current temporal background model of the pixel. A target is detected in those pixels with observed intensities that vary from the model.

Since time is a factor, a general method for comparison is simple normalized differences [25]:

$$\Delta_{Pixel}(r, c, f) = \frac{I_{Pixel}(r, c, f) - I_{Back}(r, c, f)}{\zeta_{Pixel}(r, c, f-1)},$$

where $\Delta_{Pixel}$ is the normalized difference, $I_{Back}$ is the current background model and $\zeta_{Pixel}$ is the previous standard deviation estimate. The standard deviation estimate can be initialized with the first few background frames, $B$, such that

$$\zeta_{Pixel}(r,c,0) = \sqrt{\frac{\sum_{f=1}^{B}(I_{Pixel}(r,c,f) - I_{Back}(r,c,f))^2}{B}}. \quad (4.1)$$

Recursive standard deviation estimates are updated with

$$\zeta_{Pixel}(r,c,f) = \sqrt{\gamma\zeta_{Pixel}^2(r,c,f-1) + (1-\gamma)(I_{Pixel}(r,c,f) - I_{Back}(r,c,f))^2},$$

where $\gamma \epsilon [0,1]$ is the forgetting factor which applies a weight to the old and new frames. The forgetting factor allows for the ability to apply importance to new information. In essence it is a weighting factor, preserving the opportunity to tailor the standard deviation function later [25].

The assessment of compliance can be determined via thresholding, where

$$\Delta_{Pixel}(r,c,f) > T$$

is the condition required for target detection. The threshold value, $T$, is user defined and sensor specific. An effective threshold value compares the target detection rate vs. the false alarm rate, discussed in Section 4.4., where the false alarm rate is much lower than the target detection rate [25]. A generic method, which allows for algorithm adaptability, uses a threshold value as $N$ standard deviations from the modeled background [41] [42] [43]. Target pixels are those frames that satisfy

$$\Delta_{Pixel}(r,c,f) > N\sigma,$$

where $\sigma$ is the standard deviation and $N$ is the tunable integer factor.

### 4.1.1. The Background Model

Due to the relatively small timeframe of the pixel sequence, it can reasonably be assumed that the background will not change significantly in temporal intensity throughout the sequence capture. The initial standard deviation estimate found using Eq. 4.1 and the first few data frames serves as a simplistic background model, with the stipulation that no targets are located in these frames such that the background intensity, $I_{Back}$, is given by

$$I_{Back} = \zeta_{Pixel}(r, c, 0),$$

where $\zeta_{Pixel}$ is found using Eqn. 4.1.

This method of background estimation eases the burden of requiring *a priori* knowledge of the scene and allows for utilization in real or near-real-time processing. After establishing a historical database, it would be possible to cross check new background models to ensure accuracy or determine long term background changes.

### 4.1.2. Establishing the Initial Threshold

The threshold value determination varies with each sensor. The effectiveness of the threshold value is measured via the target detection rate vs. the false alarm rate.

One method of detection assumes the first few frames are background data and Gaussian noise. The intensity values in these frames are used to find the background value and the noise threshold. The standard deviation, $\sigma$ is given by

$$\sigma = \sqrt{\frac{\sum(I_{Pixel}(r, c, f) - I_{Back})^2}{n}},$$

where $I_{Pixel}$ is the pixel intensity values, $I_{Back}$ is the mean value of the first few frames, and $n$ is the number of frames used for the background calculation. The threshold value is calculated as

$$Threshold = \pm N\sigma,$$

with $N$ being the number of standard deviations. For example, an $N$ value of two captures approximately 95% of the noise deviation, assuming a Gaussian profile [44]. The variability of $N$ is important because if the SNR is very low, a large $N$ value will completely ignore the target. Too low of a $N$ value will cause a higher false alarm rate, which can be mitigated with the False Alarm Rate filters (FAR filter), which is explained in Section 4.4. The range of $N$ values varies based on sensor. Testing refines the $N$ value to achieve the ideal number.

4.1.3. Threshold Detection Binary Matrix

The detection binary matrix, $Pixel_{Binary}(r,c,f)$, stores the pixel target detections using

$$Pixel_{Binary}(r,c,f) = \begin{cases} 1, & if\ \Delta_{Pixel}(r,c,f) > T \\ 0, & if\ \Delta_{Pixel}(r,c,f) \leq T' \end{cases}$$

where the row, $r$, column, $c$, and frame, $f$, represent the matrix indices, $\Delta_{Pixel}$, is the normalized difference, and $T$ is the threshold. A value of one indicates a detected target. The temporal profile of the matrix gives a good visual indication of actual targets versus the false alarms and can be used as a tool in the FAR filter algorithm to reduce the number of false alarms.

A potential problem with threshold detection is that the target may enter in the first few frames, skewing the noise threshold which would allow the target to go undetected. This is mitigated using Duration Detection (Section 4.2).

### 4.1.4. Threshold Detection Example

A HUMMWV traveling at 23.5 m/s is inserted into a 30x30 $m^2$ pixel sampled at a rate of 120 frames/s (Figure 4.1.A). Noise is added to the signal (Figure 4.1.B) and then the array is sent to the Threshold Detection algorithm.

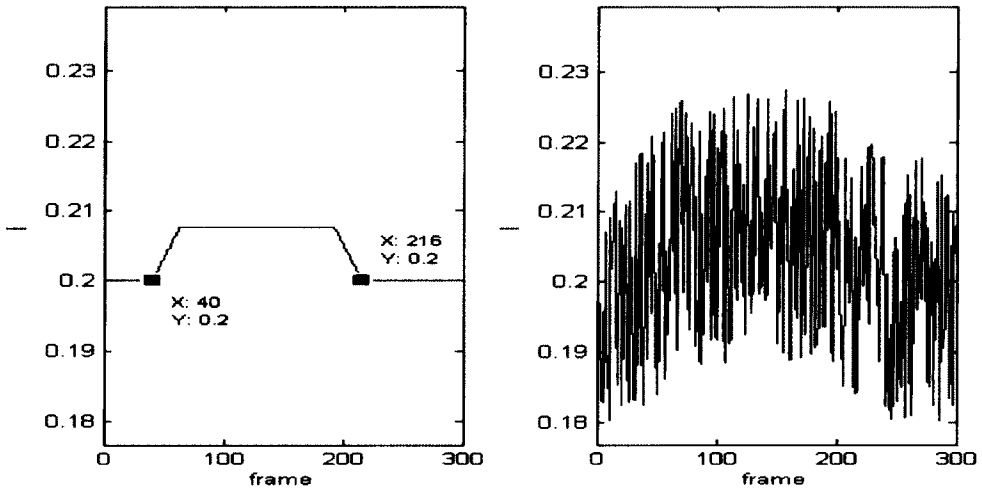



Figure 4.1 (A) HUMMWV signal before noise and (B) after noise

The standard deviation estimates are compared to the background model threshold. The frames that exceed the threshold are shown in Figure 4.2.

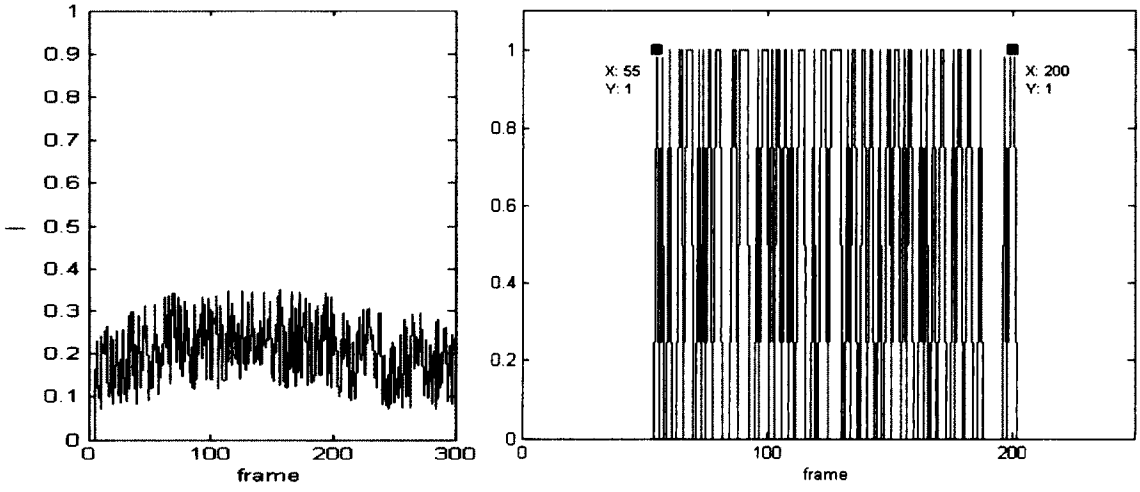



Figure 4.2 Standard deviation estimates and Threshold Binary Matrix

In depth analysis and discussion of the target transit points' accuracy, the detection vs. false alarm rates, noise, jitter and threshold selection can be found in Chapter Nine.

## 4.2. Duration Change Detection

Duration Change Detection finds the most frequent data points and uses this intensity as a critical point for filtering. The philosophy behind this method is that for the majority of the time, the pixel will be comprised solely of background information, with targets sporadically interrupting the constancy. As a result, the critical filter point is considered the background model. The critical filter point is found at the highest frequency bin of a frame sequence histogram.

### 4.2.1. Finding the Minimum Required Sample Set

Duration Detection capitalizes on the stationary background characteristic of a staring sensor. Assuming a single target profile, a minimum number of sample frames need to be recorded ensuring the background data points are the majority in the set. In this scenario, a duration filter segments out the background signal and leaves only the target profile. Whole and partial targets can be detected during any portion of the dataset, providing this condition is met:

$$\#frames_{background} > (f_{transit} - 2f_{entry}) + \epsilon,$$

where $\#frames_{background}$ is the minimum number of background data frames required, $f_{transit}$ is the target transit time, and $f_{entry}$ it the number of frames it takes for the target to fully enter the pixel. The error term, ε, accounts for rounding error due to an integer requirement for the frame number.

If the sensor pixel size and sample rate, *SampleRate*, are known, the total minimum frame number required in the dataset to filter out the background signal for specific targets can be found. The number of frames the target will be in the pixel, $f_{transit}$, is found via

$$f_{transit} = \frac{(x_{pixel} + L_{target})SampleRate}{V_{target}},$$

where $x_{pixel}$ is the size of the pixel in the x-direction, $L_{target}$ is the length of the target, and $V_{target}$ is the target velocity. In order for the duration process to work, the number of background frames, $\#frames_{background}$, must be larger than $f_{transit}$ such that

$$\#frames_{background} > \frac{(x_{pixel} - L_{target})SampleRate}{V_{target}} + \epsilon.$$

The total number of frames required is given by

$$\#frames_{total} > \#frames_{target} + \#frames_{background}$$

and

$$\#frames_{total} > \frac{2x_{pixel}SampleRate}{V_{target}} + \epsilon.$$

For example, for each of the four Army vehicles defined earlier, the expected targets, the minimum number of frames needed to detect each vehicle was calculated and presented in Table 4.1.

Table 4.1 Example Frame Calculation Table for Army Vehicles

| Vehicle | Max Speed (m/s) | Pixel Size(m) | Sample Rate (frame/sec) | Min Frame # |
|---|---|---|---|---|
| HUMMWV | 29.06 | 10 | 30 | 21 |
| HEMTT (M977) | 25.48 | 10 | 30 | 24 |
| M1070 | 20.12 | 10 | 30 | 30 |
| - Tractor | 20.12 | 10 | 30 | 30 |
| - Trailer | 20.12 | 10 | 30 | 30 |
| - w/ payload | 13.41 | 10 | 30 | 45 |
| PLS | | | | |
| - Truck | 25.48 | 10 | 30 | 24 |
| - Trailer | 25.48 | 10 | 30 | 24 |
| - w/ payload | 15.64 | 10 | 30 | 39 |

4.2.2. Create Histogram and Establish Filter Boundaries

The histogram computes the frequency distribution of the intensity elements in a pixel dataset and sorts them into a number of discrete bins, $b$. The bins have equal width of

$$w = \frac{B_u - B_l}{b},$$

where $B_u$ is the upper limit of the histogram and $B_l$ is the lower limit of the histogram. The pixel dataset used for the histogram is a normalized dataset, with intensity values between zero and one, which are $B_l$ and $B_u$ respectively. The centers of the bins are found by the equation:

$$center = B_l + \left(k + \frac{1}{2}\right)w \quad k = 0,1,\ldots,b-1.$$

The input values that fall on the border between two bins are placed into the lower valued bin, and each bin includes its upper boundary [45]. The array of bin frequencies, $Hist_{pixel}$, comprises the temporal frequency histogram.

The bin with the maximum frequency establishes the background baseline. The bins within $N$ sigma of the baseline establish the temporal frequency filter boundaries, *Filter*, such that

$$Filter = \pm N\sigma_{Baseline},$$

where $N$ is a tunable parameter that can be adjusted by the user, and $\sigma$ is the standard deviation.

4.2.3. Duration Detection Binary Matrix

The detection binary matrix, $Pixel_{Binary}(r,c,f)$, stores pixel target detections.

$$Pixel_{Binary}(r,c,f) = \begin{cases} 1, & if\ I_{Pixel}(r,c,f) > Filter \\ 0, & if\ I_{Pixel}(r,c,f) \leq Filter \end{cases},$$

where the row, $r$, column, $c$, and frame, $f$, represent the matrix indices, $I_{Pixel}$ is the pixel intensity, and a value of one indicates a detected target.

4.2.4. Duration Detection Example

A M1070 traveling at 20 m/s is inserted into a 30x30 $m^2$ pixel sampled at a rate of 120 frames/s (Figure 4.3.A). Noise is added to the signal (Figure 4.3.B) and then the array is sent to the Duration Detection algorithm.

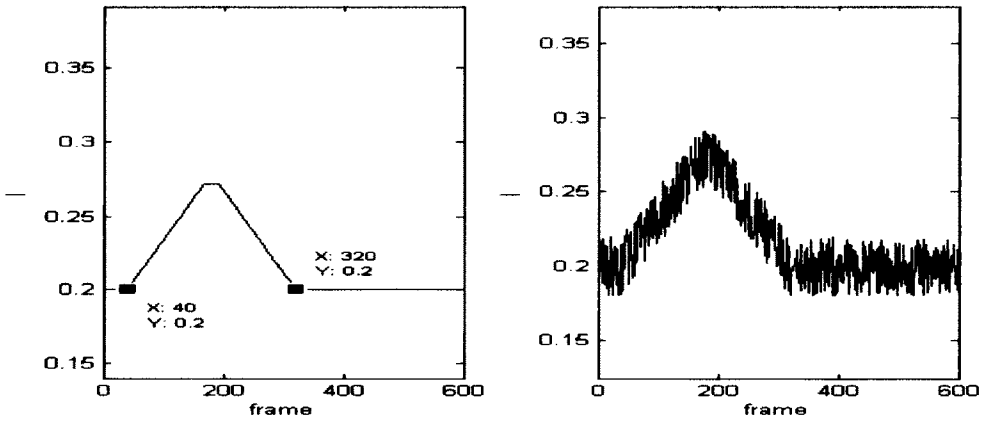



Figure 4.3 (A) M1070 profile with no noise (B) M1070 profile with 0.02 noise

The histogram plot is shown in Figure 4.4.

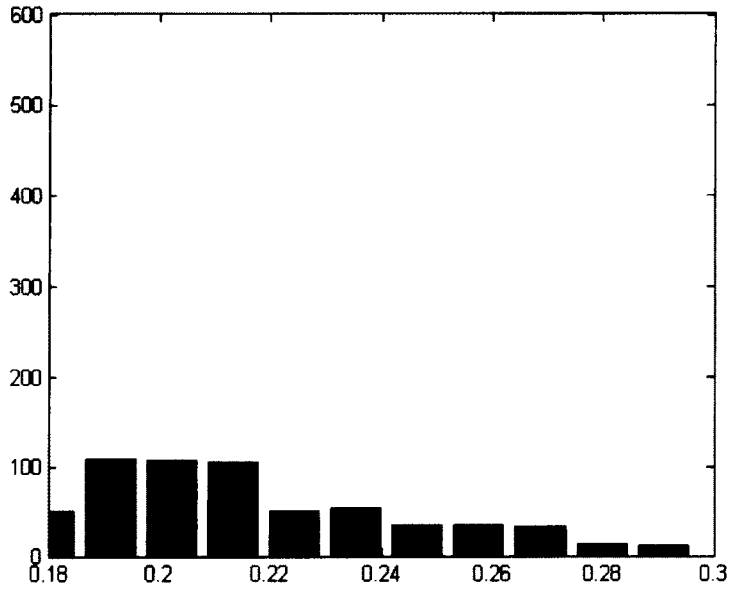



Figure 4.4 Duration Detection Histogram Plot (Intensity vs. # Frames)

The resulting Duration Binary Matrix has quite a few detections, as seen in Figure 4.5. The addition of the false alarm rate filters (Section 4.4) reduce the false alarms to zero in Figure 4.5, increasing the critical point accuracy times.

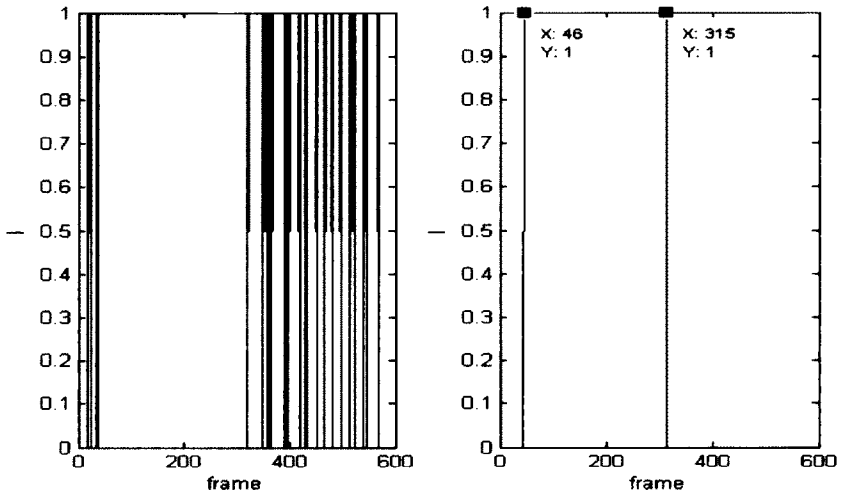



Figure 4.5 (A) Duration Binary Matrix (B) Utilizing the FAR filters to reduce false alarms

## 4.3. Fast Adaptive Power Iteration (FAPI) Background Estimation

The previously mentioned methods of change detection are extremely sensitive to jitter. The Fast Adaptive Power Iteration (FAPI) method mitigates problems in finding targets in high jitter environments, by providing robust adaptive updates to the background model [46].

Although computationally more expensive than the background averaging method or duration filtering, the increase in detection magnitude is worth the extra processing time [46]. Examining the FAPI background estimation method is required now versus earlier because the previous concepts and corresponding equations are used.

The FAPI algorithm provides estimates for the orthonormal basis vector at each iteration step. This vector is used for the adaptive background modeling. The estimates are such that the dominant subspace is spanned by the covariance matrix,

$$C_{XX}(f) = \sum_{u=-\infty}^{f_0} \beta C_{XX}(f-1) + I_{Pixel_v}(f) I_{Pixel_v}^T(f),$$

where the weighting parameter, $\beta$, is contained within [0,1] and determines the importance of each update in the estimation. It is imperative to note that the intensity pixel matrix $I_{Pixel}$ must be vectorized in an Nx1 matrix for the FAPI method. The notation $I_{Pixel_v}$ is used to identify the vectorized matrix [46] and $f$, represents the pixel frame.

### 4.3.1. FAPI Estimation

Instead of continually computing and decomposing $C_{XX}(f)$, the current basis vectors are stored in a weighting matrix, $W(f)$, which are updated at every frame. Badeau et al. developed a comprehensive method for the $W(f)$ update and the algorithm and mathematic specifics are found in the referenced publication [46].

Badeau utilizes an $R$ dimensional identity matrix for the initialization of an auxiliary matrix, $Z$, where the value of $R$ is user defined. He initializes $W$, an RxN orthonormal matrix, with an identity block sized RxR and zeros everywhere else. Simonson refined the initialization and achieves rapid and repeatable convergence using a Gram-Schmidt process given by

$$V_1 = \sum_{f=1}^{8} I_{Pixel}(f),$$

$$V_r = I_{Pixel}(r+7) - \sum_{s=1}^{r-1} \frac{I_{Pixel}(r+7)^T V_s}{V_s^T V_s} V_s, \quad r = 2, \ldots, R,$$

and

$$W_r(r+7) = \frac{V_r}{\|V_r\|}, \quad r = 1, \ldots, R.$$

where $V_1$ and $V_r$ are the basis vectors, and $W_r$ is the initial weighting matrix. $V_1$ utilizes the first $R+7$ frames for the initialization. Simonson found $R = 3$ to give the most advantage vs. computational cost [25].

The background model, $I_{Back}$, can be found with

$$I_{Back}(f) = W(f-1)W^T(f-1)I_{Pixel}(f),$$

where $f$ is the frame.

This value is used in the Threshold Change Detection formulas to check for consistency using

$$\Delta_{Pixel}(r, c, f) > T,$$

where $\Delta_{Pixel}$ is the normalized difference, $r$ is the row, $c$ is the column, $f$ is the frame, and $T$ is the threshold.

### 4.3.2. FAPI Binary Matrix

The FAPI Binary Matrix uses the same decision tools as the Threshold Detection algorithm and the Duration Detection algorithm. The detection binary matrix, $Pixel_{Binary}(r, c, f)$, stores the pixel target detections by

$$Pixel_{Binary}(r, c, f) = \begin{cases} 1, & if\ \Delta_{Pixel}(r, c, f) > T \\ 0, & if\ \Delta_{Pixel}(r, c, f) \leq T' \end{cases}$$

where the row, $r$, column, $c$, and frame, $f$, represent the matrix indices, $\Delta_{Pixel}$ is the pixel intensity, $T$ is the threshold, and a value of one indicates a detected target.

### 4.3.3. FAPI Background Suppression Example

A 50x50 pixel array of a grayscale camouflage background represents a jitter sensitive complex background with high spatial gradients. Random jitter with a three pixel maximum shift is introduced throughout a 50 frame data sequence (Figure 4.6.A). The first 10 frames are used to initialize the FAPI basis vectors using $R = 3$ and $\beta = 0.95$. After cycling through 40 additional frames, a FAPI background estimation is done on the 51st frame which is shown in Figure 4.6.B. The background estimation is subtracted from the original frame in Figure 4.6 to show the effectiveness of FAPI for background suppression (Figure 4.6.C).

Figure 4.6 (A) Original Background Image (B) FAPI Estimation (C) FAPI Background suppression

4.3.4. FAPI Change Detection

Using over 300 frames of the camo background, an M1070 (target) is introduced in a region with high spatial gradient, pixel(23,19) through pixel(23,21), and subjected to random jitter with a 0.25 pixel maximum shift. (Figure 4.7.A).

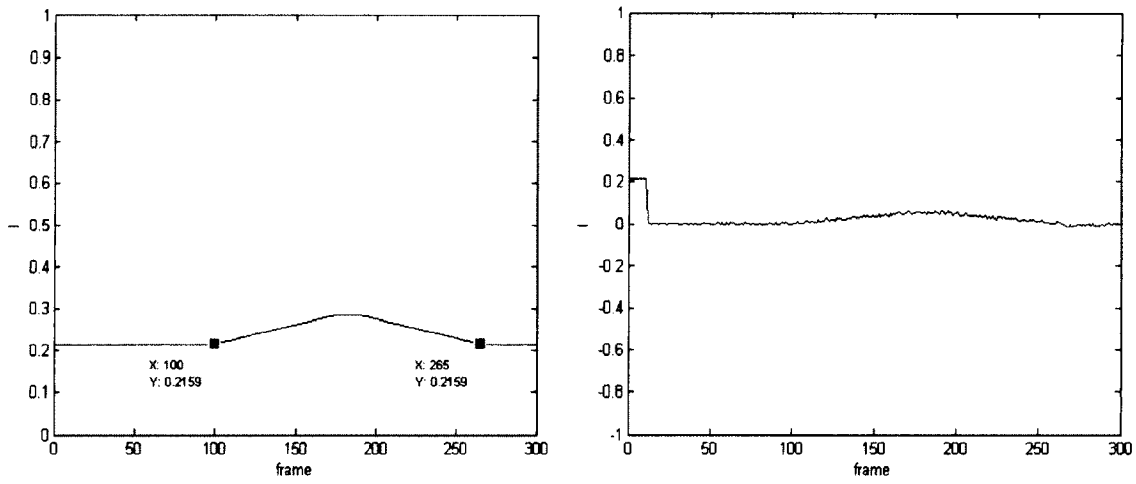



Figure 4.7 (A) M1070 modeled target (B) FAPI suppression results

The pixel array is sent to the FAPI Detection algorithm with $R = 3$ and $\beta = 0.95$. The background suppression residuals are shown in Figure 4.7. The background model is sent to the Threshold Change Detection algorithm for processing. The background model in Section 4.1.1. was used for comparison purposes (Figure 4.8.A and 4.8.B.). The FAPI background estimation method increased the normalized standard deviation estimates by more than a factor of 10 (Figure 4.8.C and 4.8.D) and significantly reduced the number of false alarms.

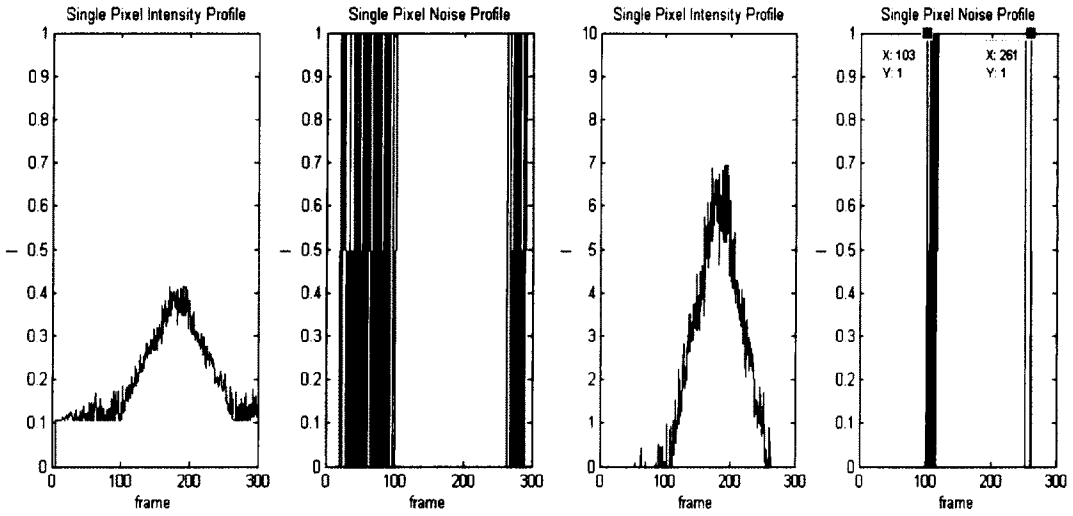



Figure 4.8 (A) Threshold Example (B) Threshold Binary Matrix (C) FAPI estimation (D) FAPI Binary Matrix

The FAR1 filter application (Section 4.4) eliminates the false alarms completely and comes within three frames accuracy for the target $Enter_1$ and $Exit_2$ points (Figure 4.9). In Figure 4.8.D, point $Enter_1$ had a value of $x$ = 100 and point $Exit_2$ had a value of $x$ = 265. That same binary matrix, with FAR1 applied, shows $Enter_1$ with a value of $x$ = 103 and $Exit_2$ with a value of $x$ = 261 (Figure 4.9). The significance and locations of points $Enter_1$ and $Exit_2$ will be explained later in the analysis section in Chapter Five.

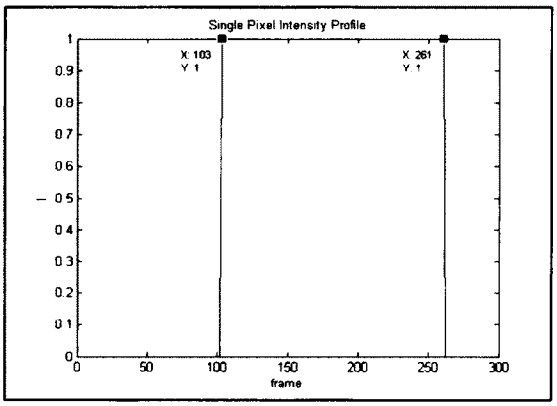



Figure 4.9 Results after applying FAR1 filter to FAPI Binary Matrix

## 4.4. False Alarm Rate Filter

A detected change in a signal does not always constitute an actual target, but may be a false alarm. False alarms can arise from bad pixels, jitter and noise that are not eliminated via proper boundaries, or a partial target signal loss in the threshold or filtering process. There is a negotiable balance between the probability of detecting the target and an acceptable False-Alarm Rate (FAR). With dim targets, a low threshold or filter is required or the target will not be detected. However, this allows a high number of false alarms to arise, which would lead to a high computational or manpower expense if all these events were processed.

When looking for targets at the sub-pixel level, it is possible to temporally differentiate between actual target detections and false alarms. As an example, it is assumed that a target transverses a single pixel over multiple frames. Therefore, any signal that lasts under a set temporal threshold can be considered a false alarm. By capitalizing on the target temporal transversal boundaries, the FAR can be significantly reduced.

Two types of FAR filter tools are used to reduce the False-Alarm Rate. The first tool responds to the situation where a single target appears as several targets and reconnects the signal. The second tool eliminates false alarms that do not meet the minimum target duration timeframe. For scenarios with both types of false alarms, it is necessary to first reconnect the split target before applying the duration threshold.

### 4.4.1. FAR1: Reconnect Disjointed Signals

Due to noise and jitter, part of the target signal is lost during the detection algorithm. As a result, a single signal might be detected as a multiple signal (Figure 4.10.A). To correct this problem, a temporal filter can be applied to the detection signal.

If spacing between each frame shows values less than the minimum number of frames between detected targets, $T_{FAR1}$, then the binary matrix value is set to 1, thereby reconstituting the original signal (Figure 4.10.B). $T_{FAR1}$ is user defined based on the sensor and detection algorithm chosen.

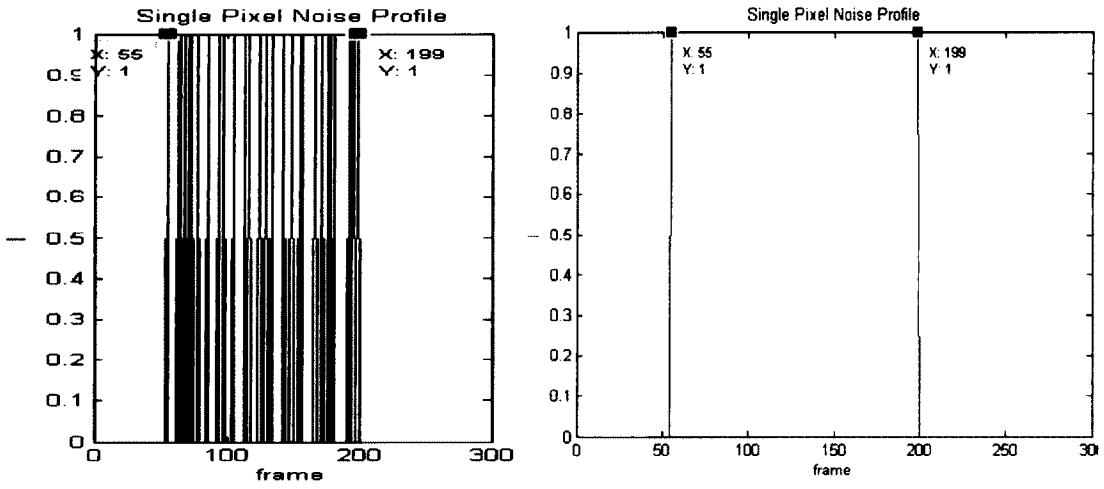



Figure 4.10 (A) Threshold Binary Matrix (B) Application of FAR1 filter to recapture critical target points

## 4.4.2. FAR2: Minimum Threshold Requirement

If the detection threshold or filters are too low or if bad pixels exist, the algorithm will classify non-target frames as targets. However, these false alarms are detected over a minimum number of consecutive frames. FAR2 establishes that a detection is only considered an actual target if the number of consecutive frame detections meet the minimum threshold requirement, $T_{FAR2}$. Therefore, if the number of frames of the detected target is less than $T_{FAR2}$, it is a false alarm.

Figure 4.11 is target profile of a M1070 travelling at 20 m/s through a 30x30 $m^2$ pixel frame. Figure 4.11 are the noise and standard deviation plots, respectively. Figure 4.11 is the result from the Threshold Change Detection algorithm.

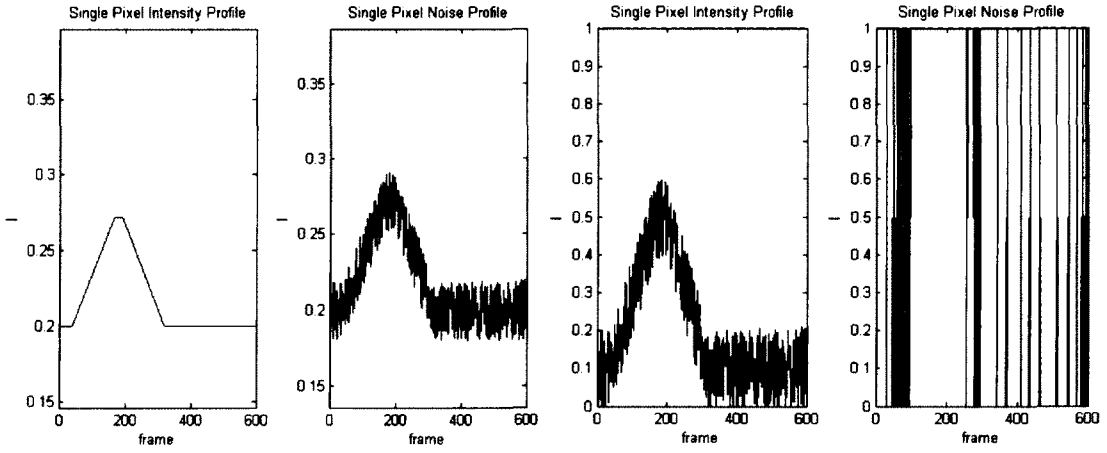



Figure 4.11 (A) Initial target (B) Target with noise (C) Threshold Change Detection (D) Threshold Binary Matrix

The FAR1 filter is used first, followed by the FAR2 filter, resulting in a complete reduction of the false alarms (Figure 4.12). $T_{FAR2}$ is user defined and based on background knowledge of the target.

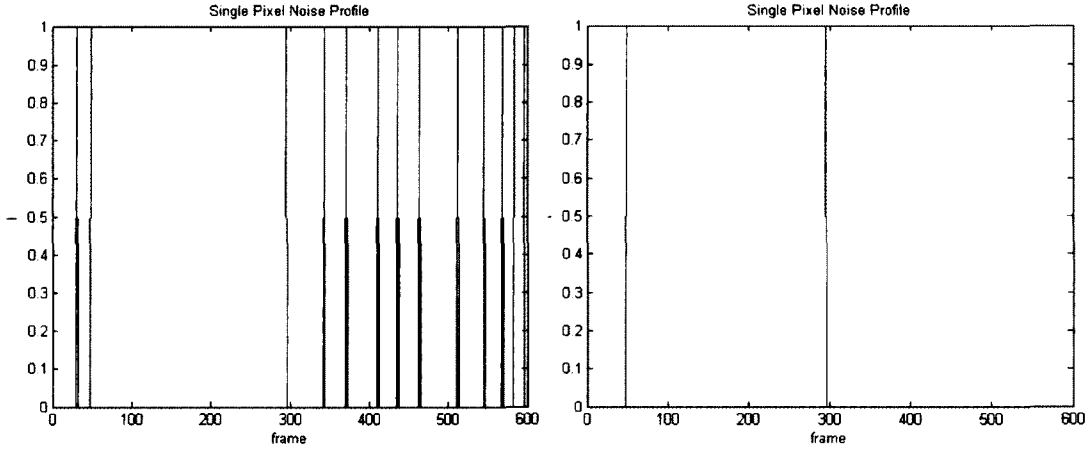



Figure 4.12 (A) FAR1 Results (B) FAR2 Results

# CHAPTER FIVE

# TARGET EXTRACTION AND ANALYSIS

Current target extraction and analysis methods are geared toward multi-pixel targets. Analysis techniques are spatial in nature and use feature and texture criterion to train the algorithms and create extensive databases for target comparison and classification.

Choi and Mei use a method of whole template matching to find feature similarities between the detected target and an existing database. Choi uses scale-invariant features transform (SIFT) algorithm to recognize a target and extract rotation-invariant features from the image [47]. The features are matched with a database for target classification. Prior training and sample tests are needed to create an indexed database. Mei first finds a target bounding box via image Euclidean distance and image weighted distance, then does a template matching comparison against a stored database [12].

Instead of whole template matching, Nair uses a classification of the target by parts and creates a very complex hierarchical modular level Bayesian object recognition system. The recognition is based on the whole static image and is spatially processed [48]. Vehicles are classified into 3 types; tanks, trucks and armored personnel carriers. The levels of recognition are tiered into low-level, which does basic edge detection and image segmentation, mid-level, which represents and describes the pattern shape and

extraction features, and the higher level, which provides visible pattern category assignment.

Manikandan does an evaluation of edge detection spatial processing for automatic target recognition, which goes on the basis that the edge of an object will have a higher intensity gradient than the object itself and the background [49]. The different edge techniques evaluated were Prewitt, Sobel, Roberts, Canny (with and without thresholding) and LoG (with and without thresholding). The simulation results showed a 95-97% detection rate for IR images. Clarkson's target recognition is very complex and requires the storage of a model database which he indexed as a single hierarchically constructed model tree using a neural network. The edge detected target is compared to the list of available target databases for recognition. [50]

Gupte uses a novel method which implements a GUI spatial process to calculate vehicle parameters [51]. The vehicles' dimensions are used to categorize vehicles into two categories, cars and non cars. This process requires graphical manipulation by the end user and the calculations are based on the user's ability to correctly surround the region of interest.

These methods use groups of pixels to identify the target. As the target decreases in size, these methods lose their effectiveness. The sub-pixel algorithm in this dissertation focuses on each individual pixel and processes them independently for target detection. This independent focus allows for sub-pixel level discovery. In addition, the temporal method of target analysis does not require an extensive database or target training to extract target dimension characteristics.

## 5.1. Extract Target

As briefly discussed earlier, each change detection algorithm produces a Target Binary Matrix, which captures all detected targets held within a frame sequence. For each target detected, the beginning frame, $Enter_1$, and the last frame, $Exit_2$, of the target sequence are recorded as target counters. The counters are critical points as they are integral in finding the target characteristics. The counters and the original frame sequence are forwarded to the target analysis algorithm for processing.

### 5.1.1. Considerations

The FAR filters are designed to temporally separate detections and false alarms in order to reduce the false alarm rate. However, in certain scenarios, false alarms are not completely eliminated, or split targets are not rejoined. Since the extraction algorithm does not discriminate between an actual target and a false alarm, the entire frame sequence, not just the target, is available for further differentiation in the analysis section.

One major issue in target extraction is the complete capture of the detected signal. Based upon the amount of noise and jitter, partial loss of the target frame length is expected when building the Target Binary Matrix. The methods used in the target analysis section address this issue.

### 5.1.2. Extraction Example

This modeled example has the considerations and assumptions stated in previous sections, where a M1070 traveling at 20 m/s is the target with a noise value of $\sigma = 0.02$ applied to the signal. The signal undergoes all the filtering mechanisms also stated in the previous sections to yield values for points $Enter_1$ and $Exit_2$ of 60 and 270, respectively (Figure 5.1).

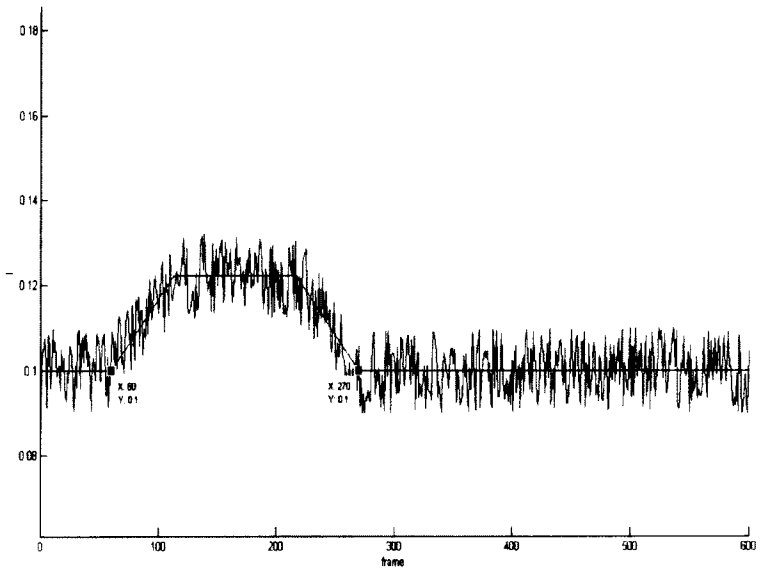



Figure 5.1 M1070 signal with noise profile and critical entry and exit points marked

The extraction algorithm records $Enter_1$ = 66 and $Exit_2$ = 255, which have % errors of 10% and 5.6% respectively, with a transit time % error of 11.1% (Figure 5.2). These critical points along with the original signal are forwarded to the target analysis algorithm for processing.

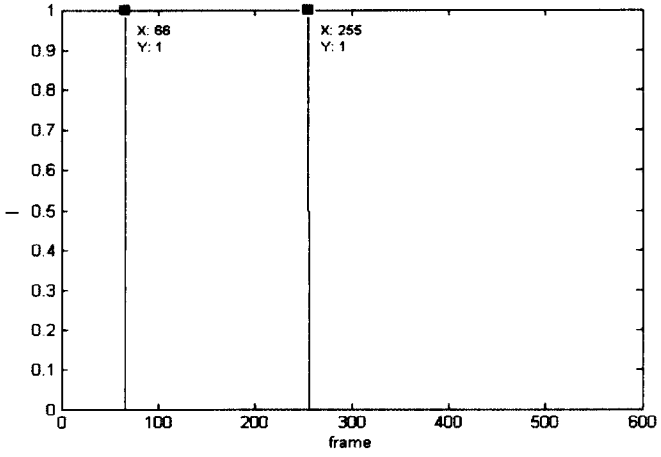



Figure 5.2 Target Binary Matrix with Extraction Points

## 5.2. Target Analysis

Once detection occurs, the target is extracted from the frame sequence and analyzed. The target analysis is temporal in nature and based on the expected target model profile, resulting in the approximated sub-pixel target characteristics of velocity and length.

### 5.2.1. Assumption Review

The target is going in a horizontal linear direction with a constant velocity during the frame sequence. The ground terrain slope is negligible and there is a single target in transit. Additional assumptions are that the pixels sizes are known, are uniform in size and have a rectangular shape.

### 5.2.2. Calculating Target Velocity and Length

When the target is smaller than a pixel and the pixel size is known, target information, such as velocity and object length can be found using the linear temporal signal and the pixel's geometric configuration.

Given the critical target points, $Enter_1$, $Enter_2$, $Exit_1$, and $Exit_2$,

$$f_{entry} = Enter_2 - Enter_1 = Exit_2 - Exit_1,$$

where $f_{entry}$ is the number of frames it takes for the target to fully enter the pixel. In an ideal situation, $f_{enty}$ and $f_{exit}$ should be the same value. However, differences do arise due to noise, jitter, and other effects. To accommodate these differences, $f_{entry}$ is calculated by averaging the entry and exit frames, using

$$f_{entry} = \frac{(Enter_2 - Enter_1) + (Exit_2 - Exit_1)}{2},$$

The number of frames it takes for the target to transit the pixel is found by

$$f_{transit} = Exit_2 - Enter_1,$$

and the velocity of the target is found by

$$V_{target} = \frac{x_{pixel}}{(f_{transit} - f_{entry})},$$

where $V_{target}$ is the target velocity in m/frames.

Due to the horizontal motion cons$L_{target} = V_{target} f_{entry}$,ength can be found by

$$L_{target} = V_{target} f_{entry},$$

where the length of the target, $L_{target}$, is in meters. If the trajectory constraint is removed, information concerning the target's width and area may be found. This concept is discussed in detail in Chapter Seven.

5.2.3. FermiFit Analysis

The relationship between the trapezoid model and the Fermi derivative is explained in Chapter Three for the purpose of target modeling. Capitalizing on this relationship, if the Fermi derivative can be fitted to a target's signal data, the critical points can be recovered.

*5.2.3.1. Finding the Fermi Derivative Parameters*

From Eq. 3.1, the Fermi derivative is found with the equation:

$$f_{df}(k; a, b, c, d) = \frac{ae^{\frac{k-b}{c}}}{(e^{\frac{k-b}{c}} + 1)^2} + d,$$

where $k$ is the frame number and $a$, $b$, $c$, and $d$ are constants.

5.2.3.1.1. Parameter d

As $k$ approaches infinity, the pixel intensity reduces to $d$,

$$\lim_{k\to\infty} \frac{ae^{\frac{k-b}{c}}}{(e^{\frac{k-b}{c}}+1)^2} + d = d = I_{Back},$$

which is the background intensity model found by the change detection algorithms.

5.2.3.1.2. Parameter b

The detection algorithm discovers the frame range where the temporal profile deviates from the background, where the first frame in this range is the frame of target entry into the pixel, $Enter_1$, and the last frame in the range is the frame of target exit from the pixel, $Exit_2$. These two values comprise the target transit frames given by

$$f_{transit} = Exit_2 - Enter_1.$$

The $b$ constant correlates to the midpoint frame of the target signature using

$$b = \frac{f_{transit}}{2} + Enter_1.$$

5.2.3.1.3. Parameter a

At the midpoint frame ($k = b$), the equation reduces to:

$$f_{df}(k = b; a, b, c, d) = \frac{a}{2^2} + d.$$

When $k = b$, the pixel intensity is the intensity value of the target fully within the pixel frame as shown in the trapezoidal model in Section 3.4.1. It is now possible to estimate the value of $a$ given by

$$a = 4(f_{df}(k = b) - d).$$

5.2.3.1.4. Parameter c

To find parameter $c$, an additional data point is required where $k \neq b$. Since $c$ represents the width of the Fermi derivative function, the data point is chosen from the target entry or exit slope ($k_{slope}$, $f_{dfslope}$). Using this data point in the Fermi derivative model results in

$$f_{dfslope} = \frac{ae^{\frac{k_{slope}-b}{c}}}{(e^{\frac{k_{slope}-b}{c}}+1)^2} + d.$$

Let

$$z = e^{\frac{(k_{slope}-b)}{c}}.$$

Substitute $z$ into the Fermi derivative model,

$$f_{dfslope} = \frac{az}{(z+1)^2} + d,$$

rearrange into quadratic form,

$$z^2 + \left(2 - \frac{a}{(f_{dfslope}-d)}\right)z + 1 = 0,$$

and solve for $z$,

$$z = -1 + \frac{1 \pm \sqrt{1 - 4\frac{(f_{dfslope}-d)}{a}}}{\frac{2(f_{dfslope}-d)}{a}}.$$

Combining the $z$ equations results in

$$e^{\frac{(k_{slope}-b)}{c}} = -1 + \frac{1 \pm \sqrt{1 - 4\frac{(f_{dfslope}-d)}{a}}}{\frac{2(f_{dfslope}-d)}{a}},$$

and

$$\frac{(k_{slope}-b)}{c}=\ln\left(-1+\frac{1\pm\sqrt{1-4\frac{(f_{dfslope}-d)}{a}}}{\frac{2(f_{dfslope}-d)}{a}}\right).$$

Finally, solve for $c$ using

$$c=\frac{(k_{slope}-b)}{\ln\left(-1+\frac{1\pm\sqrt{1-4\frac{(f_{dfslope}-d)}{a}}}{\frac{2(f_{dfslope}-d)}{a}}\right)},$$

and substitute in the parameters $a$, $b$, $c$ and $d$ to complete the Fermi derivative model.

The $c$ parameter has two solutions, such that

$$c_+=-c_-,$$

where $c_+$, is the solution using the positive sign and $c_-$ is the solution using the negative sign in the $c$ parameter equation. If $k_{slope}$ is chosen on the target's exit slope and $b$ is the midpoint, $k_{slope}-b$ is positive and $c_+$ is used. If $k_{slope}$ is chosen on the target's entry slope, $k_{slope}-b$ is negative and $c_-$ is used.

*5.2.3.2. Recovering Critical Points*

In Chapter Four, the critical points, $Enter_1$ and $Exit_2$ were shown to be found via the change detection algorithm. The other critical points, $Enter_2$ and $Exit_1$ are found using the parameters $b$ and $c$. Using $b$ as the midpoint of the trapezoid and $Nc$ as the time the target is fully in the pixel, $Enter_2$ is the smallest value given by

$$Enter_2=b-\frac{Nc}{2},$$

and $Exit_1$ is the largest value given by

$$Exit_1=b+\frac{Nc}{2},$$

where $N$ is a tunable parameter between one and two.

*5.2.3.3. FermiFit Example*

A HEMTT target model is inserted into a single pixel, travelling at 20 m/s and sampled at 120 frames/s (Figure 5.3.A). A noise value of $\sigma = 0.01$ is added to the pixel signal before sending the data to the Threshold Detection algorithm and the FAR filters (Figure 5.3.B).

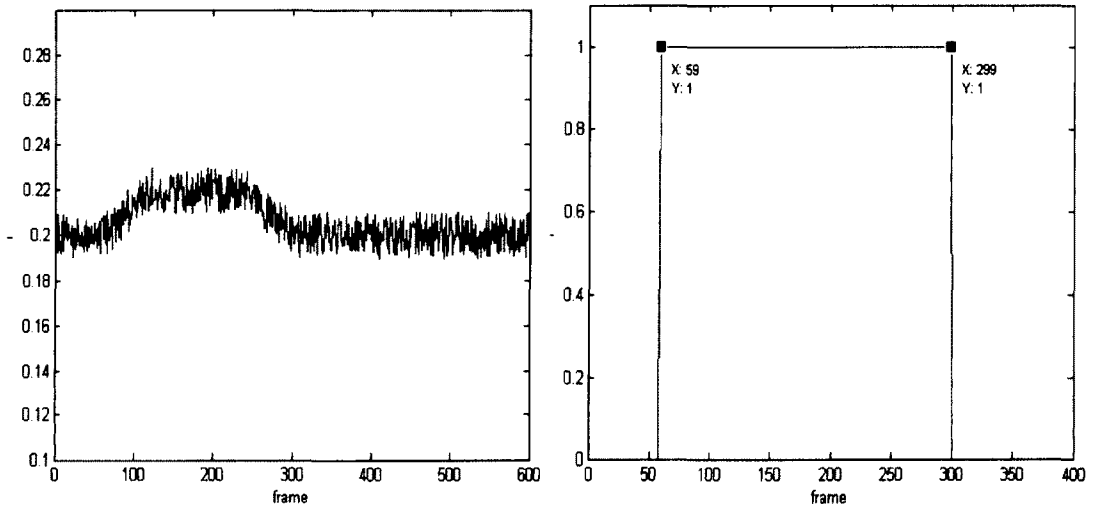



Figure 5.3 (A) HEMTT Target Signal (B) Threshold Binary Matrix

From the detection portion, two critical points are found where $Enter_1 = 59$ and $Exit_2 = 299$. From the FermiFit analysis (Figure 5.4), the calculated Fermi derivative parameters were found to be $a = 0.6789$, $b = 179.5$, $c = 35.2447$ and $d = 0.1104$. The tunable parameter $N$, was set to 1.7.

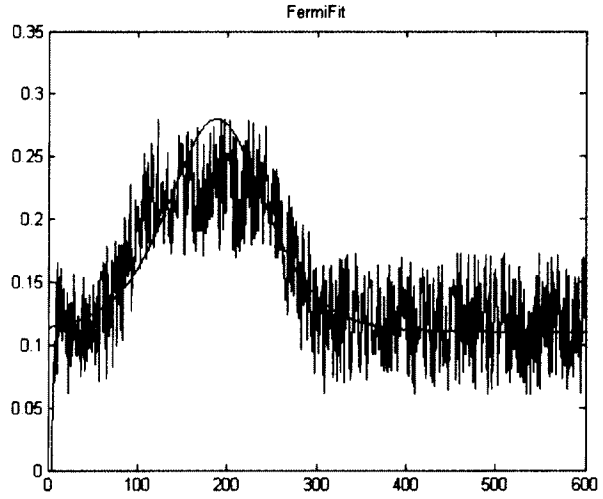



Figure 5.4 FermiFit approximation superimposed over the HEMTT target signal + noise

The two remaining critical points were then calculated as $Enter_2$ = 120 and $Exit_1$ = 239. The target velocity and length are calculated to be 19.95 m/s (*% error* = 0.25) and 10.07 m (*% error* = 1.18), respectively. The original signal is found in Figure 5.5 and has the critical points marked for comparison purposes.

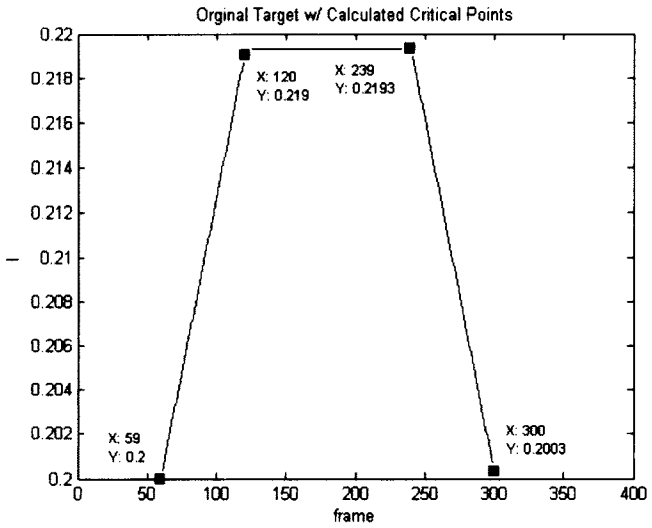



Figure 5.5 Modeled HUMVEE target with critical points marked

### 5.2.4. PolyFit Analysis

The purpose of this method is to find an approximation of the trapezoid profile. The critical points are located at the extreme points of trapezoid's intersections. While a single approximation function may get the general shape, the function will not recover the intersections with any accuracy. The next section explains how using the least squares approximation models the original signal in the form of a polynomial. Utilizing two approximating polynomials, a sixth and fourth degree, the six different intersections from these functions can be used to estimate the original target. This is done by using the six intersections to create the three lines forming the trapezoid as in Figure 5.6. The trapezoid approximation uncovers the four critical points, $Enter_1$, $Enter_2$, $Exit_1$ and $Exit_2$.

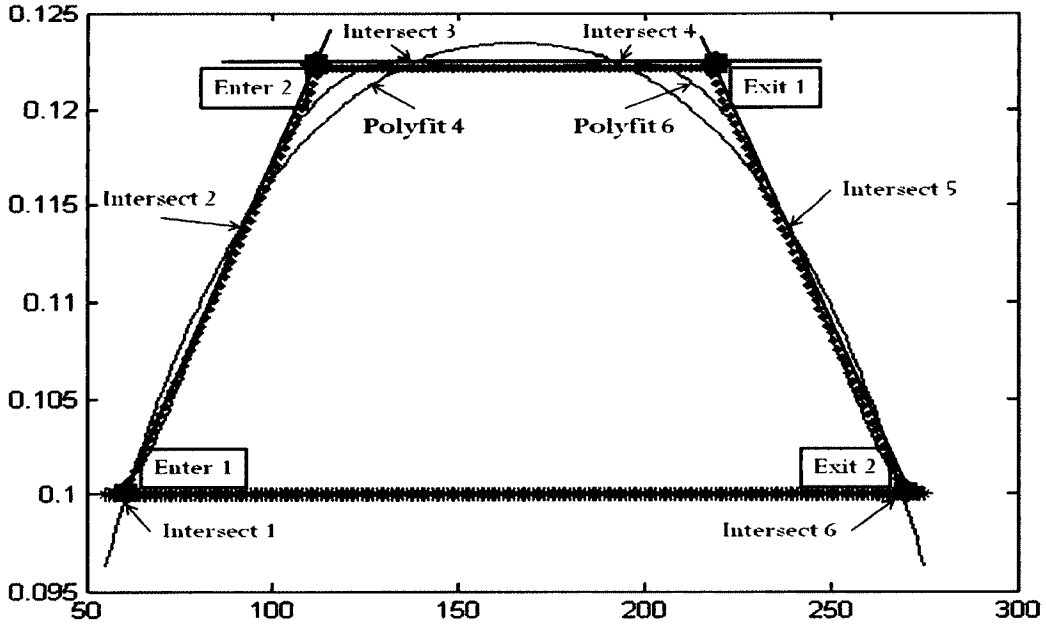



Figure 5.6 A trapezoid approximation using the 6 intersection points to uncover the 4 critical points

*5.2.4.1.* *Least Squares Approximation*

A signal is a sequence of discrete data points measured by the sensor. The Least Squares Approximation finds an estimated fit of the signal data via an approximating function that passes through the set of points in the best possible manner without requiring the function to pass through each point.

Given $N$ data points, $N(x_i, y_i)$, the goal is to find the best fit to the $n^{th}$ degree polynomial approximating function,

$$P_n(x) = a_1x^n + a_2x^{n-1} + \cdots + a_nx + a_{n+1},$$

where $a_1, \ldots, a_n$ are coefficients.

The sum of the squares of the deviations, $S$, are found by

$$S(a_1, a_2, \ldots, a_{n+1}) = \sum_{i=1}^{N} (y_i - a_{n+1} - a_n - \cdots - a_1x_i^n)^2$$

and $S$ is at a minimum when the partial derivatives are equal to zero as shown by

$$\frac{\partial S}{\partial a_0} = \sum_{i=1}^{N} 2(y_i - a_{n+1} - a_n - \cdots - a_1x_i^n)(-1) = 0,$$

$$\vdots$$

$$\frac{\partial S}{\partial a_n} = \sum_{i=1}^{N} 2(y_i - a_{n+1} - a_n - \cdots - a_1x_i^n)(-x_i^n) = 0,$$

and

$$\frac{\partial S}{\partial a_{n+1}} = \sum_{i=1}^{N} 2(y_i - a_{n+1} - a_n - \cdots - a_1x_i^n)(-x_i^n) = 0.$$

Dividing the partial derivatives by two yields the normal equations,

$$\sum_{i=1}^{N} y_i = a_1 \sum_{i=1}^{N} x_i^n + \cdots + a_n \sum_{i=1}^{N} x_i + a_{n+1} N,$$

$$\vdots$$

and

$$\sum_{i=1}^{N} x_i^n y_i = a_1 \sum_{i=1}^{N} x_i^{2n} + \cdots + a_n \sum_{i=1}^{N} {x_i}^{n+1} + a_{n+1} \sum_{i=1}^{N} {x_i}^n,$$

that can be solved for the coefficients $a_1, \ldots, a_n$ by Gaussian elimination [52].

The resulting solutions for the $n^{th}$ degree polynomials, $Poly_n$, are stored in the matrix form

$$Poly_n(r, c, :) = [a_1 \quad . \quad . \quad . \quad a_n \quad a_{n+1}],$$

where the $r$ is the row index and $c$ is the column index.

This method can encounter problems for higher $N$ values because the coefficients can vary over a range of several orders of magnitude. Normalizing each equation and using double precision variables help prevent the ill-conditioned system problem. However, it is not recommended to have $N$ values over six [52].

*5.2.4.2.* *Trapezoid Approximation*

The target's critical points can be found by the linear geometry relationships between $PolyValue_6$, $PolyValue_4$ and the estimated background.

The first step is to approximate the trapezoid. First, set

$$Poly_6 = Poly_4,$$

and find the zeros

$$PolyZeros_6(r, c, :) = [z_{x1} \quad z_{x2} \quad z_{x3} \quad z_{x4} \quad z_{x5} \quad z_{x6}],$$

where $z_1, z_2, \ldots, z_6$ are the $x$ or frame values where $Poly_6$ and $Poly_4$ intersect.

Next, solve for the corresponding $y$ or intensity values for each $x$ to gather the trapezoid's six intersection points given by

$$P_6(z_{x1}) = z_{y1},$$

$$\vdots$$

and

$$P_6(z_{x6}) = z_{y6}.$$

From these points, three lines are created such that

$$Line\ 1(x) = z_{y1} + \frac{(z_{y2} - z_{y1})}{(z_{x2} - z_{x1})}(x - z_{x1}),$$

$$Line\ 2(x) = z_{y3} + \frac{(z_{y4} - z_{y3})}{(z_{x4} - z_{x3})}(x - z_{x3}),$$

and

$$Line\ 3(x) = z_{y5} + \frac{(z_{y6} - z_{y5})}{(z_{x6} - z_{x5})}(x - z_{x5}).$$

The fourth line is created from the background model,

$$Line\ 4(x) = I_{Back}.$$

The intersections of the lines create the four critical points,

$$Enter_1 = \text{zeros}(Line1 - Line4),$$

$$Enter_2 = \text{zeros}(Line2 - Line1),$$

$$Exit_1 = \text{zeros}(Line3 - Line2),$$

and

$$Exit_2 = \text{zeros}(Line4 - Line3),$$

where zeros is a MATLAB function that finds the x-intercept.

*5.2.4.3.* *PolyFit Example*

A HEMTT target ($L$ = 10.19 m) traveling at 23 m/s is inserted into a 30x30 $m^2$ single-pixel with noise ($\sigma$ = 0.02) added (Figure 5.7.A., 5.7.B). After running the change threshold detection algorithm, the binary matrix is sent to the PolyFit subroutine.

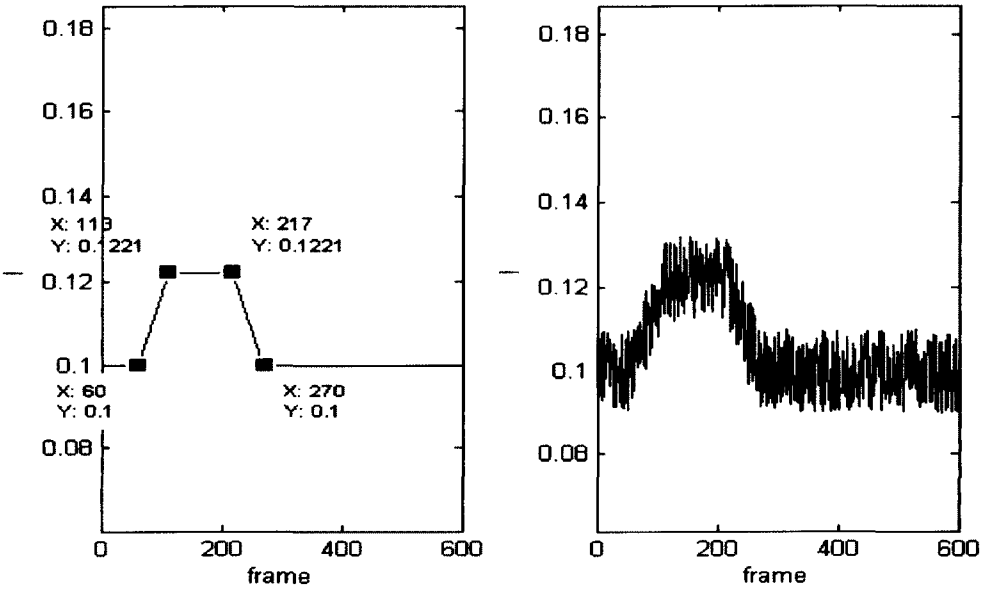



Figure 5.7 (A) Modeled target with critical points marked (B) Target profile with noise

The PolyFit algorithm finds the fourth and sixth degree polynomial fits and calculates the critical points. Figure 5.8 shows the original signal, $Poly_6$, and $Poly_4$. Lines are drawn to show the critical point intersections.

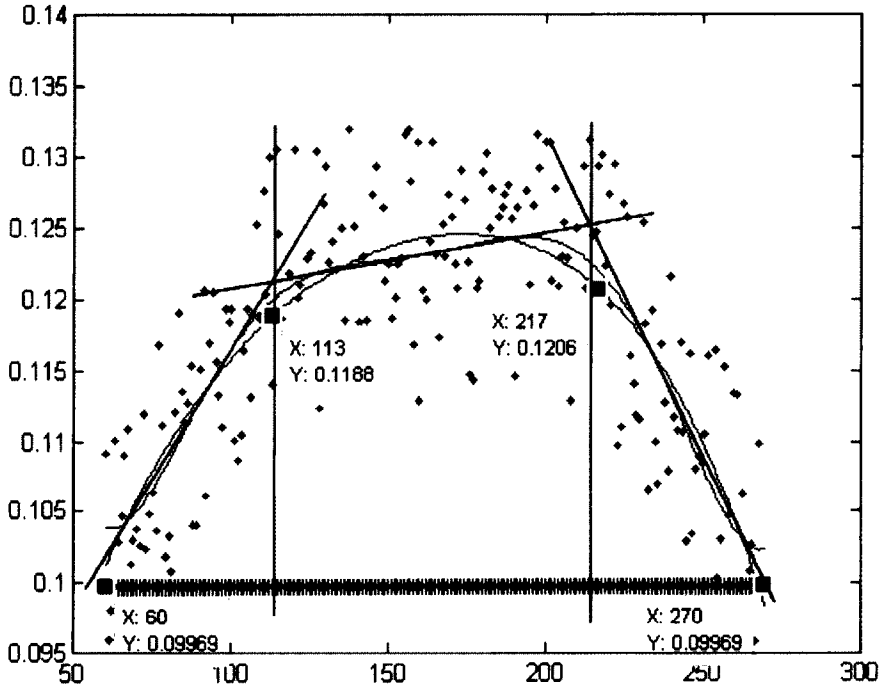



Figure 5.8 PolyFit analysis with original critical points marked

The calculated critical points are $Enter_1$ = 55, $Enter_2$ = 113, $Exit_1$ = 216, and $Exit_2$ = 270. The calculated velocity and target length are 22.93 m/s and 10.13 m, respectively. Comparing the calculated values with true values, $Enter_2$ and $Exit_2$ were exact. $Enter_1$ and $Exit_1$ had a five or less frame difference. From Chapter Three, a HEMTT vehicle is 10.19 m long, showing differences of only 0.06 m (error of 0.5%) and a velocity delta of 0.07 m/s (error of 0.3%).

# CHAPTER SIX

# SUPER PIXEL REFINEMENT

During target extraction significant information can be missed during single-pixel analysis when a target is split amongst several pixels, with consequential inaccuracies in the target characteristics. By reconnecting the split pixels, the entire target profile is captured. When using actual sensor data, two more tools become necessary to refine the signal enough to derive critical information. At certain times, the target does not travel completely through a single pixel horizontally. Instead the target may travel through part of two different pixels. At other times, the target travels at such a speed to give an insufficient temporal profile. Yet another potential problem exists when the target may be just a bit bigger than a pixel, but not big enough to distinguish its characteristics by using conventional spatial methods.

Therefore, instead of super-resolution, this section investigates super-pixel refinement. The two tools to accomplish this refinement are the Sum-pixel and Enlarge-pixel methods. The balance is to ensure the signal is not lost in this process.

The figures in Figure 6.1 represents sensor data taken from a video clip of a remote control vehicle moving horizontally through the screen on a grass and sand background. In the first figure (set of four graphs), the target is split amongst the four pixels [53]. The second figure (larger single graph) uses the sum pixel process to

capture the entire target. The critical target entry and exit points are now well defined (Figure 6.1).

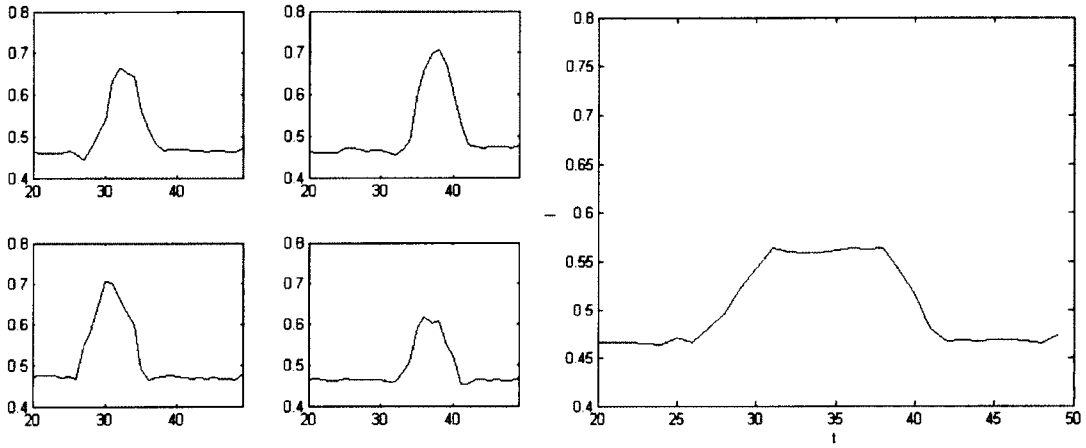


Figure 6.1 (A) 2x2 pixel array showing temporal profile of same target (B) Sum-pixel result of same target showing trapezoid profile

### 6.1. Sum Pixel Refinement

Sum-pixel refinement creates a better understanding of the detected target by merging neighboring pixels in order to gather more temporal target data within a pixel. The benefits of this process are longer target profiles and some reduction in noise and jitter. This method generally exacts a price of a reduced target to pixel ratio and lower target to background intensities, so care must be taken not to lose the target signal.

Figure 6.2 shows a 4x4 array with a modeled target signal inserted into a non uniform background with a jitter value of 0.25 pixel and a noise value of 1.5%. The signal is split between the second and third rows of the array simulating a non centered target.

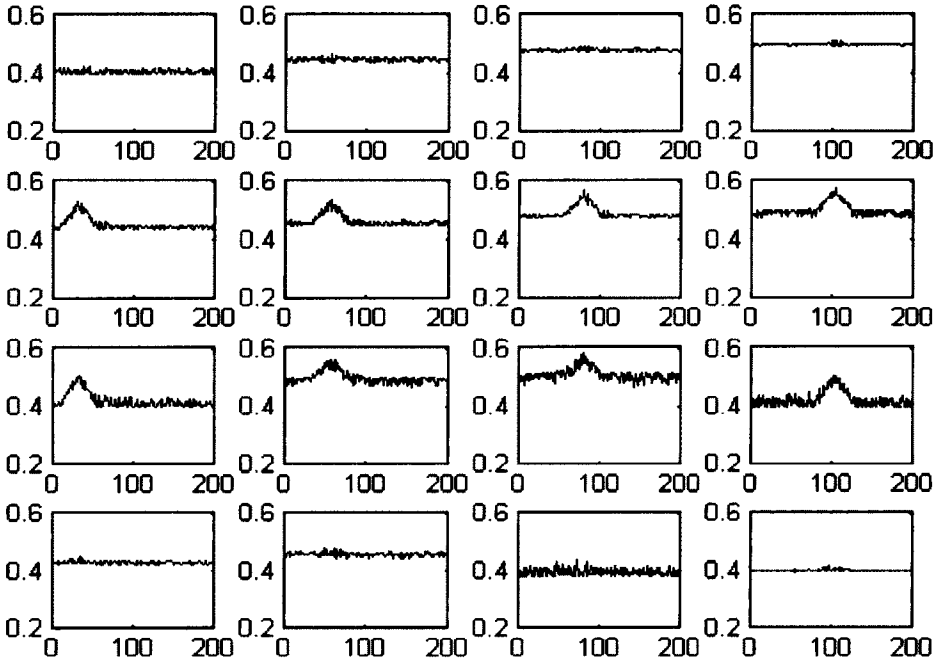


Figure 6.2 Multi-pixel target split among two rows

Figure 6.3.A is a close up view of $Pixel_{(2,1)}$ from Figure 6.2. Notice that the target signal has a very short time peak where the target is fully within the frame, making it harder to get the correct velocity. Figure 6.3.B is the summed pixel, $Pixel_{(2:3,1)}$, which is the result of summing rows two and three from the array in Figure 6.2.

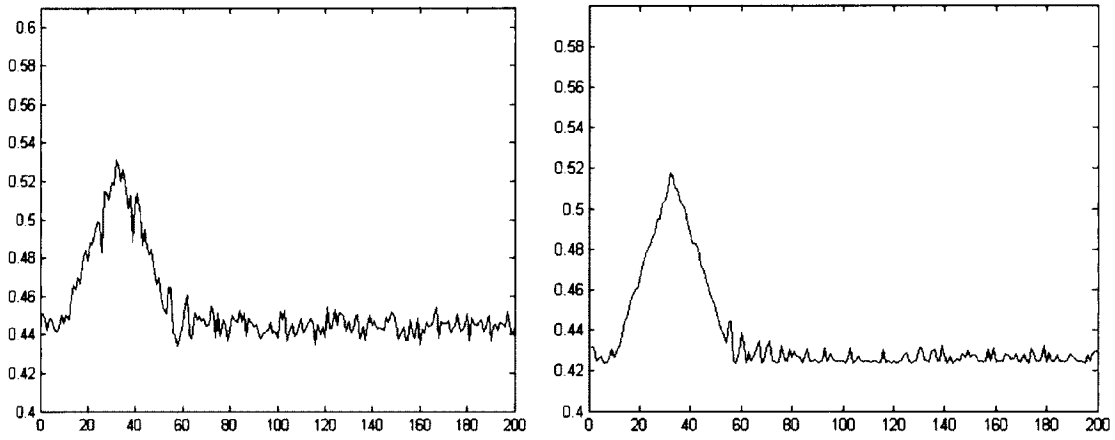


Figure 6.3 (A) Close up view of $Pixel_{(2,1)}$ (B) Result of summing pixels in rows 2 and 3

Figure 6.4.A and 6.4.B are the summed pixel, $Pixel_{(2:3,1:2)}$, and the summed pixel, $Pixel_{(2:3,1:4)}$, respectively.

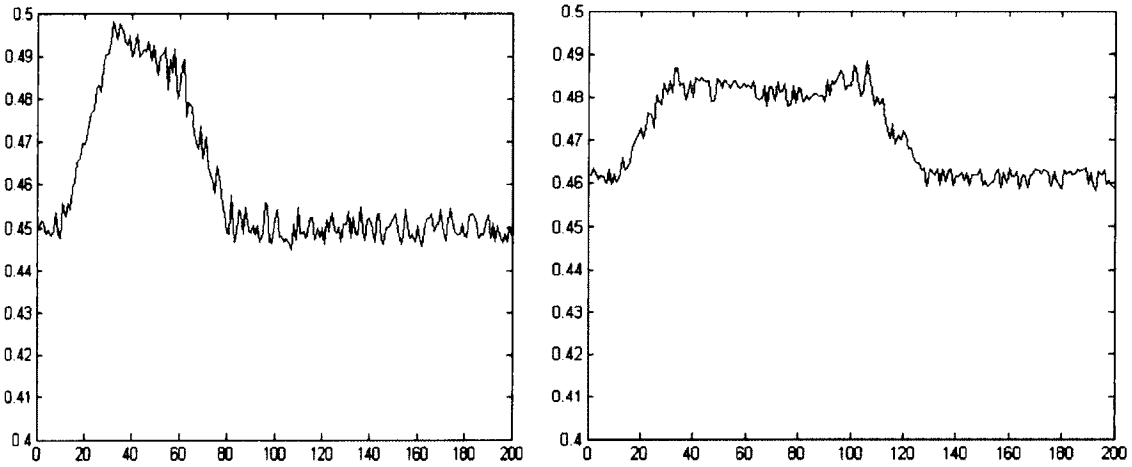



Figure 6.4 (A) Result of summing rows 2 and 3 with columns 1 and 2 (B) Result from summing rows 2 and 3 with columns 1-4

As the pixels are summed in the direction of travel, i.e. the columns, there are more quality frames for target analysis. In addition, summing the two rows of the split targets reduced the amount of noise and jitter present.

## 6.2. Enlarge Pixel Refinement

Enlarging the pixel is especially useful for targets with a high ratio of target to pixel (RTP), in which case the target area encompasses the majority of the pixel. These targets have high detection probabilities, but cause problems within the target characterization algorithms. In these scenarios, the duration of the target remaining completely within the pixel is minimal, creating a single point peak, instead of the desired trapezoid profile. By enlarging the pixel along the target trajectory, the temporal profile is expanded, recovering the trapezoid. This tool has an advantage over the Sum

Pixel process in that it can expand partial pixels and can focus on narrower regions of interest (Figure 6.5.A and 6.5.B).

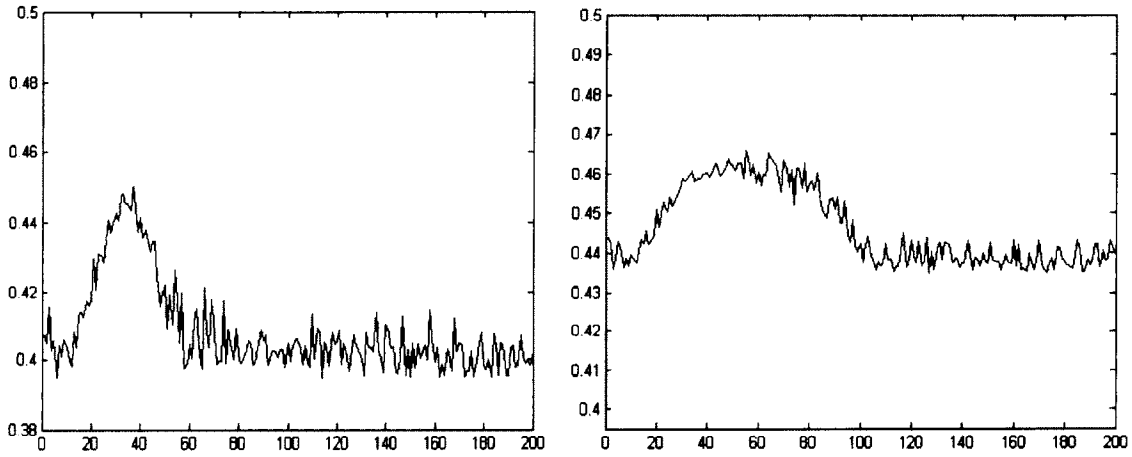



Figure 6.5 (A) Single-pixel target with 0.25 jitter and 1.25% noise (B) Result from enlarging pixel by a factor of 0.8

### 6.3. Binary Target Matrix

The Sum Pixel process is considered a refinement step because prior target detection knowledge is integral in selecting regions of interest to sum or enlarge. The task of choosing the appropriate pixel increase factor is made easier via the Binary Target Matrix produced in the change detection algorithms. The Binary Target Matrix shows the temporal target neighbors and serves as the Sum Pixel refinement starting point (Figure 6.6).

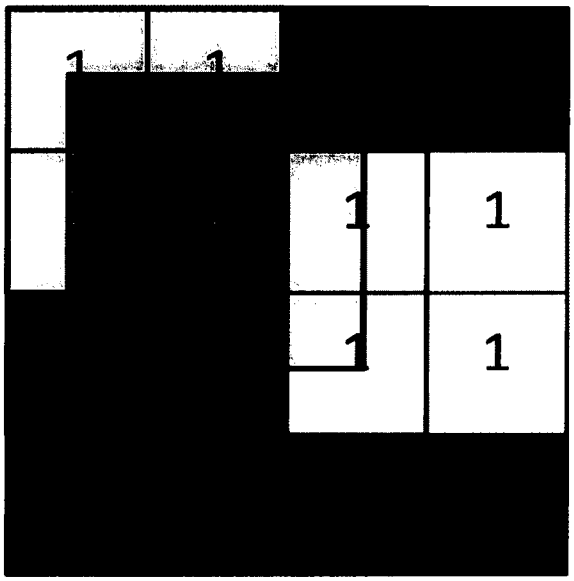



Figure 6.6 Pictoral description of temporal target binary matrix

Using the binary matrix in Figure 6.6, a good input for the Sum-pixel algorithm would be the highlighted sub matrix, $Pixel_{(1:2,1:2)}$ and a good input for the Enlarge-pixel algorithm would be the highlighted partial pixel box, $Pixel_{(0.5:2.5,0.5:2.5)}$, using the expansion factors of $N_{row} = 0.5$ and $N_{col} = 0.5$.

Super-pixels do not have to be equal in the row and column direction. Different options for choosing the best increase factor include resizing the entire matrix or a sub-matrix by an integer multiple, or expanding a single pixel to include a target's temporal neighbors. The best choices involve using a minimum number of pixels with a zero value. Currently, the summation and enlargement factors are chosen manually as a post-processing step, but future work would include automating this process.

## 6.4. Refinement Algorithms and Tools

The two tools for super-pixel processing are the Sum-pixel algorithm, which can resize the frame array into a smaller number of pixels and the Enlarge-pixel algorithm, which enlarges a single pixel to focus onto a region of interest and give a larger viewpoint.

### 6.4.1. Sum Pixel Algorithm

In this algorithm, the frames are resized to accommodate the new pixel size and the intensities of the old pixels are averaged to create the new pixel intensity.

The Sum-pixel algorithm resizes the *mxnxf* array to a *pxqxf* array:

1) Calculate the combination factor for the rows and columns given by

$$F_{row} = \frac{m}{p},$$

and

$$F_{col} = \frac{n}{q},$$

where $F_{row}$ and $F_{col}$ are the number of old rows and old columns that will make up a new pixel size. As of this time, $F_{row}$ and $F_{col}$ need to be an integer value.

2) Calculate the new pixels for each frame of the new matrix given by

$$\sum_{f=1}^{NumFrames} \sum_{c_n=1}^{q} \sum_{r_n=1}^{p} Pixel_{new}(r_n, c_n, f) = \sum_{c_o=col_{start}}^{col_{stop}} \sum_{r_o=row_{start}}^{row_{stop}} \frac{Pixel_{old}(r_o, c_o, f)}{F_{row}F_{col}},$$

where

$$row_{start} = (F_{row}(r_n - 1)) + 1,$$

$$row_{stop} = F_{row}r_n,$$

$$col_{start} = (F_{col}(c_n - 1)) + 1,$$

and

$$col_{stop} = F_{col} \cdot c_n.$$

The new pixel, $Pixel_{new}$, uses $r_n$, $c_n$, and $f$ for the row, column and frame indices, while the old pixel, $Pixel_{old}$, uses $r_o$, $c_o$, and $f$ for the row, column and frame indices.

For example, given a pixel frame,

$$Pixel_{old}(:,:,1) = \begin{bmatrix} 118 & 123 & 125 & 80 \\ 122 & 121 & 86 & 85 \\ 120 & 119 & 91 & 96 \\ 121 & 120 & 124 & 117 \end{bmatrix},$$

and given new row and column values of two, the new pixel frame would be,

$$\sum_{f=1}^{1} \sum_{c_n=1}^{2} \sum_{r_n=1}^{2} Pixel_{new}(r_n, c_n, 1) = \begin{bmatrix} 121 & 94 \\ 120 & 107 \end{bmatrix}.$$

It is important to note that the dimensions of the pixel will change during this process such that the size of the new pixel in the x-direction is given by

$$x_{pixel_{new}} = x_{pixel_{old}} F_{col},$$

and the size of the new pixel in the y-direction is given by

$$y_{pixel_{new}} = y_{pixel_{old}} F_{row}.$$

6.4.2. Enlarge Pixel Algorithm

To enlarge a single pixel, $Pixel_{old}(Row, Col, f)$ by a factor of $N_{row}$ and $N_{col}$, to create a new pixel, $Pixel_{new}(1, 1, f)$, use the Enlarge-pixel algorithm:

1) Calculate start and stop values for the row and column summations using

$$row_{start} = Row - \lfloor N_{row} \rfloor,$$

$$row_{stop} = Row + \lfloor N_{row} \rfloor,$$

$$col_{start} = Col - \lfloor N_{row} \rfloor,$$

and

$$col_{stop} = Col - \lfloor N_{row} \rfloor,$$

where *Row* and *Col* are row and column index of the chosen pixel to enlarge, $Pixel_{old}$, and the $f$ index represents all of the frames in the data sequence.

2) Calculate $\text{Pixel}_{\text{new}}$ using

$$\sum_{f=1}^{NumFrames} Pixel_{new}(1,1,f) = \frac{\sum_{c_o=col_{start}}^{col_{stop}} \sum_{r_o=row_{start}}^{row_{stop}} F_r F_c Pixel_{old}(r_o, c_o, f)}{\sum_{c_o=col_{start}}^{col_{stop}} \sum_{r_o=row_{start}}^{row_{stop}} F_r F_c},$$

where the partial pixel row factor is $F_r$, and the partial pixel column factor is, $F_c$. $F_r$ is equal to one unless $r$ is at the start and stop value of the row summation. At these points,

$$F_r = N_{row} - \lfloor N_{row} \rfloor.$$

$F_c$ is equal to one unless $c$ is at the start and stop values of the column summation. At these two points,

$$F_c = N_{col} - \lfloor N_{col} \rfloor.$$

For example, given a pixel frame of

$$Pixel_{old}(:,:,1) = \begin{bmatrix} 118 & 123 & 125 & 80 \\ 122 & 121 & 86 & 85 \\ 120 & 119 & 91 & 96 \\ 121 & 120 & 124 & 117 \end{bmatrix},$$

and the task of finding $Pixel_{new}(1,1,1)$ using $Pixel_{old}(2,2,1)$, $N_{row} = 0.5$, and $N_{col} = 0.7$, the new pixel value would be

$$Pixel_{new}(1,1,1) = [114].$$

As with the Sum-pixel algorithm, the dimensions of the pixel will change such that the size of the new pixel in the *x-direction* is given by

$$x_{pixel_{new}} = x_{pixel_{old}}(1 + 2N_{col}),$$

and the size of the new pixel in the *y-direction* is given by

$$y_{pixel_{new}} = y_{pixel_{old}}(1 + 2N_{row}).$$

## 6.5. Refined Change Detection

Once the Sum-pixel algorithm is complete, $Pixel_{new}$ is sent to the change detection algorithms as a new source of data. Several refinements can be compared or averaged together to increase the accuracy of the target assessments. The main concern in using this algorithm is not to lose the target signal.

# CHAPTER SEVEN

# REFINEMENT THROUGH TRACKING (RTT)

In a staged environment, one can create scenarios with only horizontal target movement. Outside of this environment, these cases are not readily available. This is an issue as the target length characteristic calculated in Chapter Five is actually the dimension of the target in the x-direction, $d_x$. If the target moves in the horizontal direction, this accurately corresponds to the target length, but any other direction of travel would produce a length calculation error. The actual length of the target would be bounded by the horizontal length calculation and movement at a 45° angle. The error is found by,

$$\varepsilon_{length} \leq \sqrt{2} d_x,$$

where $\varepsilon_{length}$ is the length error and $d_x$ is the calculated length found in Chapter Five. It is possible to minimize this error by using a Refinement Through Tracking technique. For the purposes of this discussion, it is assumed that the target trajectory is known or historical data exists where the trajectory can be calculated. Further work will investigate different tracking methods for algorithm implementation.

If you know the trajectory angle of the target, it is possible to find out more information than just the target velocity and length. The spatial target radiometry describes a target's area. By combining the temporal profile of the sub-pixel target with the spatial trajectory, it is geometrically possible to refine the target's parameters of

length and width. With the given information of the target's entry time, transit time and trajectory angle, the target length and width are not singular. This problem can be resolved by calculating an additional trajectory and from another temporal profile. Some suggestions for gaining the additional trajectory include finding a pixel where the target changes directions or rotating the static sensor. Another solution consists of applying the expected target width to length ratios to test the target profile potential categories.

## 7.1. Historical Binary Target Matrix (HBTM)

The Historical Binary Target Matrix (HBTM) is a useful tool in calculating the target trajectory angle. The HBTM historically captures the spatio-temporal progress of the target in zeros and ones, where a value of one indicates target detection occurred during the frame sequence (Figure 7.1).

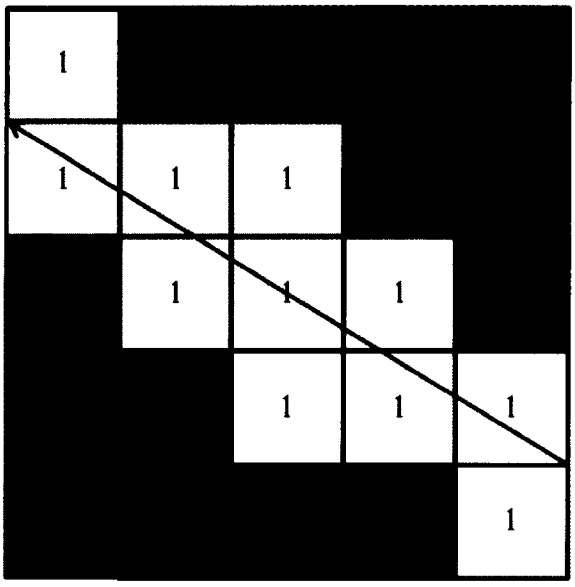



Figure 7.1 Utilizing the HBTM for calculating target trajectory

An estimated target trajectory can be found via the slope of the target pixels in the HBTM. Using the row and column range values from the HBTM, and the pixel's dimensions, the slope of the target trajectory is found by

$$Slope = \frac{(row_B - row_A)Pixel_y}{(col_B - col_A)Pixel_x},$$

where $row_B - row_A$ and $col_B - col_A$ are the rows and columns the target traverses, $Pixel_x$ is the x-dimension of the pixel, and $Pixel_y$ is the y-dimension of the pixel.

The target trajectory, $\theta$, is the inverse tangent of the slope,

$$\theta = \tan^{-1}(Slope).$$

## 7.2. Considerations

The target must traverse the entire pixel in the x- or y-direction. Several techniques mitigate this requirement. The Historical Binary Target Matrix is one such technique. If $I_{Pixel}(r,c,:)$ is the pixel of interest over a period of frames, one of these conditions must be true:

$$HBTM(r-1,c) = 1 = HBTM(r+1,c)$$

or

$$HBTM(r,c-1) = 1 = HBTM(r,c+1).$$

If the pixel of interest does not meet these conditions, then the user can choose a valid pixel from the HBTM or use one of the Super-pixel algorithm tools to ensure compliance. In Figure 7.2 the arrow marks the trajectory and the bold pixel is the Pixel of Interest (POI), in which the target does not traverse the entire pixel. By using the Sum-pixel algorithm to average the shaded pixels, the new POI is now in compliance because the target passes the entire pixel from right to left.

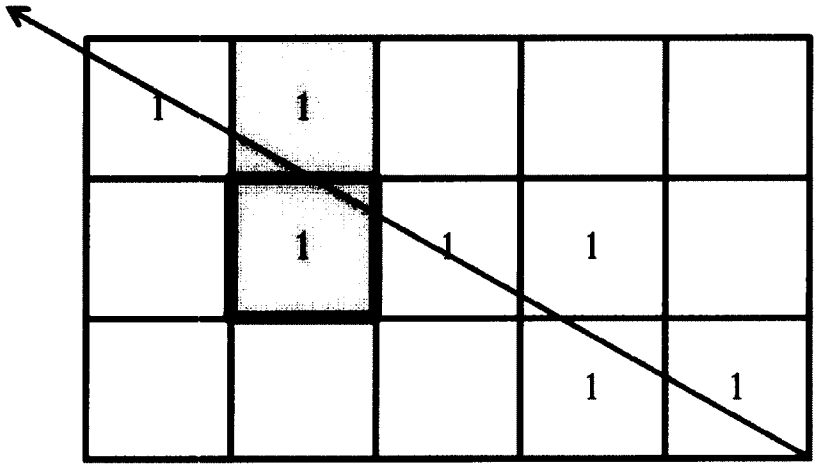



Figure 7.2 Target trajectory assessment with Sum-pixel refinement to meet pixel conditions

The Refinement Through Tracking process considers all the same assumptions as the work in the previous sections, with the exception that horizontal movement restriction is no longer required.

### 7.3. Trajectory Scenarios

Depending on the direction of travel, different dimensions of the target are found using methods described in the Target Analysis chapter, but require different calculations for each scenario. The two most interesting cases are the X-dominant and Y-dominant travel. The term dominant describes the aspect that the target will cross the entire pixel in a particular dimension (x or y). In these cases, it is still possible to calculate the target's length and width.

There are two special cases, where the target travels in a completely horizontal or vertical direction and where the target travels in a diagonal nature. In both of these cases, it is possible to find dimension for the target's length. The target's width, however, is non-singular.

### 7.3.1. X Dominant Travel

X-dominant travel describes the scenario where the target travels across the complete pixel in the x- direction (Figure 7.3). $\theta$ indicates the target's trajectory angle in the direction of travel and $d_x$ is the x component of the target's dimension.

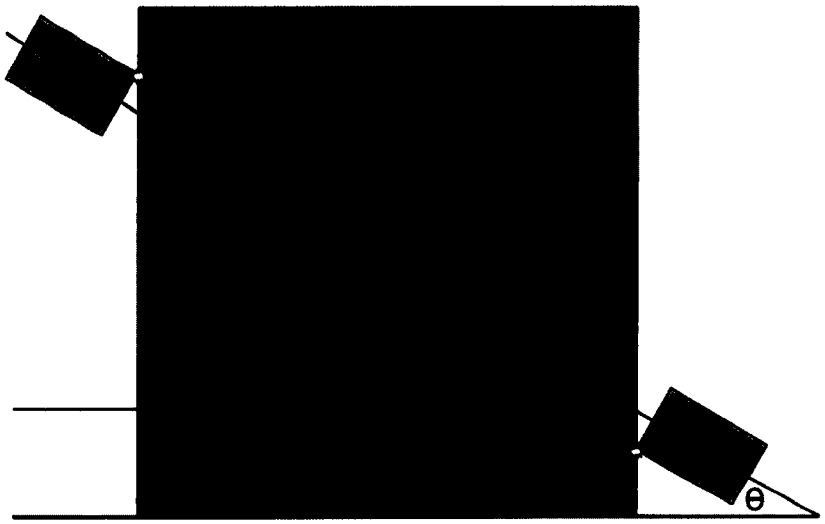



Figure 7.3 Temporal elapse of target traveling through a pixel.

The target analysis algorithm calculates the velocity in the horizontal, or x-direction. Knowing this velocity,

$$V_x = \frac{Pixel_x}{(f_{transit} - f_{entry})},$$

calculating the x component of the target's dimension, $d_x$, is possible using

$$d_x = V_x f_{entry},$$

where $Pixel_x$ is the length of the pixel in the x-direction, $f_{entry}$, is the number of frames it takes the target to completely enter the pixel and $f_{transit}$ is the number of frames it takes the target to completely cross the pixel. The Target Analysis Section describes the calculations for both $f_{entry}$ and $f_{transit}$.

A boundary box surrounds the target and $d_x$ describes the x-dimension of the target (Figure 7.4.A). Let $\alpha$ describe the partial segment of $d_x$ where the target intersects the boundary box (Figure 7.4.A).

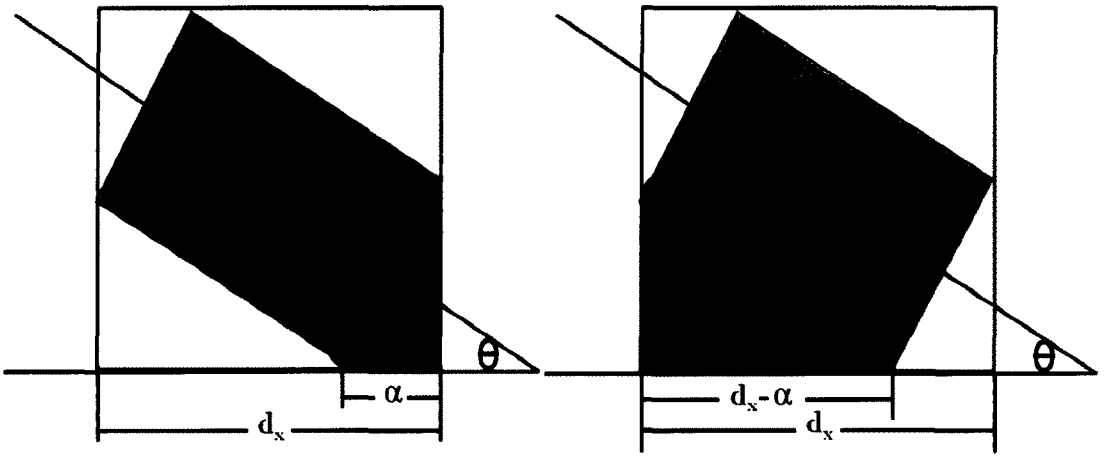



Figure 7.4 (A) Depiction of width parameters (B) Depiction of length parameters

Using the laws of sines and cosines of a right triangle, these geometrical relationships are true:

$$\sin\theta = \frac{\alpha}{d_w},$$

and

$$\cos\theta = \frac{d_{x_\theta} - \alpha}{d_l},$$

where $\theta$ is the target trajectory angle, $d_{x\theta}$ is the x-dimension of the target found by $\theta$, $d_w$ is the width of the target and $d_l$ is the length of the target. If $\theta > 90^\circ$, the supplementary angle must be used.

To solve for $d_{x\theta}$, set the equations equal to $\alpha$ to get

$$d_{x_\theta} = d_w \sin\theta + d_l \cos\theta.$$

Now, there is one equation and two unknowns, $d_w$ and $d_l$, which is a non-singular solution. If the target changes direction or the sensor is rotated, then another trajectory angle, $\gamma$, exists for the equation,

$$d_{x_\gamma} = d_w \sin\gamma + d_l \cos\gamma,$$

and combined with $d_{x\theta}$, can be solved for the target's width and length,

$$d_w = \frac{d_{x_\gamma} - \frac{d_{x_\theta}\cos\gamma}{\cos\theta}}{\sin\gamma - \tan\theta\cos\gamma},$$

and

$$d_l = \frac{d_{x_\theta}}{\cos\theta} - d_w \tan\theta.$$

Note that this process assumes both equations have X-dominant travel. If there is one trajectory with Y-dominant travel, the equation in the next section must be substituted for trajectory angle $\gamma$.

7.3.2. Y Dominant Travel

Y-dominant travel describes the scenario where the target travels across the complete pixel in the y-direction (Figure 7.5). $\theta$ indicates the target's trajectory angle in the direction of travel and $d_y$ is the y component of the target's dimension.

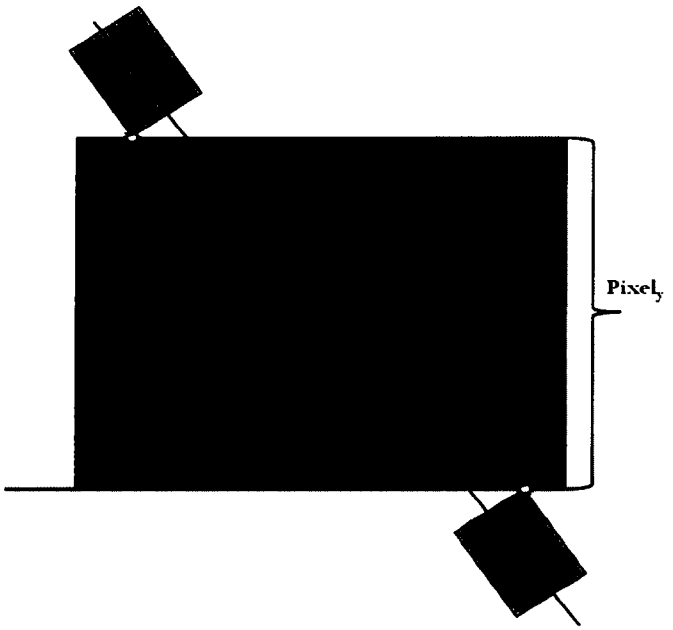



Figure 7.5 Temporal elapse of target traveling in Y-dominant direction

Due to the dominant y-direction, using the $Pixel_y$ values becomes critical in the velocity calculation. Knowing the velocity in the y-direction,

$$V_y = \frac{Pixel_y}{(f_{transit} - f_{entry})},$$

calculating the y component of the target's dimension, $d_y$, is possible using

$$d_y = V_y f_{entry},$$

where $Pixel_y$ is the length of the pixel in the y-direction, $f_{entry}$, is the number of frames it takes the target to completely enter the pixel and $f_{transit}$ is the number of frames it takes the target to completely cross the pixel. The Target Analysis Section describes the calculations for both $f_{entry}$ and $f_{transit}$.

A boundary box surrounds the target and $d_y$ describes the y-dimension of the target (Figure 7.6.A). Let $\beta$ describe as the partial segment of $d_y$ where the target intersects the boundary box (Figure 7.6.B).

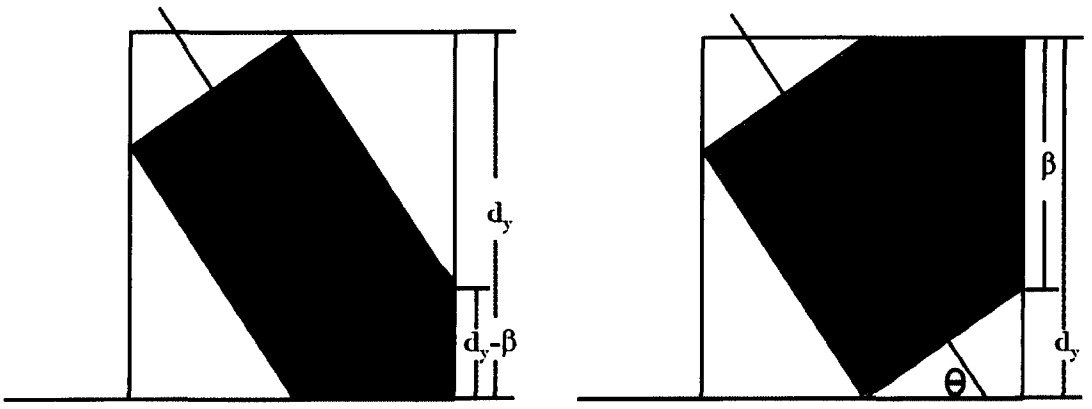



Figure 7.6 (A) Depiction of width parameters in Y-dominant travel (B) Depiction of length parameters in Y-dominant travel

Using the laws of sines and cosines of a right triangle, these geometrical relationships are true:

$$\sin\theta = \frac{\beta}{d_l},$$

and

$$\cos\theta = \frac{d_{y_\theta} - \beta}{d_w},$$

where $\theta$ is the target trajectory angle, $d_{y\theta}$ is the y-dimension of the target found by $\theta$, $d_w$ is the width of the target and $d_l$ is the length of the target. If $\theta > 90°$, the supplementary angle must be used.

Set the equations equal to $\beta$,

$$d_l \sin\theta = \beta = d_{y_\theta} - d_w \cos\theta,$$

and solve for $d_{y\theta}$ using

$$d_{y_\theta} = d_l \sin\theta + d_w \cos\theta.$$

Just as in the X-dominant travel calculations, there is one equation and two unknowns, $d_w$ and $d_l$, which is a non-singular solution. If the target changes direction or the sensor is rotated, then another trajectory angle, $\gamma$, can be used for the equation

$$d_{y_\gamma} = d_l \sin\gamma + d_w \cos\gamma,$$

and combined with $d_{y\theta}$ and solved for the target's width and length,

$$d_l = \frac{d_{y_\gamma} - \frac{d_{y_\theta}\cos\gamma}{\cos\theta}}{\sin\gamma - \tan\theta\cos\gamma},$$

and

$$d_w = \frac{d_{y_\theta}}{\cos\theta} - d_w \tan\theta.$$

Note that this process assumes that both equations have Y-dominant travel. If there is one trajectory with X-dominant travel, the equation from Section 7.3.1. must be substituted for trajectory angle $\gamma$.

7.3.3. Dominant Travel Considerations

In both the X- and Y- dominant travel calculations, if the target fails to change directions and the sensor cannot rotate, then comparative analysis may uncover the target's parameters. This concept involves bounding the target with the non-singular solution equations found earlier for $d_{x\theta}$ and $d_{y\theta}$.

Given these mathematical equations, there are, by definition, a number of values for length and width that will make this equation valid. Furthermore, for every value of length that falls within the equation boundaries, there is a specific and unique value for width that will fit and vice versa (Figure 7.7).

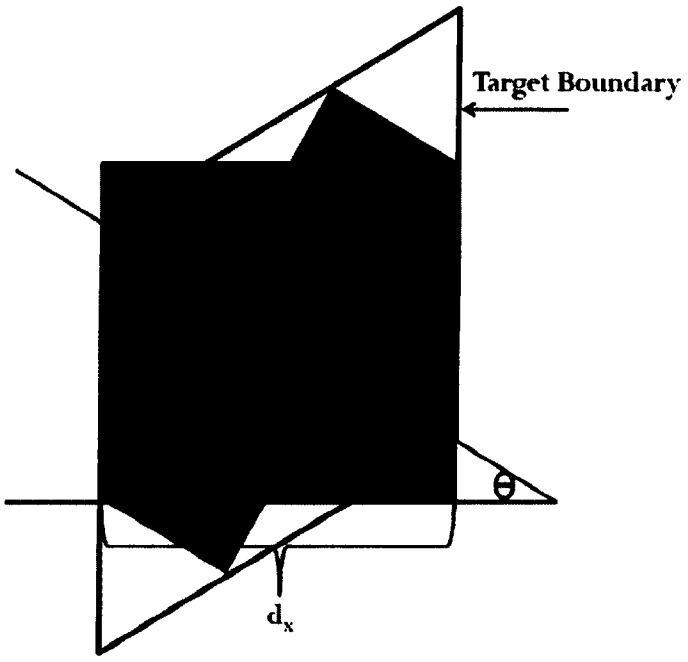



Figure 7.7 Possible Target Shapes for a Given Pixel, Velocity, and Trajectory

Therefore, the target in question is said to be described by the non-singular solution equation. Using deductive reasoning and a database of known length and width values for similar targets, inserting the values into the equation will determine if the database target matches the unidentified target's parameters.

7.3.4. Special Case 1

For targets that transverse the pixel horizontally, at 0°and 180°, or vertically, at 90° and 270° (Figure 7.8.A and 7.8.B) the length can be calculated geometrically while the target width remains an unknown parameter.

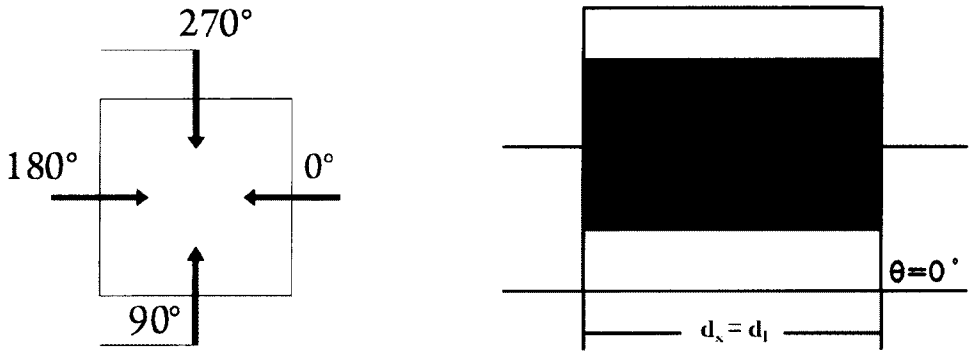



Figure 7.8 (A) Special Case 1 angles of entry (B) Pictorial depiction of Special Case 1

For cases with 0°and 180° target trajectory, the target velocity is given by

$$V_x = \frac{Pixel_x}{(f_{transit} - f_{entry})},$$

and the target length is given by

$$d_x = d_l = V_x f_{entry}.$$

For targets travelling in the vertical direction, the 90° and 270° cases, $Pixel_y$, would be substituted for $Pixel_x$, resulting in a target velocity of

$$V_y = \frac{Pixel_y}{(f_{transit} - f_{entry})},$$

and a target length of

$$d_y = d_l = V_y f_{entry}.$$

7.3.5. Special Case 2

For targets that enter a pixel from the corner at $\theta$ = 45°, 135°, 225°, or 315° (Figure 7.9), the length can be calculated geometrically while the target width will remain an unknown parameter.

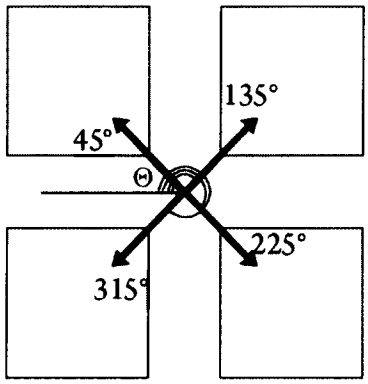



Figure 7.9 Pictoral description of Special Case 2 angles of entry into pixel

The difference between this case and Special Case 1 is that the pixel size must be converted by a $\sqrt{2}$ factor. The dominant direction of travel, shown in Figure 7.10, determines the proper equation set to use.

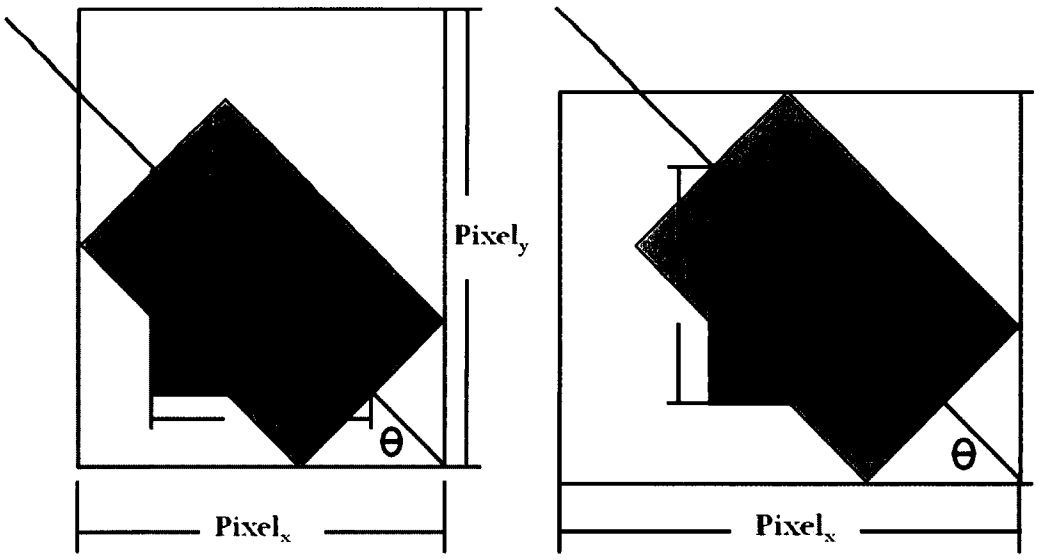



Figure 7.10 (A) Target crosses completely in x-direction (B) Target crosses completely in y-direction

For Special Case 2 scenarios with $Pixel_x < Pixel_y$, the target will completely cross the pixel in the x-direction and these equations are valid:

$$V_x = \frac{Pixel_x}{(f_{transit} - f_{entry})},$$

$$d_x = V_x \cdot f_{entry},$$

and the length of the target is found via

$$L_{target} = \sqrt{2} \cdot d_x.$$

For Special Case 2 scenarios with $Pixel_y < Pixel_x$, the target will completely cross the pixel in the y-direction and $Pixel_y$ is substituted for $Pixel_x$ in the velocity and length calculations.

# CHAPTER EIGHT

# TESTING

The sub-pixel algorithm can be used for a variety of situations and targets. For this dissertation, the model base uses ground based vehicles for detection and analysis. Therefore, the test scenarios were designed to analyze the algorithm performance for this specific situation.

There are three different sections for testing. The first is algorithm verification, which tests the computer algorithms against a compliance method, such as hand calculations, to ensure they work properly. The second test section uses the modeled data to find the boundaries of the sub-pixel algorithm. The final test section uses sensor data sequences for authentication of algorithm work and sub-pixel processing theories.

## 8.1. Algorithm Verification

MATLAB was used for algorithm development and testing. The basic data structure in MATLAB is the array, which is naturally suited to the representation of real-valued images annotated as ordered sets of intensity data. MATLAB stores the images as matrices, allowing for efficient computation of multifaceted data.

Upon coding completion, individual function testing verifies run time accuracy for each algorithm or function tool. The tables containing the different computer modules, expected results, verification method and the algorithm compliance assessment are located in Appendix A.

## 8.2. Modeled Data Testing

### 8.2.1. Overview

Each modeled simulation set starts with a staring sensor having set characteristics for sample rate, pixel size, noise and jitter. Different ground vehicles, listed in Chapter Two, are randomly chosen for insertion into single or multiple pixels (Note: Single-pixel tests will have a value of zero for jitter). The vehicle's speed is randomly selected with values ranging from half of its maximum velocity to full maximum velocity. The intensity of the vehicle is randomly chosen as well. The background intensities for the single-pixel tests are constant values. The multiple-pixel scenarios have a low gradient intensity variation across the background array.

Tests are run in batches to emulate near-real-time processing. The data outputs to an Excel Spreadsheet for further analysis and processing. The results of each test determine the parameters to refine in the change detection and analysis algorithms for optimum sensor performance. Performance metrics are Detection Rate vs. False Alarm Rate and % error evaluation of the calculated target characteristics of velocity and length to the expected values.

## 8.2.2. Scenarios

### *8.2.2.1. Change Detection Testing*

The goal of these scenarios is the comparison of the different detection methods performance on sub-pixel moving targets. The testing algorithm processes a batch of pixel frame sequences containing a modeled target through all of the change detection methods (Threshold, Duration, FAPI and the FAR filters). The Excel output files records the modeled sensor information (*Sample Rate, Pixel X, Pixel Y, Noise* and *Jitter*), the modeled target information (*Target ID, Target Length, Target Width, Target Velocity,* and *Target Intensity*), the expected critical points ($Enter_1$, $Enter_2$, $Exit_1$ and $Exit_2$), each change detection's calculated critical points from the binary matrices ($Enter_1$, and $Exit_2$), and the number of targets detected from each change detection method (Table 8.1 – Table 8.4).

Table 8.1 Example of Excel Output of Modeled Sensor and Target Information for Change Detection Testing

| Run | S Info | S Rate | Pixel X | Pixel Y | Noise | Jitter | T Info | # | Length | Width | Speed | Jbk | JTgt | No Noise | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 120 | 30 | 30 | 0 | 0 | | 6 | 8.32 | 2.43 | 24.06 | 0.2 | 0.9 | | 60 | 101 | 210 | 251 |
| 2 | | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 19.022 | 0.2 | 0.9 | | 60 | 124 | 249 | 313 |
| 3 | | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 21.784 | 0.2 | 0.9 | | 60 | 116 | 225 | 281 |
| 4 | | 120 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 15.491 | 0.2 | 0.9 | | 60 | 209 | 292 | 441 |
| 5 | | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 23.079 | 0.2 | 0.9 | | 60 | 113 | 216 | 269 |
| 6 | | 120 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 18.843 | 0.2 | 0.9 | | 60 | 89 | 251 | 280 |
| 7 | | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 20.828 | 0.2 | 0.9 | | 60 | 119 | 233 | 292 |
| 8 | | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 21.293 | 0.2 | 0.9 | | 60 | 117 | 229 | 286 |
| 9 | | 120 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 16.892 | 0.2 | 0.9 | | 60 | 125 | 273 | 338 |
| 10 | | 120 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 18.755 | 0.2 | 0.9 | | 60 | 183 | 252 | 375 |
| 11 | | 120 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 17.892 | 0.2 | 0.9 | | 60 | 189 | 261 | 390 |
| 12 | | 120 | 30 | 30 | 0 | 0 | | 6 | 8.32 | 2.43 | 17.802 | 0.2 | 0.9 | | 60 | 116 | 262 | 318 |
| 13 | | 120 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 12.968 | 0.2 | 0.9 | | 60 | 144 | 338 | 422 |
| 14 | | 120 | 30 | 30 | 0 | 0 | | 3 | 24.89 | 3.68 | 12.673 | 0.2 | 0.9 | | 60 | 296 | 344 | 580 |

Table 8.2 Example of Excel Output of Modeled Target Threshold Change Detection with FAR Filters for Change Detection Testing

| Run | Threshold | N | Y | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 1 | 1 | 1 | 61 | 251 | 15 | 1 | 61 | 251 | 30 | 1 | 61 | 251 |
| 2 | | 1 | 1 | 1 | 61 | 313 | 15 | 1 | 61 | 313 | 30 | 1 | 61 | 313 |
| 3 | | 1 | 1 | 1 | 61 | 281 | 15 | 1 | 61 | 281 | 30 | 1 | 61 | 281 |
| 4 | | 1 | 1 | 1 | 61 | 441 | 15 | 1 | 61 | 441 | 30 | 1 | 61 | 441 |
| 5 | | 1 | 1 | 1 | 61 | 269 | 15 | 1 | 61 | 269 | 30 | 1 | 61 | 269 |
| 6 | | 1 | 1 | 1 | 61 | 280 | 15 | 1 | 61 | 280 | 30 | 1 | 61 | 280 |
| 7 | | 1 | 1 | 1 | 61 | 292 | 15 | 1 | 61 | 292 | 30 | 1 | 61 | 292 |
| 8 | | 1 | 1 | 1 | 61 | 286 | 15 | 1 | 61 | 286 | 30 | 1 | 61 | 286 |
| 9 | | 1 | 1 | 1 | 61 | 338 | 15 | 1 | 61 | 338 | 30 | 1 | 61 | 338 |
| 10 | | 1 | 1 | 1 | 61 | 375 | 15 | 1 | 61 | 375 | 30 | 1 | 61 | 375 |
| 11 | | 1 | 1 | 1 | 61 | 390 | 15 | 1 | 61 | 390 | 30 | 1 | 61 | 390 |
| 12 | | 1 | 1 | 1 | 61 | 318 | 15 | 1 | 61 | 318 | 30 | 1 | 61 | 318 |
| 13 | | 1 | 1 | 1 | 61 | 422 | 15 | 1 | 61 | 422 | 30 | 1 | 61 | 422 |
| 14 | | 1 | 1 | 1 | 61 | 580 | 15 | 1 | 61 | 580 | 30 | 1 | 61 | 580 |

Table 8.3 Example of Excel Output of Modeled Target Duration Change Detection with FAR Filters for Change Detection Testing

| Run | Duration | N | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 1 | 1 | 73 | 239 | 15 | 1 | 73 | 239 | 30 | 1 | 73 | 239 |
| 2 | | 1 | 1 | 71 | 303 | 15 | 1 | 71 | 303 | 30 | 1 | 71 | 303 |
| 3 | | 1 | 1 | 70 | 272 | 15 | 1 | 70 | 272 | 30 | 1 | 70 | 272 |
| 4 | | 1 | 1 | 92 | 410 | 15 | 1 | 92 | 410 | 30 | 1 | 92 | 410 |
| 5 | | 1 | 1 | 69 | 261 | 15 | 1 | 69 | 261 | 30 | 1 | 69 | 261 |
| 6 | | 1 | 1 | 68 | 273 | 15 | 1 | 68 | 273 | 30 | 1 | 68 | 273 |
| 7 | | 1 | 1 | 70 | 283 | 15 | 1 | 70 | 283 | 30 | 1 | 70 | 283 |
| 8 | | 1 | 1 | 70 | 277 | 15 | 1 | 70 | 277 | 30 | 1 | 70 | 277 |
| 9 | | 1 | 1 | 69 | 330 | 15 | 1 | 69 | 330 | 30 | 1 | 69 | 330 |
| 10 | | 1 | 1 | 86 | 350 | 15 | 1 | 86 | 350 | 30 | 1 | 86 | 350 |
| 11 | | 1 | 1 | 88 | 363 | 15 | 1 | 88 | 363 | 30 | 1 | 88 | 363 |
| 12 | | 1 | 1 | 77 | 302 | 15 | 1 | 77 | 302 | 30 | 1 | 77 | 302 |
| 13 | | 1 | 1 | 72 | 411 | 15 | 1 | 72 | 411 | 30 | 1 | 72 | 411 |
| 14 | | 1 | 1 | 82 | 559 | 15 | 1 | 82 | 559 | 30 | 1 | 82 | 559 |

Table 8.4 Example of Excel Output of Modeled Target FAPI Change Detection with FAR Filters for Change Detection Testing

| Run | FAPI | N | Y | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 1 | 1 | 1 | 63 | 290 | 15 | 1 | 63 | 290 | 50 | 1 | 63 | 290 |
| 2 | | 1 | 1 | 1 | 64 | 349 | 15 | 1 | 64 | 349 | 50 | 1 | 64 | 349 |
| 3 | | 1 | 1 | 1 | 63 | 319 | 15 | 1 | 63 | 319 | 50 | 1 | 63 | 319 |
| 4 | | 1 | 1 | 1 | 64 | 475 | 15 | 1 | 64 | 475 | 50 | 1 | 64 | 475 |
| 5 | | 1 | 1 | 1 | 63 | 307 | 15 | 1 | 63 | 307 | 50 | 1 | 63 | 307 |
| 6 | | 1 | 1 | 1 | 64 | 314 | 15 | 1 | 64 | 314 | 50 | 1 | 64 | 314 |
| 7 | | 1 | 1 | 1 | 63 | 329 | 15 | 1 | 63 | 329 | 50 | 1 | 63 | 329 |
| 8 | | 1 | 1 | 1 | 63 | 324 | 15 | 1 | 63 | 324 | 50 | 1 | 63 | 324 |
| 9 | | 1 | 1 | 1 | 64 | 374 | 15 | 1 | 64 | 374 | 50 | 1 | 64 | 374 |
| 10 | | 1 | 1 | 1 | 64 | 411 | 15 | 1 | 64 | 411 | 50 | 1 | 64 | 411 |
| 11 | | 1 | 1 | 1 | 64 | 426 | 15 | 1 | 64 | 426 | 50 | 1 | 64 | 426 |
| 12 | | 1 | 1 | 1 | 64 | 354 | 15 | 1 | 64 | 354 | 50 | 1 | 64 | 354 |
| 13 | | 1 | 1 | 1 | 64 | 455 | 15 | 1 | 64 | 455 | 50 | 1 | 64 | 455 |
| 14 | | 1 | 1 | 2 | 78 | 581 | 15 | 2 | 78 | 581 | 50 | 1 | 78 | 581 |

For visual comparison, a comparative graphic displays the modeled target signal and the change detection information (Figure 8.1). The change detection testing results are located in Appendix B.

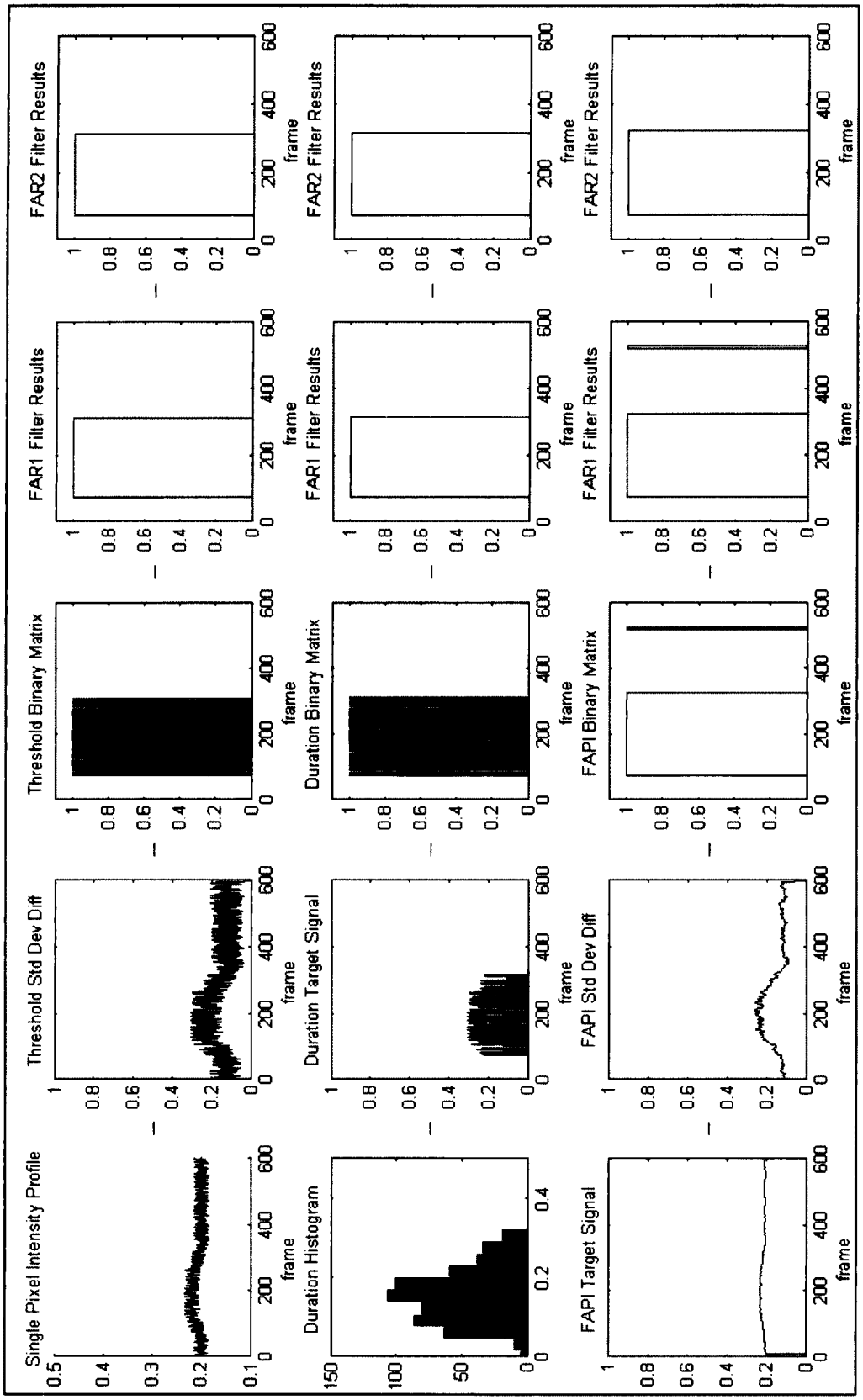



Figure 8.1 Comparative graphic display of change detection algorithms

The target model parameters for the first batch run has a constant sample rate (120 frames/s), pixel size (30x30 $m^2$), zero noise and zero jitter with random variations, as described in the previous section. Consecutive batches vary the sensor parameters as shown in Table 8.5. To ensure sub-pixel targets, values for Pixel X and Y must be larger than the respective modeled target vehicle parameters in Chapter Two.

Table 8.5 Batch Run Variations for Sensor Parameters

| | Variations |
|---|---|
| Sample Rate (frames/sec) | 30, 60, 90, 120 |
| Pixel X (m) | 30 – 60 |
| Pixel Y (m) | 5 - 60 |
| Noise | 0 - 0.05 |
| Jitter | 0 - 0.5 |

*8.2.2.2.* *Target Analysis Testing*

The target analysis testing utilizes the same sensor and target inputs as the change detection testing scenarios with the same variations as seen in Table 8.5. Every test batch includes three runs for each sensor and target model. The first run uses the Threshold Change Detection and feeds the results into a simultaneously run Polyfit and Fermifit algorithm. MATLAB exports the outcome to an Excel file.

Table 8.6 Example of Excel Output of Modeled Sensor and Target Information for Target Analysis Testing

| Run | S Info | S Rate | Pixel X | Pixel Y | Noise | Jitter | T Info | # | Length | Width | Speed | Jbk | JTgt | No Noise | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 120 | 30 | 30 | 0.01 | 0.1 | | 4 | 9.08 | 2.59 | 16.36 | 0.2 | 0.9 | | 60 | 127 | 280 | 347 |
| 2 | | 120 | 30 | 30 | 0.01 | 0.1 | | 5 | 19.27 | 2.44 | 16.52 | 0.2 | 0.9 | | 60 | 200 | 278 | 418 |
| 3 | | 120 | 30 | 30 | 0.01 | 0.1 | | 5 | 19.27 | 2.44 | 20.5 | 0.2 | 0.9 | | 60 | 173 | 236 | 349 |
| 4 | | 120 | 30 | 30 | 0.01 | 0.1 | | 6 | 8.32 | 2.43 | 22.02 | 0.2 | 0.9 | | 60 | 105 | 224 | 269 |
| 5 | | 120 | 30 | 30 | 0.01 | 0.1 | | 3 | 24.89 | 3.68 | 19.27 | 0.2 | 0.9 | | 60 | 215 | 247 | 402 |
| 6 | | 120 | 30 | 30 | 0.01 | 0.1 | | 3 | 24.89 | 3.68 | 14.99 | 0.2 | 0.9 | | 60 | 259 | 300 | 499 |
| 7 | | 120 | 30 | 30 | 0.01 | 0.1 | | 1 | 4.57 | 2.16 | 18.59 | 0.2 | 0.9 | | 60 | 89 | 254 | 283 |
| 8 | | 120 | 30 | 30 | 0.01 | 0.1 | | 4 | 9.08 | 2.59 | 18.16 | 0.2 | 0.9 | | 60 | 120 | 258 | 318 |
| 9 | | 120 | 30 | 30 | 0.01 | 0.1 | | 5 | 19.27 | 2.44 | 17.03 | 0.2 | 0.9 | | 60 | 196 | 271 | 407 |
| 10 | | 120 | 30 | 30 | 0.01 | 0.1 | | 1 | 4.57 | 2.16 | 17.35 | 0.2 | 0.9 | | 60 | 92 | 268 | 300 |

Table 8.7 Example of Excel Output of Modeled Target Threshold Change Detection with FAR Filters for Target Analysis Testing

| Run | Threshold | N | Y | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 3 | 1 | 51 | 272 | 283 | 15 | 1 | 87 | 328 | 30 | 1 | 87 | 328 |
| 2 | | 3 | 1 | 25 | 154 | 313 | 15 | 1 | 101 | 378 | 30 | 1 | 101 | 378 |
| 3 | | 3 | 1 | 24 | 126 | 279 | 15 | 1 | 73 | 336 | 30 | 1 | 73 | 336 |
| 4 | | 3 | 1 | 23 | 103 | 106 | 15 | 1 | 84 | 240 | 30 | 1 | 84 | 240 |
| 5 | | 3 | 1 | 38 | 153 | 303 | 15 | 2 | 69 | 383 | 30 | 1 | 69 | 383 |
| 6 | | 3 | 1 | 47 | 176 | 391 | 15 | 5 | 81 | 465 | 30 | 1 | 81 | 465 |
| 7 | | 3 | 1 | 27 | 236 | 239 | 15 | 8 | 115 | 256 | 30 | 1 | 115 | 256 |
| 8 | | 3 | 1 | 74 | 136 | 144 | 15 | 10 | 67 | 298 | 30 | 2 | 67 | 298 |
| 9 | | 3 | 1 | 42 | 186 | 191 | 15 | 4 | 167 | 305 | 30 | 1 | 167 | 305 |
| 10 | | 3 | 1 | 20 | 236 | 238 | 15 | 9 | 236 | 270 | 30 | 2 | 236 | 270 |

Table 8.8 Example of Excel Output of Modeled Target PolyFit and FermiFit Analysis for Target Analysis Testing

| Run | Poly | En1 | En2 | Ex1 | Ex2 | Fermi | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 63 | 131 | 289 | 345.3 | | 63 | 131 | 289 | 345.3076 |
| 2 | | 59 | 196 | 276 | 419.4 | | 59 | 196 | 276 | 419.3685 |
| 3 | | 50 | 172 | 239 | 352.4 | | 50 | 172 | 239 | 352.3753 |
| 4 | | 20 | 86 | 223 | 259.2 | | 20 | 75 | 204 | 259 |
| 5 | | 51 | 218 | 241 | 406.1 | | 51 | 160 | 297 | 406 |
| 6 | | 56 | 256 | 308 | 503.8 | | 56 | 197 | 363 | 504 |
| 7 | | 100 | 126 | 238 | 270.6 | | 100 | 135 | 236 | 271 |
| 8 | | 64 | 99 | 245 | 307.8 | | 64 | 127 | 245 | 308 |
| 9 | | 130 | 189 | 292 | 330.1 | | 130 | 185 | 275 | 330 |
| 10 | | 233 | 238 | 284 | 320.4 | | 233 | 260 | 293 | 320 |

The second and third runs exchange the Threshold Change Detection algorithm for the Duration Change Detection and FAPI Change Detection, respectively. As with the change detection testing, a visual comparison graph displays for the user (Figure 8.2).

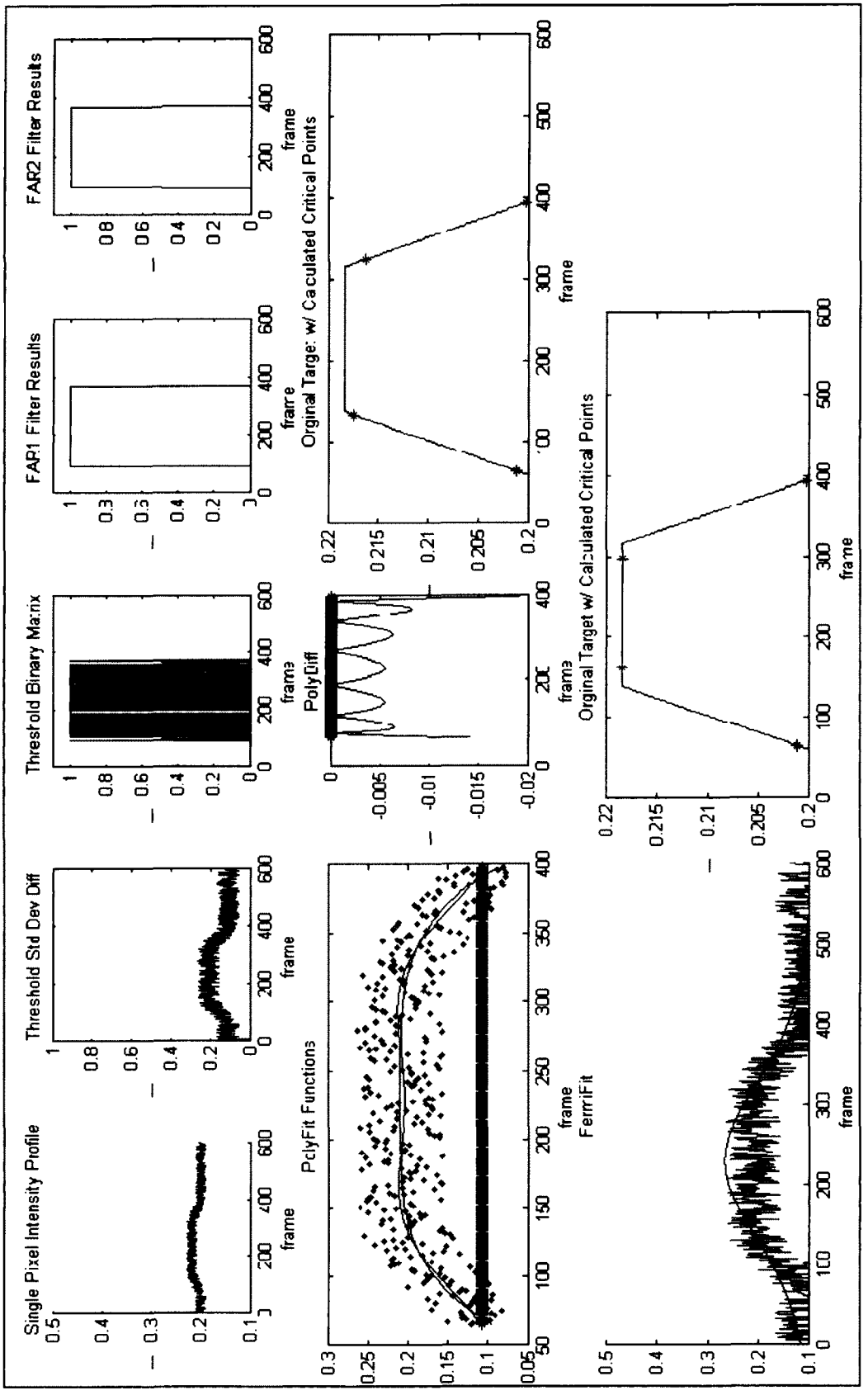



Figure 8.2 Example illustration of target analysis plots from PolyFit and FermiFit

## 8.3. Sensor Data Testing

### 8.3.1. Data Information

RP Flight Systems Inc., who specializes in aerial data collection systems, provided a series of sensor data for testing. A stationary Panasonic Lumix LX2 secured to a tower 18 m above ground level shot sequences of continuous video of a remote controlled truck crossing a bumpy, grassy terrain. The target has a body color that, starting from the front, changes from orange to yellow to white to silver to black. The windshield and windows are solid black. The dimensions of the truck body are 0.406 m long by 0.165 m wide [53].

The data sequences have both horizontal and diagonal target travel, which provides information to test both the main and the refinement through tracking algorithms. The target travels through portions of heavily shadowed background as well as extremely bright areas (Figure 8.3).

Figure 8.3 Image of RP Flight Systems Inc., remote controlled truck with varying background taken from 18 m above ground

The target in the datasets is not sub-pixel. To ensure conformity with the nature of this dissertation, the Super-pixel algorithms enlarge or sum the pixels so they are larger than the target.

8.3.2. Scenarios

The data scenarios test the change detection, target analysis, super-pixel and Refinement Through Tracking (RTT) algorithms. In order to keep computational time to a minimum, specific pixel sequences of the original dataset pertinent to testing the algorithm robustness are chosen and saved in smaller data files. The criteria for choosing pixel sections are background intensity, target motion and pixel shape (Table 8.9).

Table 8.9 Sensor Data Testing Scenario Criteria

| **Algorithm Focus** | **Pixel Shape** | **Background** | **Target Motion** |
|---|---|---|---|
| Change Detection<br>Target Analysis<br>Sum Pixel<br>Enlarge Pixel | Multipixel | Uniform bright | Horizontal |
| | Large Square | Uniform bright | Horizontal |
| | Small Square | Uniform bright | Horizontal |
| | Short Rectangle | Uniform bright | Horizontal |
| | Long Rectangle | Uniform bright | Horizontal |
| | Multipixel | Uniform shadow | Horizontal |
| | Large Square | Uniform shadow | Horizontal |
| | Small Square | Uniform shadow | Horizontal |
| | Short Rectangle | Uniform shadow | Horizontal |
| | Long Rectangle | Uniform shadow | Horizontal |
| | Multipixel | Split bright/shadow | Horizontal |
| | Large Square | Split bright/shadow | Horizontal |
| | Small Square | Split bright/shadow | Horizontal |
| | Short Rectangle | Split bright/shadow | Horizontal |
| | Long Rectangle | Split bright/shadow | Horizontal |

The testing algorithm imports the saved file, formats the sequences into a single intensity $I_{Pixel}(r, c, f)$ array, and runs the appropriate sub-pixel algorithm. MATLAB exports the testing results to an Excel file, similar to the ones created in the modeled data testing, for post-run analysis.

# CHAPTER NINE

# RESULTS AND DISCUSSION

This section encompasses a summary and discussion of the modeled and sensor data test results. Appendix B contains the unabridged tabular data outputs.

## 9.1. Sensor Parameter Analysis and Performance

As expected, the sensor characteristics of sample rate, pixel size, noise and jitter varied throughout the modeled testing portion, and affected the algorithm's performance. The analysis in this section examines these effects in detail.

### 9.1.1. Sensor Sample Rate

As sensor sample rate increases, there are more data points during a specified time interval. In the change detection algorithms, this translates to a higher false alarm rate as a higher number of noisy or jittered points exceed the thresholds or duration filters.

Table 9.1 shows the results of the Duration Change Detection algorithm for a 30x30 $m^2$ pixel frame, with a noise and jitter value of 0.01 and 0.05 respectively. The false alarms were calculated from an average of 10 runs per sample rate (Appendix B).

Table 9.1 Sample Rate Effects on Detection and False Alarms

| Sample Rate (frames/s) | Correct Detections | False Alarms |
|---|---|---|
| 10 | 1 | 2.3 |
| 30 | 1 | 8.7 |
| 60 | 1 | 14.7 |
| 90 | 1 | 20.8 |
| 120 | 1 | 22 |

An advantage to the longer temporal profile of the higher sampled sequences is reduction in the target characteristic percent error calculations. Table 9.2 shows the results of the PolyFit algorithm for a 30x30 $m^2$ pixel frame, with a noise and jitter value of 0.01 and 0.05 respectively. The error was calculated from an average of 10 runs per sample rate and normalized to seconds instead of frames to provide a consistent comparison (Appendix B).

Table 9.2 Sample Rate Effects on Critical Point Given in Seconds of Error

| Sample Rate | $Enter_1$ (s) | $Enter_2$ (s) | $Exit_1$ (s) | $Exit_2$ (s) | Total Error (s) |
|---|---|---|---|---|---|
| 10 | 0.316 | 2.353 | 3.464 | 5.141 | 2.818 |
| 30 | 0.163 | 0.208 | 0.258 | 0.246 | 0.219 |
| 60 | 0.128 | 0.206 | 0.362 | 0.145 | 0.210 |
| 90 | 0.109 | 0.170 | 0.195 | 0.111 | 0.146 |
| 120 | 0.103 | 0.151 | 0.165 | 0.046 | 0.116 |

The initial assumption that a target must travel through more than one frame in the sequence limits the sample rates minimum value because the target may not be recorded at all during sensor data gathering. In addition to this consideration, the sample rate is independent of the algorithms ability to detect the target.

The parameters that influence the success of the change detection algorithm are the ones that determine the differences in intensity values between the background and the target. The pixel size, noise and jitter directly affect this difference.

9.1.2. Pixel Size

The pixel size is inversely proportional to the intensity delta. However, smaller pixels provide a shorter target travel time resulting in larger critical point % error as determined with the sample rates. A rectangular shape that minimizes the background with a shorter width and a longer horizontal length provides the best target to pixel ratio and largest intensity variation. Figure 9.1 shows the increase in intensity variation between the background and a target (HEMTT traveling at 23m/s with 0.02 noise and sampled at 120 frames/s) by changing from a 30x30 $m^2$ to a 30x5 $m^2$ pixel.

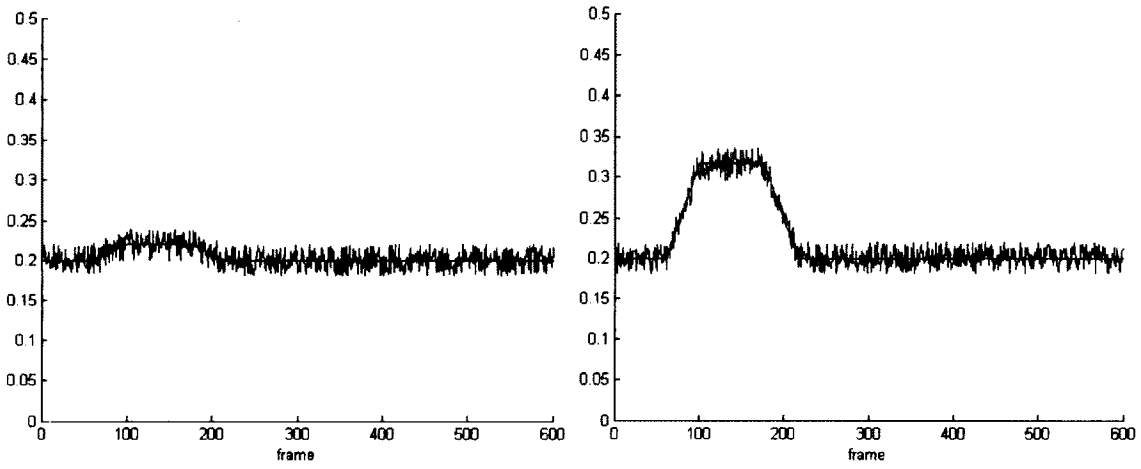



Figure 9.1 (A) 30x30 $m^2$ Pixel with HEMTT Target (B) 30x5 $m^2$ Pixel with HEMTT Target

Utilizing this concept might not be feasible in the original processing of the sensor data due to a fixed pixel size, but when employing the Super-pixel algorithms, summing or enlarging pixels to a rectangular shape in the motion of target trajectory is beneficial.

The longer target profile has a more accurate target approximation with the PolyFit and FermiFit functions. Table 9.3 shows the results of the PolyFit algorithm for modeled targets, with a noise value of 0.01 and a Sample Rate of 120. The *Pixel X* value was 30 m and the *Pixel Y* varied from 5 to 10 to 30 m. The % error was calculated from an average of 10 runs per sample rate and normalized to seconds instead of frames (Appendix B). As expected, the narrow rectangular pixel with 30x5 $m^2$ dimensions had better approximations of the critical target point.

Table 9.3 Pixel X Effects on Critical Point Given in Seconds of Error

| **Pixel Y** | **$Enter_1$ (s)** | **$Enter_2$ (s)** | **$Exit_1$ (s)** | **$Exit_2$ (s)** | **Total Error (s)** |
|---|---|---|---|---|---|
| 5 | 0.762 | 0.905 | 0.701 | 3.049 | 1.354 |
| 10 | 0.902 | 1.027 | 0.784 | 3.417 | 1.532 |
| 30 | 1.217 | 0.968 | 0.977 | 3.319 | 1.620 |

### 9.1.3. Noise

As anticipated, the higher the noise, the harder the target is to detect. Table 9.4 shows the change detection performance for different values of noise.

Table 9.4 Percent Detection and False Alarm Frames for Noise Variations

| N=2 | Threshold | | After FAR Filters | |
|---|---|---|---|---|
| **Noise** | **Detect** | **False Alarms** | **Detect** | **False Alarms** |
| 0 | 69.48% | 0.00% | 69.48% | 0.00% |
| 0.01 | 52.11% | 0.11% | 77.98% | 0.48% |
| 0.02 | 41.17% | 2.34% | 81.32% | 13.13% |
| 0.03 | 31.58% | 4.46% | 83.40% | 24.96% |
| 0.04 | 26.96% | 2.87% | 76.18% | 14.85% |
| 0.05 | 19.30% | 3.36% | 66.94% | 23.45% |

The testing scenario used for the table had a threshold $N$ value of 2.0, jitter equal to zero, a 30x30 $m^2$ pixel frame, a 120 frame/s sample rate, and modeled targets that varied in target type and velocity. The noise ranged from 0.0 to 0.5, with 50 runs per noise value (300 total runs) used for the data analysis (Appendix B). The detection percentages, *Detect %,* reflect the number of frames detected that were actual target frames, divided by the total number of target frames, which is given by,

$$Detect\ \% = \frac{Detected\ Target\ Frames}{Total\ Target\ Frames}. \tag{9.1}$$

The False-Alarm percentages reflect the number of frames that were improperly categorized as having a target, given by

$$False\ Alarm\ \% = \frac{Detected\ Non-Target\ Frames}{Total\ Non-Target\ Frames}. \tag{9.2}$$

On a positive note, the False-Alarm Rate (FAR) filters do reduce the number of false alarms by connecting the disjointed signals and by filtering out targets that do not meet the minimum temporal frame requirement. The FAR filters are successful at increasing the number of target frames detected, but this is at the expense of extra computation requirements for processing the false alarms. However, the FAR filter variables are tunable and the filter algorithm showed success when ran more than once. The testing used the same values for the FAR variables for all noise situations to provide a robust comparison. In practice, a sensor with a known noise value would have the FAR filters fine tuned for particular situations to reduce the FAR percentages.

Other tunable parameters are the $N$ values in the detection algorithms, which determine the number of standard deviations it uses as the threshold values or the number of bins to use for the duration method. A low value of $N$ ensures maximum detection, but this is at the expense of a large amount of false alarms. In many instances, with a low $N$ value, every frame is a detection, which renders the change detection algorithm useless. In addition, in these cases, the FAR filters offer little value as there is no way to temporally distinguish what is a target and what is not. If the $N$ value is too high, although there are no false alarms, the target will not be detected at all. Depending on the sensor, an analysis should be done to find the best $N$ value for target detection.

Table 9.5 shows an example analysis for a noisy sensor with zero jitter, a 120 frame/s Sample Rate, a 30x30 $m^2$ pixel size and a variety of targets travelling at different speeds.

Table 9.5 Percent Detection and False Alarm Frames and Missed Detections in Noisy Sensor with N Variations

| **Noise=0.05** | **Threshold** | | **After FAR Filters** | | **Missed Detects** |
|---|---|---|---|---|---|
| **N** | **Detect** | **False Alarms** | **Detect** | **False Alarms** | |
| 1.0 | 46.66% | 26.10% | 99.98% | 91.00% | 0/50 |
| 1.5 | 35.20% | 13.89% | 96.97% | 67.10% | 0/50 |
| 2.0 | 19.30% | 3.36% | 69.36% | 23.45% | 2/50 |
| 2.5 | 14.78% | 0.02% | 46.00% | 0.24% | 12/50 |
| 3.0 | 7.58% | 0.00% | 27.64% | 0.00% | 24/50 |
| 3.5 | 2.28% | 0.00% | 7.58% | 0.00% | 38/50 |
| 4.0 | 1.39% | 0.00% | 5.84% | 0.00% | 41/50 |

For each value varying from 1.0 to 4.0, the threshold detection algorithm tested 50 modeled targets. The number of frames detected and the number of false alarm frames were calculated using Eqns. (9.1) and (9.2). The detection algorithm kept track of the number of missed detections for additional comparison. For this example, the best *N* value is 2.0, which detected 96% of the targets.

9.1.4. Jitter

Jitter affects the capabilities of the detection algorithms much like the noise parameter. Table 9.6 shows the analysis of varying the jitter values of a pixel from 0 to 0.5.

Table 9.6 Percent Detection and False Alarm Frames for Different Jitter Variations

| **N=2.5** | **Threshold** | | **After FAR Filters** | |
|---|---|---|---|---|
| **Jitter** | **Detect** | **False Alarms** | **Detect** | **False Alarms** |
| 0 | 28.32% | 0.00% | 28.32% | 0.00% |
| 0.1 | 20.39% | 0.00% | 21.71% | 0.00% |
| 0.2 | 19.03% | 0.00% | 24.94% | 0.00% |
| 0.3 | 14.50% | 0.12% | 25.77% | 1.13% |
| 0.4 | 18.86% | 0.93% | 35.82% | 5.31% |
| 0.5 | 12.89% | 1.34% | 38.24% | 7.39% |

Fifty modeled targets varying in type and velocity traveled through an area imaged by a sensor that had zero noise, a 120 frame/s sample rate and recorded a 4x4 image array with varying background and 30x30 $m^2$ pixel sizes. The table values are calculated using Eqs. (9.1) and (9.2). A threshold detection $N$ value of 2.5 showed a decreased performance over the noise table, but the trend was still the same.

The same scenario was used to analyze the threshold $N$ parameter in jitter situations, with the exception that jitter was held constant and the $N$ value varied from 1.0 to 4.0 (Table 9.7).

Table 9.7 Percent Detection and False Alarm Frames and Missed Detections for Jittered Pixels with Different N Variations

| **Jitter=0.25** | **Threshold** | | **After FAR Filters** | | **Missed Detects** |
|---|---|---|---|---|---|
| **N** | **Detect** | **False Alarms** | **Detect** | **False Alarms** | |
| 1.0 | 78.61% | 19.58% | 99.54% | 85.68% | 0/50 |
| 1.5 | 54.00% | 5.12% | 90.09% | 31.09% | 0/50 |
| 2.0 | 40.65% | 0.35% | 65.23% | 1.51% | 2/50 |
| 2.5 | 18.80% | 0.05% | 28.90% | 0.25% | 28/50 |
| 3.0 | 17.20% | 0.01% | 23.45% | 0.00% | 33/50 |
| 3.5 | 10.55% | 0.00% | 12.75% | 0.00% | 36/50 |
| 4.0 | 8.91% | 0.00% | 11.77% | 0.00% | 39/50 |

The results mirror those from the noise analysis, with an $N$ value of 2.0 providing the best overall value for the change detection algorithm. This value remains tunable, but as a consequence of these tests, the detection algorithms were set to a default value of 2.0 in the MATLAB coding.

As shown previously, changes in sensor characteristics produce significant differences in the performance of sub-pixel detection and analysis. For this reason, the tunable variables in the threshold, duration, FAPI, and FAR filters are imperative to the algorithms success for a variety of sensors and target situations.

## 9.2. Method Analysis and Performance

### 9.2.1. Change Detection

The best measures of the change detection algorithms for the sub-pixel algorithm are the ability to detect the target and the accuracy of the critical $Enter_1$ and $Exit_2$ points. These measures are vital as detection is necessary to proceed to the target analysis algorithm and the accuracy of the critical points determines the effectiveness of the PolyFit and FermiFit methods. Table 9.8 compares the different change detection algorithms. The testing scenario ran 250 tests varying target vehicle and target speed. The Sample Rates ranged from 30 to 120 frames per second, the noise values were between 0 and 0.05, and the pixel size ranged from a 30x30 $m^2$ to a 60x60 $m^2$ pixel, with rectangular pixels included.

Table 9.8 Change Detection Error of Critical Points $Enter_1$ and $Exit_2$

| | **Error in Seconds** | |
|---|---|---|
| | **$Enter_1$** | **$Exit_2$** |
| **Threshold** | 0.821 | 1.087 |
| **Duration** | 0.870 | 1.142 |
| **FAPI** | 0.513 | 2.560 |

The Table 9.8 only includes runs where the FAPI algorithm received no run time errors. For the majority of the runs, the FAPI algorithm did not work, leaving the Threshold and Duration algorithms as the most effective. The FAPI had convergence

issues due to the weighting matrix initialization and is not very reliable. Only the tests that worked for FAPI are included in the previous table. Further work will include an investigation on how to mitigate the convergence problems.

The modular process provides an additional positive outcome. Since the already existing change detection algorithms were slightly modified using the Binary Matrix technique, newer or improved change detection algorithms can be similarly modified and used within this process.

### 9.2.2. Target Analysis

Both the PolyFit and the FermiFit methods served well in obtaining the target's critical points $Enter_2$ and $Exit_1$. Table 9.9 shows the error, in seconds, for the critical point calculation via the PolyFit and FermiFit methods (Table B.11).

Table 9.9 Target Analysis Second Error of Critical Points $Enter_1$ and $Exit_2$

| | **Error in Seconds** | | | |
|---|---|---|---|---|
| | **$Enter_1$** | **$Enter_2$** | **$Exit_1$** | **$Exit_2$** |
| **PolyFit** | 0.163 | 0.617 | 0.879 | 1.135 |
| **FermiFit** | 0.046 | 0.502 | 0.838 | 1.015 |

The FermiFit $Enter_1$ and $Exit_2$ points are independent of the FermiFit analysis and are dependent completely on the change detection method chosen in the sub-pixel process.

The challenge encountered with the PolyFit function was how to handle circumstances with less than six intersection points. If the PolyFit analysis does not find six intersection points between $Poly_6$ and $Poly_4$, the algorithm stops working. The source of this problem traces back to the $Enter_1$ and $Exit_2$ critical points retrieved from the Target Binary Matrix. When target frames are lost during the detection algorithm,

there are not enough data points to accurately recapture the trapezoid shape. The successful solution to this issue was to create a patch program that performs a quick error check during the intersect point calculation. If there are less than six intersection points found, frames are added to the target sequence until the requirement is met. If all sequence frames are accounted for, then the target does not exist and the analyzed sequence categorizes as a false alarm.

The FermiFit analysis is very sensitive to the critical points $Enter_1$ and $Exit_2$ and is highly dependent on the detection algorithm for these values. While PolyFit can adjust for a truncated Target Binary Matrix by adding frames until it meets the six intersection point criteria, FermiFit does not have this error check. As a result, critical points $Enter_1$ and $Exit_2$ do not undergo any refinement in this process and is therefore less adaptable.

Overall, the best choice for analyzing the target is the PolyFit method due to the ability of the algorithm to refine the critical $Enter_1$ and $Exit_2$ data points. For this method, the main requirement is to detect the target and to get a general estimation of the target length.

### 9.2.3. Velocity and Length Calculations

A graphical user interface (GUI) ran a series of post processing tests, which allowed the user to tune the $N$ and $FAR$ values for each scenario (Figure 9.2). The actual target velocity and lengths were entered into the upper left hand corner of the GUI and the resulting calculations were displayed adjacently for comparison purposes.

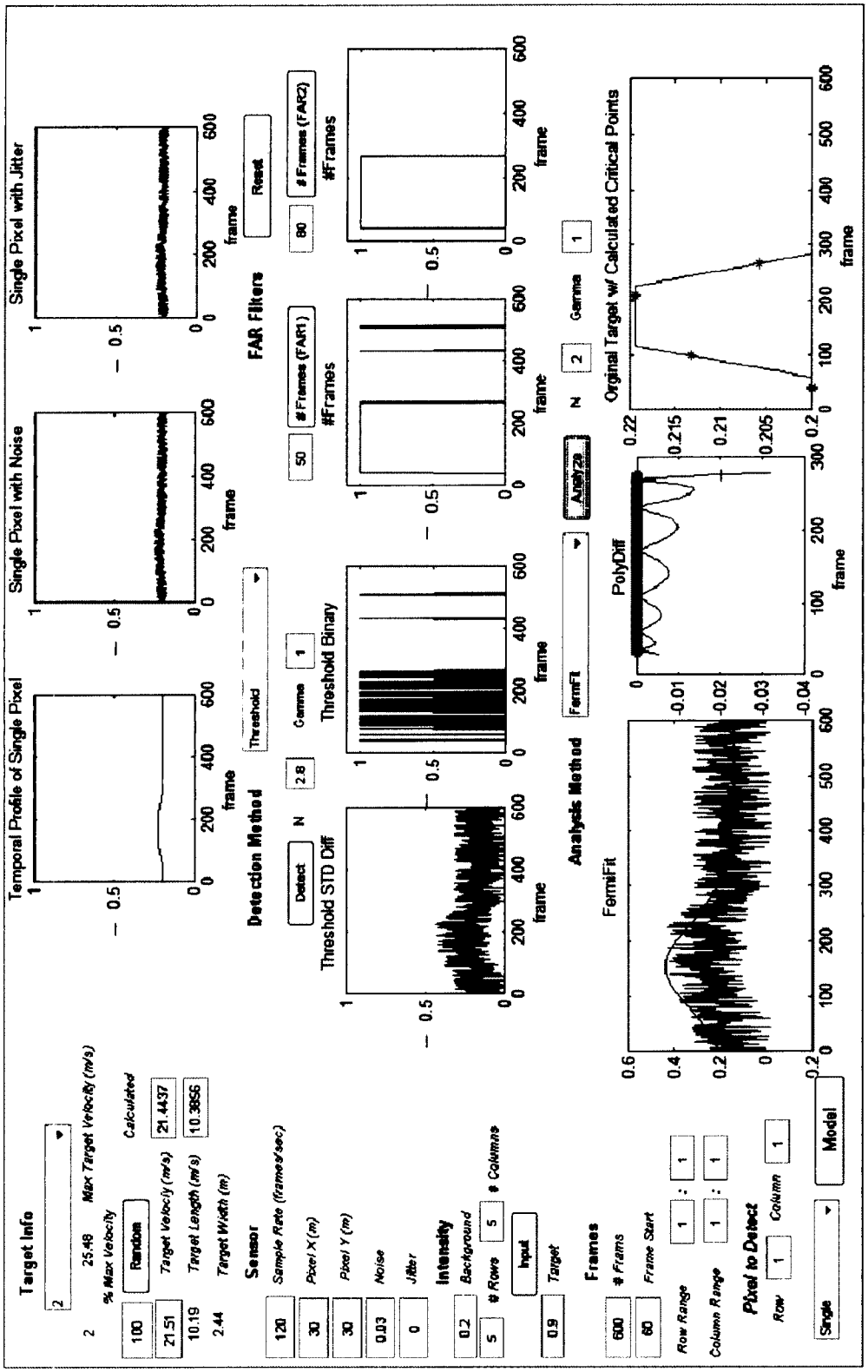



Figure 9.2 GUI for Sub-pixel Detection and Analysis

The scenarios evaluated three types of vehicles, the HUMMWV (4.57 m length), the HEMTT (10.19 m length), and the M1070 (24.89 m length). The noise values varied from 0.01 to 0.03 with 30 tests run on each scenario. The velocity values were randomly selected from 50% to maximum velocity. The detection and analysis methods varied for each run to show the modularity of the process. With tuning, the sub-pixel process detected the target, found the critical target points, and calculated the length and velocity. Table 9.10 shows the averaging of the length and the percent errors for each target and noise value; the actual test runs are located in Table B.17.

Table 9.10 Velocity and Length Test Results

| | | **Actual** | **Calculated** | | |
|---|---|---|---|---|---|
| **Target** | **Noise** | **Length** | **Length** | **Length % Error** | **Velocity % Error** |
| **HUMMWV** | 0.01 | 4.57m | 4.69 m | 2.54 | 7.99 |
| | 0.02 | | 4.40 m | 3.69 | 7.61 |
| | 0.03 | | 4.95 m | 8.22 | 14.38 |
| **HEMTT** | 0.01 | 10.19 m | 10.02 m | 1.69 | 5.27 |
| | 0.02 | | 10.45 m | 1.40 | 6.22 |
| | 0.03 | | 10.62 m | 4.31 | 3.52 |
| **M1070** | 0.01 | 24.89 m | 23.10 m | 7.18 | 6.51 |
| | 0.02 | | 23.46 m | 5.73 | 11.33 |
| | 0.03 | | 21.24 m | 14.64 | 3.21 |

The velocity and length calculations are very sensitive to the tunable parameters. The accuracy of the algorithm decreases when the noise increases in value, as expected. Table 9.11 put the results into perspective.

Table 9.11 Delta Length and Delta Velocity Calculations

| | Pixel X | Δ Length | Delta Velocity | |
|---|---|---|---|---|
| **HUMMWV** | 30 m | 0.12 m | 1.22 m/s | 2.73 miles/hr |
| **HEMTT** | 30 m | 0.10 m | 0.19 m/s | 0.42 miles/hr |
| **M1070** | 45 m | 1.98m | 1.02 m/s | 2.28 miles/hr |

As a post processing tool, the sub-pixel algorithm increases the user's knowledge of sub-pixel moving targets. Figure 9.3 shows the bar chart of the calculated lengths. The targets naturally fall into three categories, which correspond to the specific vehicle lengths in the test.

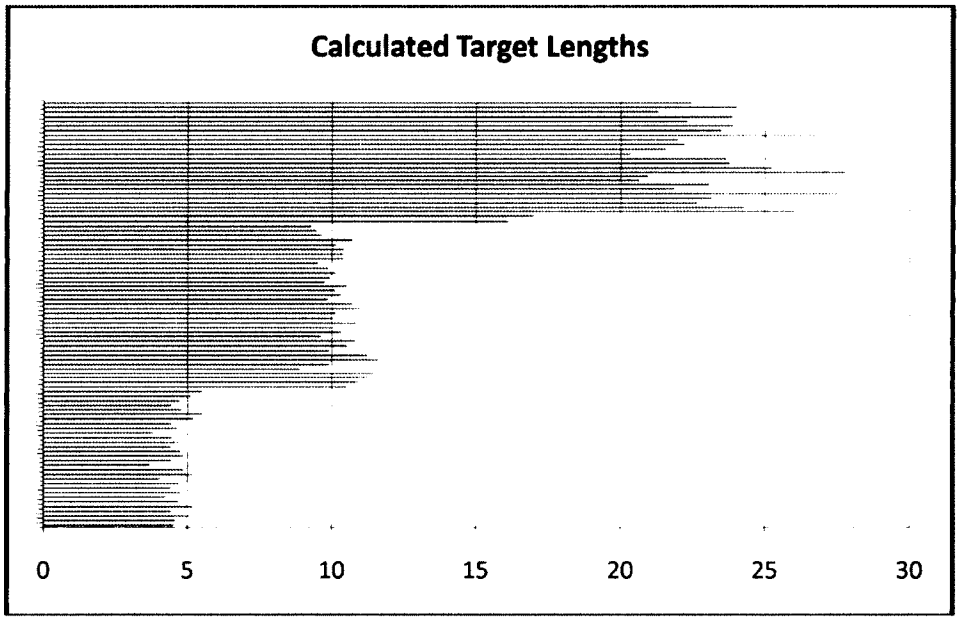



Figure 9.3 Bar Chart Showing the Calculated Target Lengths

However, the sub-pixel process is not accurate enough to distinguish between similar sized vehicles, such as the HEMTT, the M1070 tractor and the PLS truck, which range from 9.09 to 10.95 m in length. Further work in this area would focus on real-time processing techniques to automate the $N$ and $FAR$ parameter tuning.

## 9.3. Sensor Data Results

The scenarios chosen from the sensor datasets to analyze emulate different sensor and environmental effects to challenge the robustness of the sub-pixel algorithm. In the majority of the data sequences, the sensor image was shaky with unregistered frames, and the terrain was grassy and bumpy. Some scenes were shot with higher wind, causing the grass to sway, which added clutter noise to the pixel intensity. One data scene had a relatively stable sequence with minimal jitter. This scene provided the control dataset.

Using all of the sensor data scenarios, the algorithms were still able to detect and analyze the target. The results show the detection and the target analysis for selected scenarios with a brief description of the test case. Appendix B contains additional testing results.

The first set of scenarios evaluated the ability of the sub-pixel algorithm to detect a target within a variety of rectangular and square pixel shapes. The pixel shapes directly affect the target time in the frame. For horizontal travel, the size of the pixel in the y-direction was not as important factor as the size in the x-direction. A longer Pixel Y did reduce the overall delta intensity, but the detection and analysis methods continued to perform satisfactorily. Figure 9.4 and Figure 9.5 show the test results of changing the pixel size.

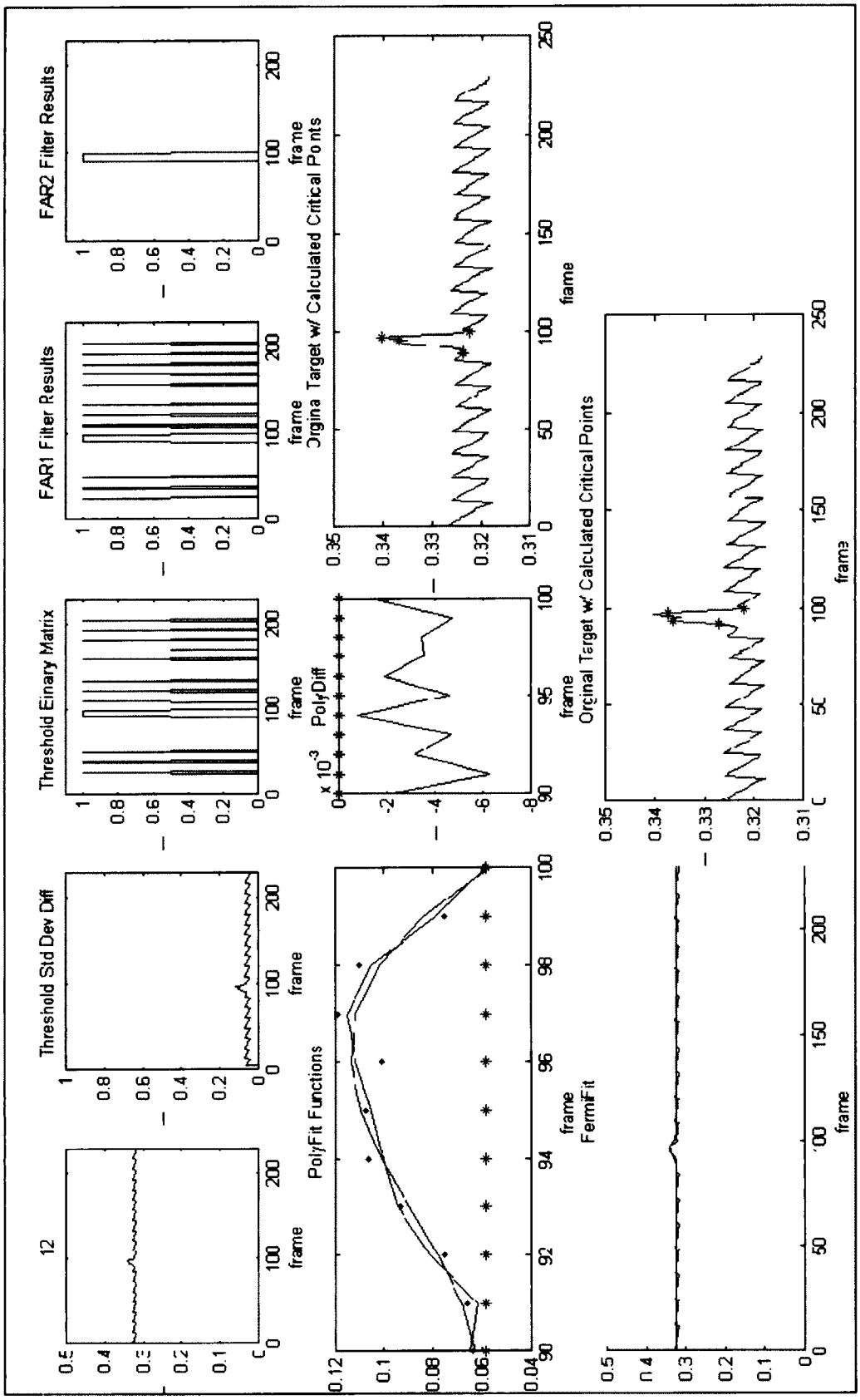



Figure 9.4 Analysis Results of Enlarging a Pixel by 0.5 in X and Y Directions

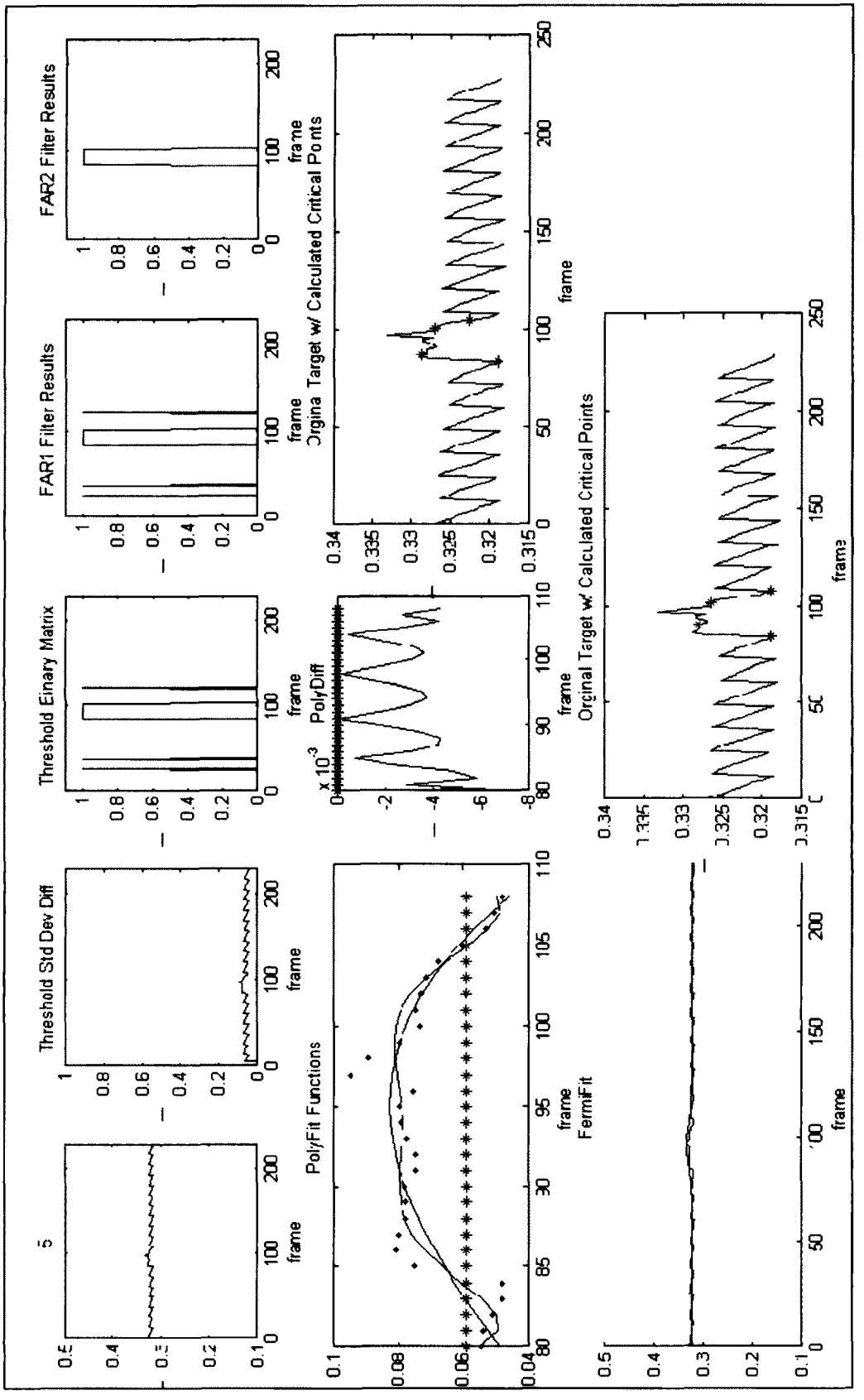



Figure 9.5 Analysis Results of Enlarging a Pixel by 1.0 in X and 0.5 in Y

The testing scenario enlarged the pixels in both the x- and y-direction as the original signal did not obtain valid results (Appendix B). Figure 9.4 enlarged the pixel by a half of a pixel in both directions, while Figure 9.5 enlarged the pixel only half a pixel in the y-direction and by a full pixel in the x-direction, providing a longer target profile. Consequently, the results in Figure 9.5 only detected 3 false alarms while the results in Figure 9.4, detected 11 false alarms. Figure 9.5 showed an improvement in the critical point calculation.

The algorithms were originally designed for darker background and lighter targets, i.e. infrared imagery. However, with visible images, this was not always the case. The pixels with a darker background, which were chosen in the tower's shadow, easily detected the lighter target (Figure.9.6). When testing a brightly lit background, detection was still possible by using the inverse binary matrix of the duration filter (Figure 9.7).

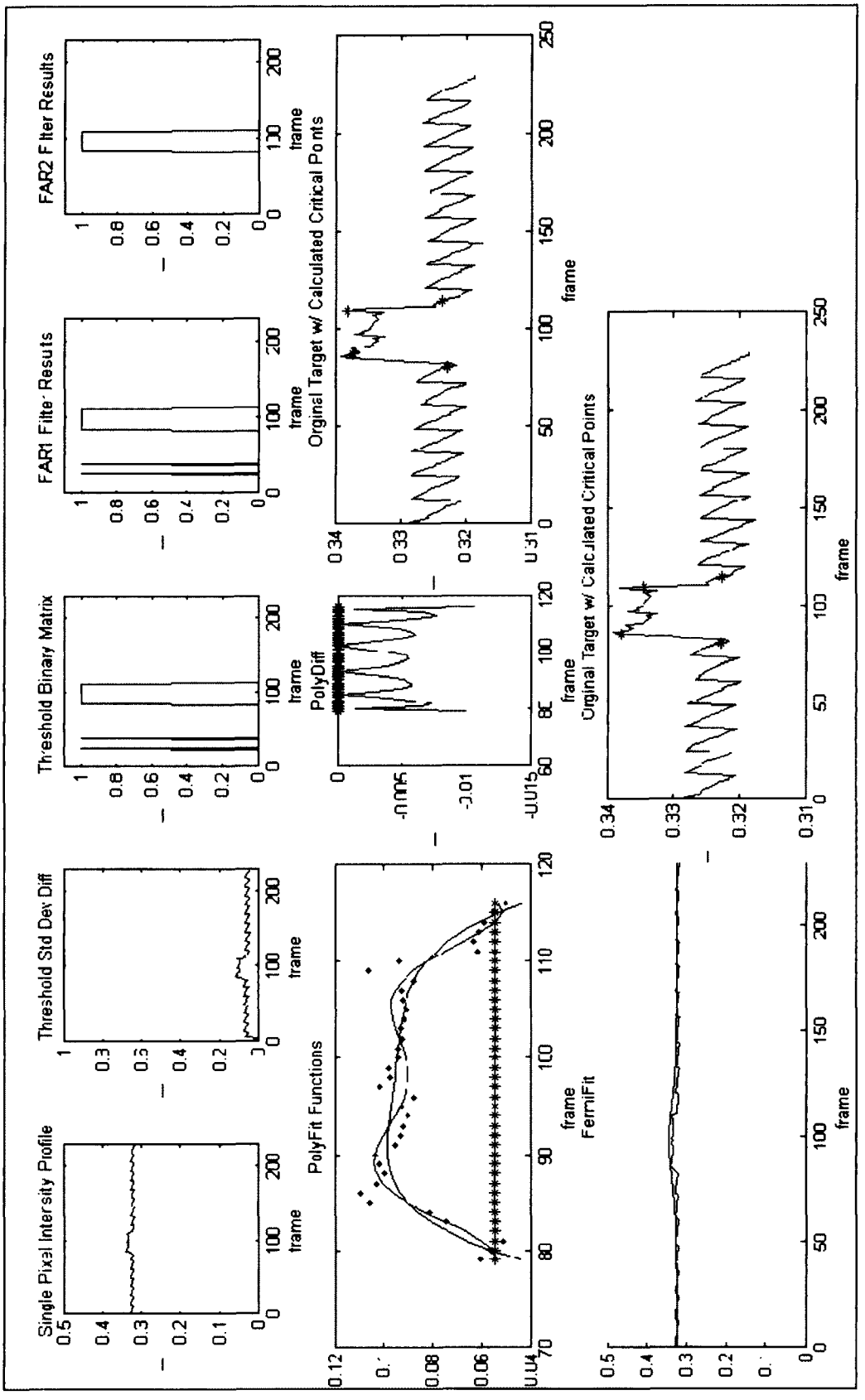



Figure.9.6 Analysis Results of Mobile Target with Shadowed Background

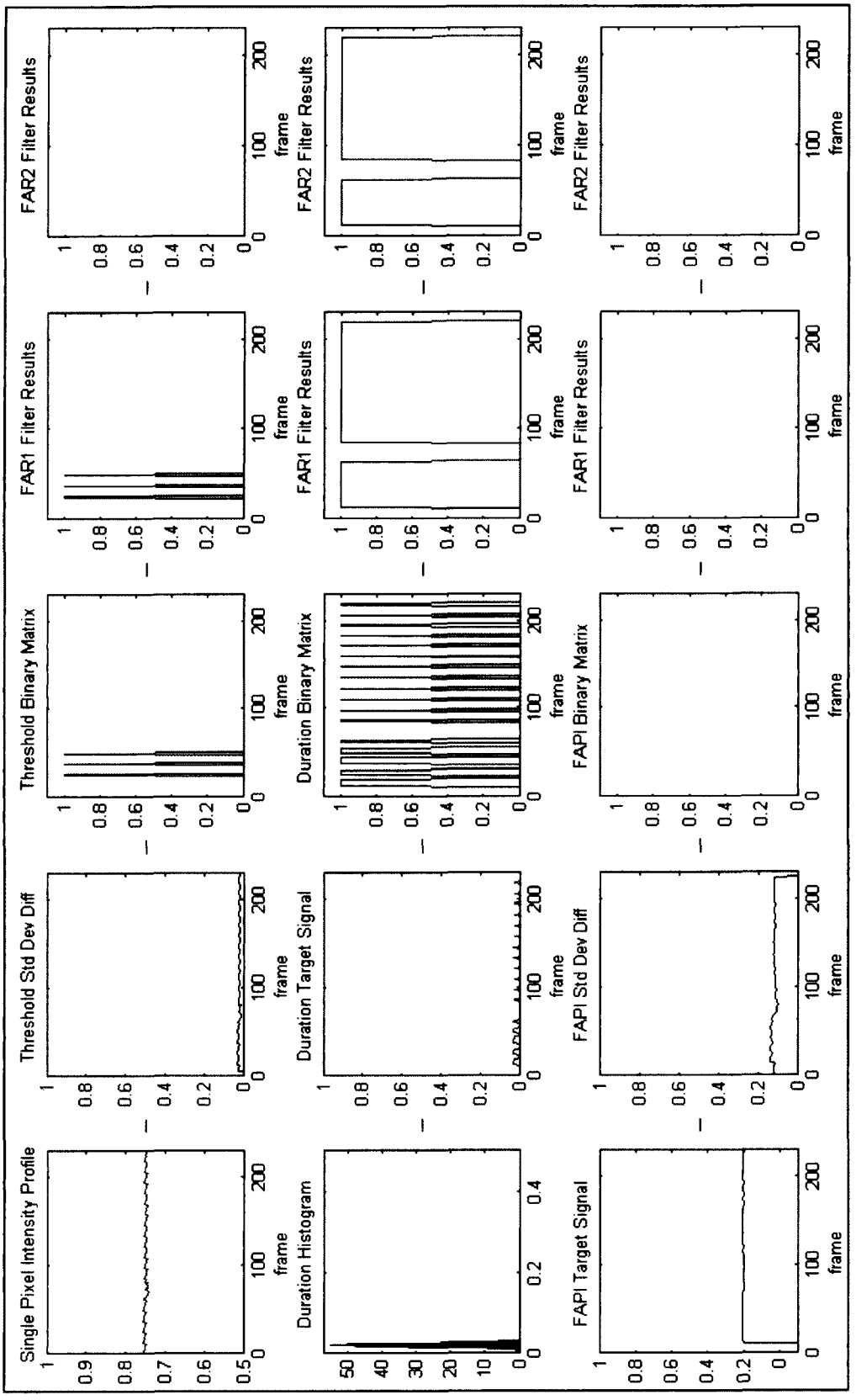



Figure 9.7 Analysis Results of Mobile Target in Bright Background

These results prompted a coding patch to incorporate the negative threshold and duration values. The desired results show effective detection in the threshold and duration algorithms in Figure 9.8.

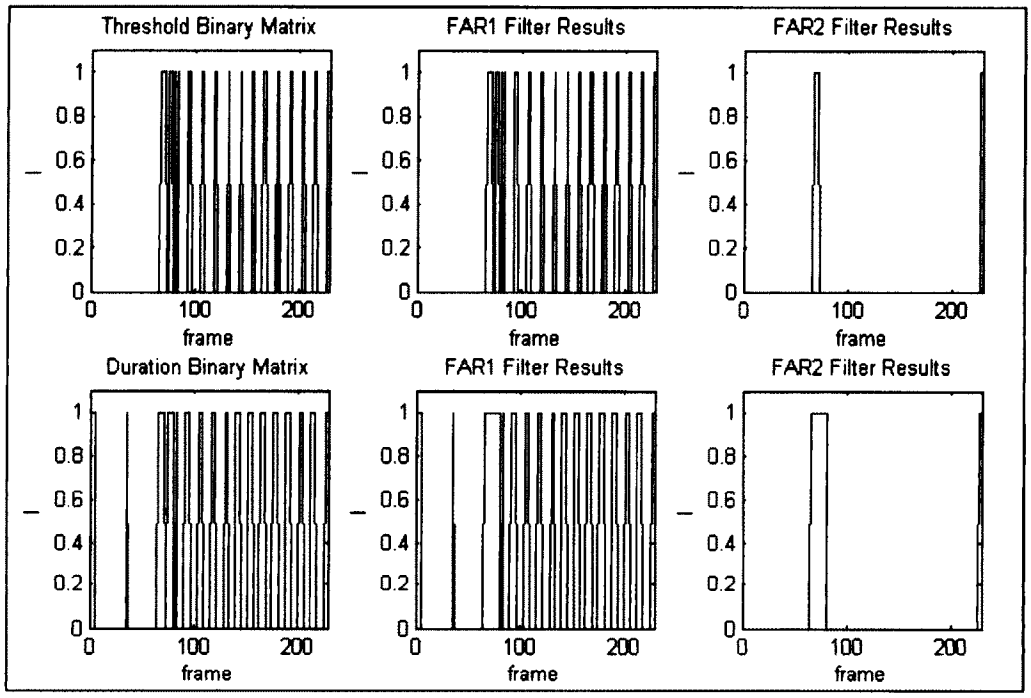



Figure 9.8 Detection in Bright Background Scenarios

The target analysis results for the bright background scenario are shown in Figure 9.9. Even though the FermiFit plot is the inverse of the actual signal, the critical points are still captured.

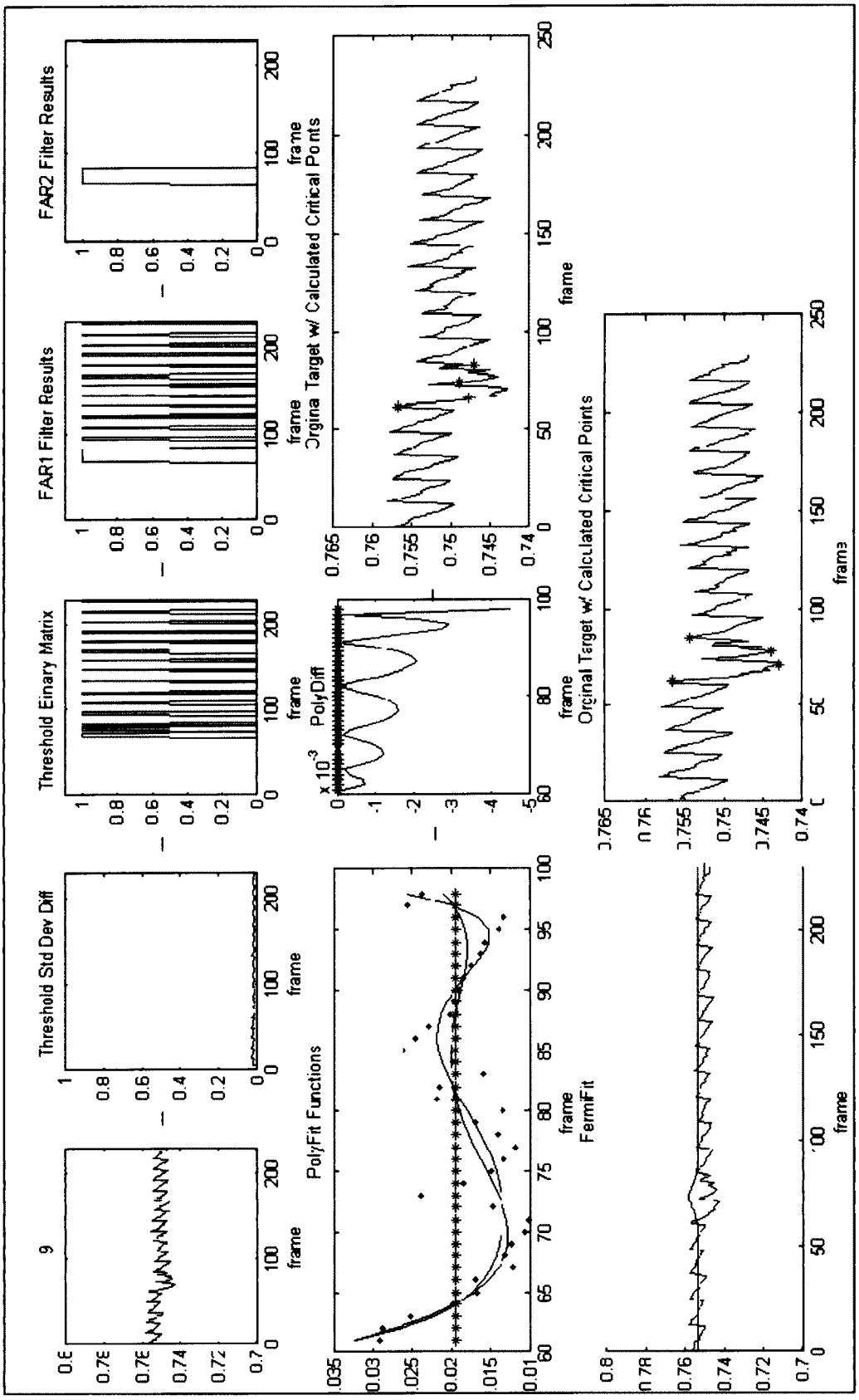



Figure 9.9 Analysis of Bright Background Scenarios

In the case of a pixel with both a light and dark background, the detection algorithm only detects a lighter target. However, the PolyFit algorithm was able to recover the whole target. It did require parameter modification to allow for a wider input range than the detected enter and exit points. In this case, the PolyFit analysis created an inversion instead of a trapezoid model, which is expected if it crosses a background intensity variation within a frame (Figure 9.10).

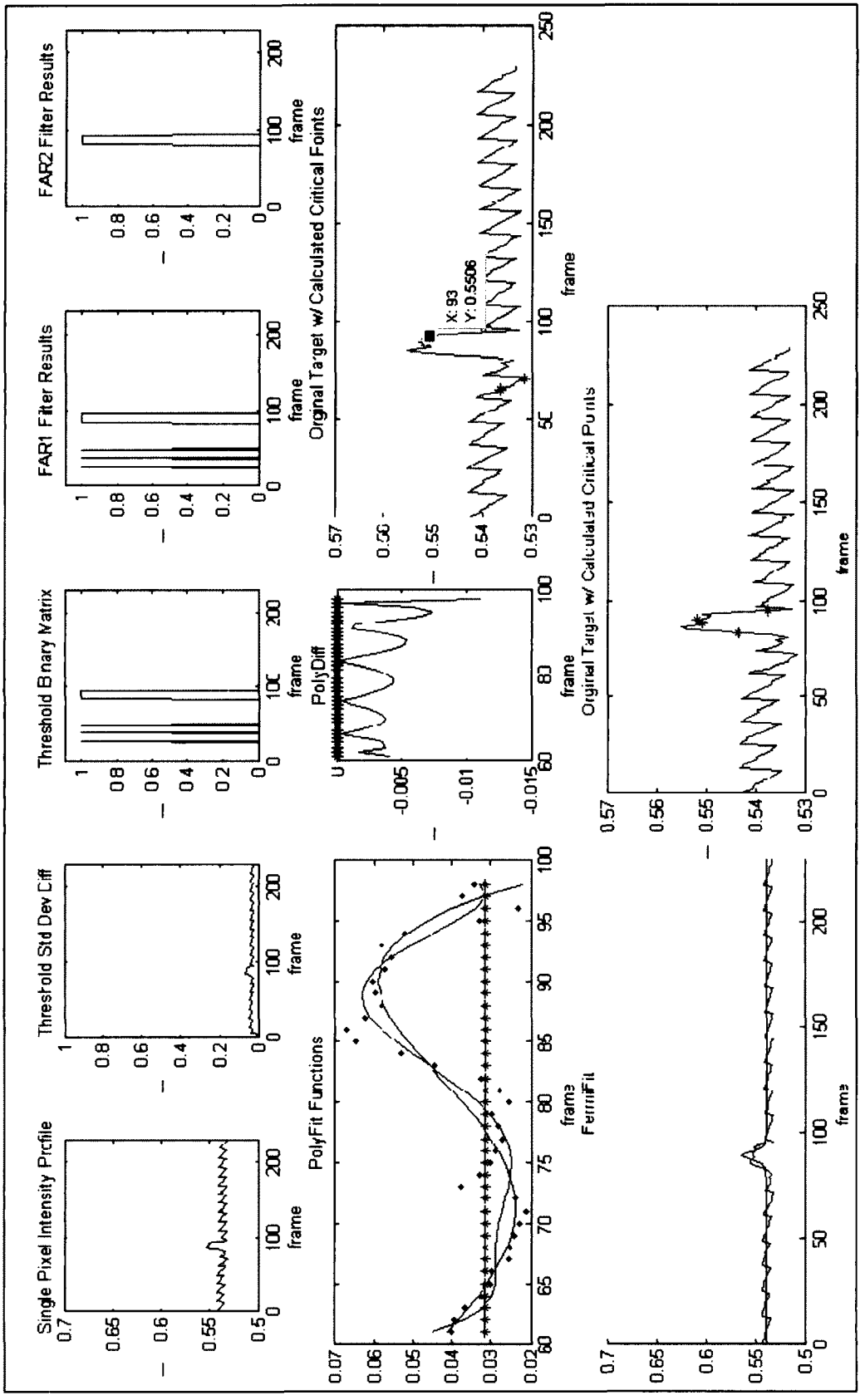



Figure 9.10 Analysis of Bright and Shadowed Background with Mobile Target

Initially a jittery or changing background during the frame sequence was not expected to produce results. However, the threshold and the duration methods did detect a target. The FAPI method had trouble detecting the target in these conditions. Conversely, the PolyFit and FermiFit were able to approximate the critical points fairly well. This situation is encountered in the next scenario that has a background intensity that significantly changes during the frame sequence. Figure 9.11 shows the Threshold Change Detection, the PolyFit analysis and the FermiFit analysis of this scenario.

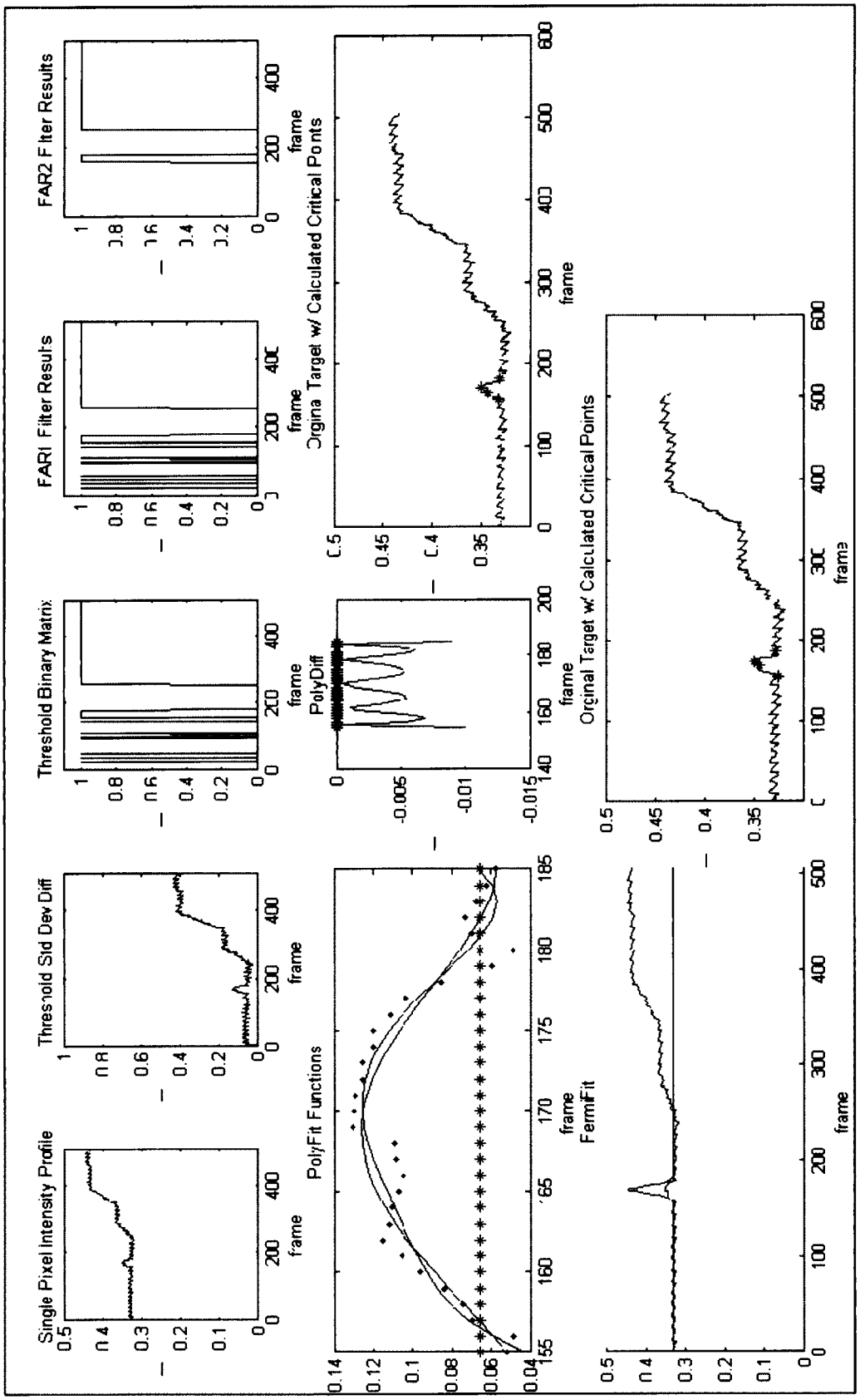



Figure 9.11 Analysis of Mobile Target in Changing Background Environment

Both the PolyFit and FermiFit algorithms performed very well in this challenging environment. Because there were enough initial background frames to establish the background model in the Threshold Change Detection, it was able to gather the target information, with only one false alarm due to the abrupt shift in the background intensity. There were enough low level background frames for the duration algorithm to work. The FAPI did not detect this target due to a convergence error (Figure 9.12).

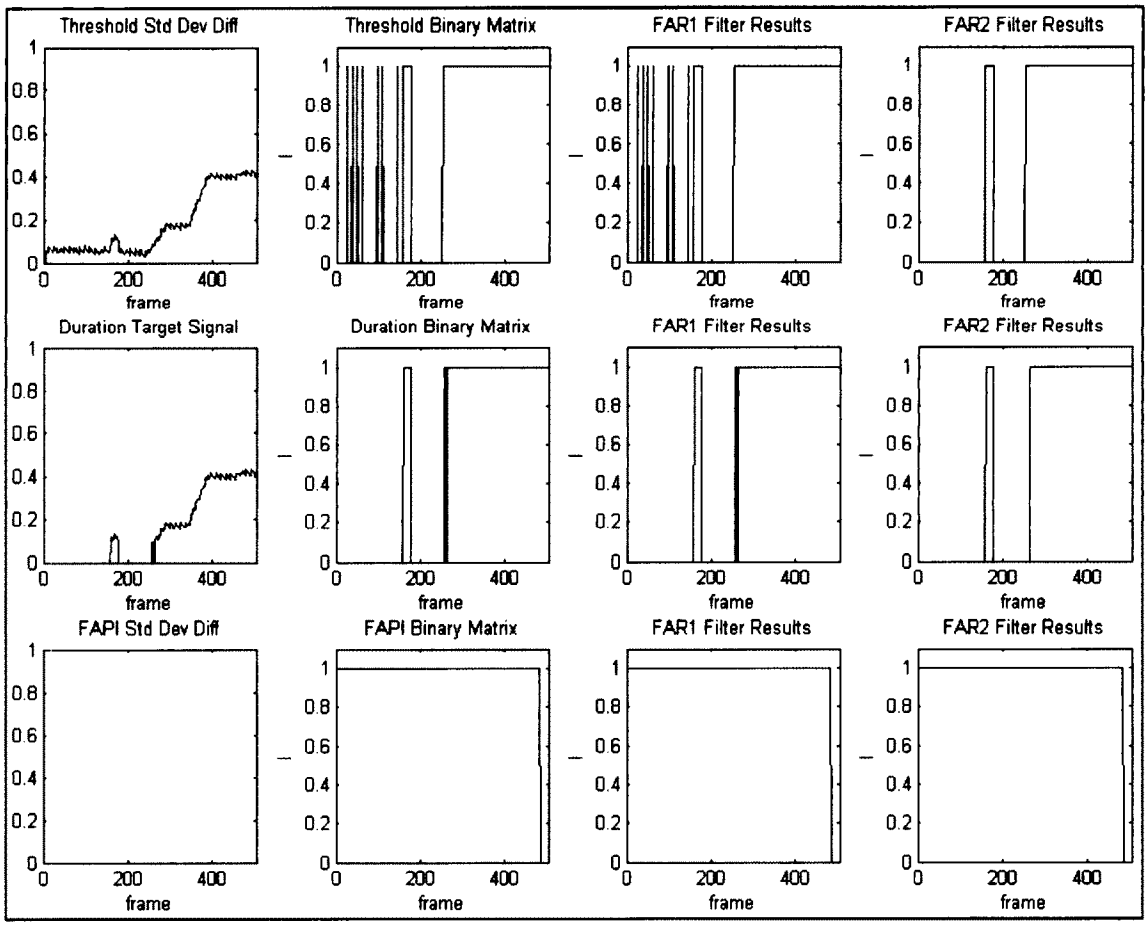



Figure 9.12 Detection of Mobile Target in Changing Background Environment

# CHAPTER TEN

# FURTHER WORK

Method analysis and testing results showed the feasibility of the sub-pixel algorithm for modeled and experimental situations. As with any research project, endless areas of exploration exist and all of them cannot be confined to a single dissertation.

The Refinement Through Tracking (RTT) methodology contained in this dissertation was theoretical in nature, but it is an imperative process needed to encompass the variety of target trajectories. Automating the target tracking, implementing the RTT algorithm into the sub-pixel algorithm, and testing the product is a project currently on the books.

Another area of research is to make the change detection algorithm more robust by investigating different initialization weighting matrices for the FAPI algorithm that provide more convergence reliability and to perform more comparisons of other change detection methods.

As the PolyFit and FermiFit analysis methods provide an estimation of target velocity, a refinement algorithm incorporating a Velocity Matched Filter technique might prove a feasible way to significantly improve target signal intensities [54]. Also, this analysis used a rectangular pixel as a basis for the calculations. Further work will investigate other pixel geometries, multiple targets and the algorithm adaptation.

A major objective of research is to end up with a marketable product. Two Graphical User Interfaces (GUIs) incorporating the sub-pixel algorithms were designed as basic prototypes for end user application testing. The next step involves field testing these designs with a variety of classified datasets containing known targets and parameters. Future works in application implementation are exploring parallel processing capabilities and near-real-time execution.

# CHAPTER ELEVEN

# CONCLUSION

Overall, the process, techniques, and methods presented in this dissertation are viable. Independent algorithm verification tested the quality and repeatability of the coding design.

The model data proved the concept works, through the various series of tests. The tests also indicated how flexible the process is. With slight adjustments, the parameters of the environment can be adjusted to fine tune the detection algorithms, thus lowering the false alarm rate. These adjustments can be done real-time, as needed.

The experimental tests validated the process. Because of the limited data availability, the datasets were taken in a controlled setting from a sensor positioned 18 meters in altitude. The sensor took videos of a single target, which was a remotely controlled car. However, since the process has proven to be so adaptable, the parameters can be adjustable and a user with knowledge of their sensor can apply this process and techniques to yield spectacular results. Due to the modularity of the sub-pixel detection and characterization process, as sensors change and detection algorithms evolve, the parameters and new algorithms can be incorporated easily. In essence, this methodology is an open and adaptable system.

Applying this technique correctly, should allow the intelligence community to process copious amounts of data more efficiently and leverage more capability from existing legacy systems with lower resolution. As an out-of-the-box solution, agencies can take existing higher resolution systems and elevate them, thus expanding the imagery capture area and lowering their resolution intentionally to detect moving targets that otherwise would be outside of their narrower field of view.

# APPENDIX A

# ALGORITHM VERIFICATION RESULTS

Table A.1 Algorithm Verification Results

| **Algorithm File** | **Tests** | **Results** | **Method** |
|---|---|---|---|
| **Target Modeling** | | | |
| InfoSensor | - Get sensor information | Verified | Test code |
| InfoTarget | - Get target information | Verified | Test code |
| DBTargetTrapP | - Plot trapezoid<br>- Incorporate background<br>- Different noise values<br>- Different jitter values<br>- Different target intensities<br>- Different background intensities<br>- Different frame total values<br>- Different frame start times | Verified | Test code,<br>plots,<br>hand calcs |
| DBTargetTrapMultiP | - Plot trapezoid<br>- Incorporate background<br>- Insert signal in multiple-pixels<br>- Plot multiple-pixels<br>- Different background intensities<br>- Different frame total values<br>- Different frame start times<br>- Check target travel in pixels | Verified | Test code,<br>plots |
| ExtractNoNoise | - Check $Enter_1$ calculation<br>- Check $Exit_2$ calculation<br>- Plot results | Verified | Test code,<br>plots,<br>hand calcs |
| AddNoise | - Add noise to single-pixel<br>- Add noise to multiple-pixel<br>- Add noise to target signal<br>- Test Gaussian profile<br>- Test random functions<br>- Plot results | Verified | Test code,<br>plots,<br>hand calcs |
| AddJitter | - Jitter complex background<br>- Test with inserted signal<br>- Jitter less than 1 pixel<br>- Jitter more than 1 pixel<br>- Plot results | Verified | Test code,<br>plots,<br>hand calcs |
| **Target Detection** | | | |
| DetectDuration | - Vary N values<br>- Test histogram function<br>- Test filter calculation<br>- Test IDuration<br>- Test TargetBinary<br>- Test repeatability<br>- Plot results | Verified | Test code,<br>plots |

| | | | |
|---|---|---|---|
| DetectThreshold | - Vary N values<br>- Vary gamma values<br>- Test threshold calculation<br>- Test ZetaPixel calculation<br>- Test PixelDelta<br>- Test TargetBinary<br>- Plot Results | Verified | Test code, plots |
| DetectFAPI | - Test Vectorization<br>- Test Badeau's initialization matrix<br>- Test Simonson's initialization matrix<br>- Test with complex background<br>- Test FAPIPixel<br>- Test XBack<br>- Plot Results | Verified | Test code, plots |
| DetectFAR1 | - Test different F1 values<br>- Plot BinaryMatrixNew | Verified | Test code, plots |
| DetectFAR2 | - Test different F2 values<br>- Plot BinaryMatrixNew | Verified | Test code, plots |
| NumberTargets | - Test $Enter_1$ calculation<br>- Test $Exit_2$ calculation<br>- Test # targets calculated<br>- Print results | Verified | Test code |
| **Target Analysis** | | | |
| AnalyzeFermiFit | - Test parameter calculations<br>- Test method to choose c<br>- Test $Enter_1$ calculation<br>- Test $Enter_2$ calculation<br>- Test $Exit_1$ calculation<br>- Test $Exit_2$ calculation<br>- Test repeatability<br>- Plot results | Verified | Test code, plots |
| AnalyzePolyFit | - Test PolyFit functions<br>- Test findpeaks functions<br>- Test intersection calculations<br>- Test $Enter_1$ calculation<br>- Test $Enter_2$ calculation<br>- Test $Exit_1$ calculation<br>- Test $Exit_2$ calculation<br>- Plot results | Verified | Test code, plots |

# APPENDIX B

# TESTING TABLES

# Table B.1 Detection Comparison: Sensor and Target Information

| S Info | S Rate | Pixel X | Pixel Y | Noise | Jitter | T Info | # | Length | Width | Speed | Ibk | ITgt | No Noise | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 30 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 21.936 | 0.2 | 0.9 | | 60 | 74 | 101 | 115 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 12.655 | 0.2 | 0.9 | | 60 | 82 | 131 | 153 |
| | 30 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 18.929 | 0.2 | 0.9 | | 60 | 91 | 108 | 139 |
| | 30 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 22.001 | 0.2 | 0.9 | | 60 | 66 | 101 | 107 |
| | 30 | 30 | 30 | 0 | 0 | | 6 | 8.32 | 2.43 | 13.812 | 0.2 | 0.9 | | 60 | 78 | 125 | 143 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 12.626 | 0.2 | 0.9 | | 60 | 82 | 131 | 153 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 14.665 | 0.2 | 0.9 | | 60 | 79 | 121 | 140 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 17.616 | 0.2 | 0.9 | | 60 | 75 | 111 | 126 |
| | 30 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 22.541 | 0.2 | 0.9 | | 60 | 74 | 100 | 114 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 13.325 | 0.2 | 0.9 | | 60 | 80 | 128 | 148 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 15.975 | 0.2 | 0.9 | | 60 | 77 | 116 | 133 |
| | 30 | 30 | 30 | 0 | 0 | | 3 | 24.89 | 3.68 | 18.333 | 0.2 | 0.9 | | 60 | 101 | 109 | 150 |
| | 30 | 30 | 30 | 0 | 0 | | 3 | 24.89 | 3.68 | 14.423 | 0.2 | 0.9 | | 60 | 112 | 122 | 174 |
| | 30 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 19.008 | 0.2 | 0.9 | | 60 | 76 | 107 | 123 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 10.872 | 0.2 | 0.9 | | 60 | 85 | 143 | 168 |
| | 30 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 16.318 | 0.2 | 0.9 | | 60 | 95 | 115 | 150 |
| | 30 | 30 | 30 | 0 | 0 | | 6 | 8.32 | 2.43 | 22.125 | 0.2 | 0.9 | | 60 | 71 | 101 | 112 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 13.464 | 0.2 | 0.9 | | 60 | 80 | 127 | 147 |
| | 30 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 21.411 | 0.2 | 0.9 | | 60 | 74 | 102 | 116 |
| | 30 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 12.339 | 0.2 | 0.9 | | 60 | 82 | 133 | 155 |
| | 30 | 60 | 60 | 0 | 0 | | 3 | 24.89 | 3.68 | 13.117 | 0.2 | 0.9 | | 60 | 117 | 197 | 254 |
| | 30 | 60 | 60 | 0 | 0 | | 6 | 8.32 | 2.43 | 25.477 | 0.2 | 0.9 | | 60 | 70 | 131 | 141 |
| | 30 | 60 | 60 | 0 | 0 | | 6 | 8.32 | 2.43 | 23.354 | 0.2 | 0.9 | | 60 | 71 | 137 | 148 |
| | 30 | 60 | 60 | 0 | 0 | | 1 | 4.57 | 2.16 | 20.849 | 0.2 | 0.9 | | 60 | 67 | 146 | 153 |
| | 30 | 60 | 60 | 0 | 0 | | 5 | 19.27 | 2.44 | 18.638 | 0.2 | 0.9 | | 60 | 91 | 157 | 188 |
| | 30 | 60 | 60 | 0 | 0 | | 4 | 9.08 | 2.59 | 11.133 | 0.2 | 0.9 | | 60 | 84 | 222 | 246 |
| | 30 | 60 | 60 | 0 | 0 | | 5 | 19.27 | 2.44 | 22.227 | 0.2 | 0.9 | | 60 | 86 | 141 | 167 |
| | 30 | 60 | 60 | 0 | 0 | | 4 | 9.08 | 2.59 | 17.759 | 0.2 | 0.9 | | 60 | 75 | 161 | 176 |
| | 30 | 60 | 60 | 0 | 0 | | 4 | 9.08 | 2.59 | 14.366 | 0.2 | 0.9 | | 60 | 79 | 185 | 204 |
| | 30 | 60 | 60 | 0 | 0 | | 6 | 8.32 | 2.43 | 15.607 | 0.2 | 0.9 | | 60 | 76 | 175 | 191 |
| | 30 | 60 | 60 | 0 | 0 | | 3 | 24.89 | 3.68 | 19.421 | 0.2 | 0.9 | | 60 | 98 | 153 | 191 |
| | 30 | 60 | 60 | 0 | 0 | | 1 | 4.57 | 2.16 | 17.418 | 0.2 | 0.9 | | 60 | 68 | 163 | 171 |
| | 30 | 60 | 60 | 0 | 0 | | 2 | 10.19 | 2.44 | 12.83 | 0.2 | 0.9 | | 60 | 84 | 200 | 224 |
| | 30 | 60 | 60 | 0 | 0 | | 4 | 9.08 | 2.59 | 18.001 | 0.2 | 0.9 | | 60 | 75 | 160 | 175 |
| | 30 | 60 | 60 | 0 | 0 | | 5 | 19.27 | 2.44 | 23.077 | 0.2 | 0.9 | | 60 | 85 | 138 | 163 |
| | 30 | 60 | 60 | 0 | 0 | | 4 | 9.08 | 2.59 | 15.397 | 0.2 | 0.9 | | 60 | 78 | 177 | 195 |
| | 30 | 60 | 60 | 0 | 0 | | 6 | 8.32 | 2.43 | 16.373 | 0.2 | 0.9 | | 60 | 75 | 170 | 185 |
| | 30 | 60 | 60 | 0 | 0 | | 3 | 24.89 | 3.68 | 19.856 | 0.2 | 0.9 | | 60 | 98 | 151 | 189 |
| | 30 | 60 | 60 | 0 | 0 | | 4 | 9.08 | 2.59 | 17.064 | 0.2 | 0.9 | | 60 | 76 | 165 | 181 |
| | 30 | 60 | 60 | 0 | 0 | | 4 | 9.08 | 2.59 | 14.865 | 0.2 | 0.9 | | 60 | 78 | 181 | 199 |
| | 10 | 30 | 30 | 0 | 0 | | 3 | 24.89 | 3.68 | 14.37 | 0.2 | 0.9 | | 60 | 77 | 81 | 98 |
| | 10 | 30 | 30 | 0 | 0 | | 6 | 8.32 | 2.43 | 19.283 | 0.2 | 0.9 | | 60 | 64 | 76 | 80 |
| | 10 | 30 | 30 | 0 | 0 | | 3 | 24.89 | 3.68 | 15.055 | 0.2 | 0.9 | | 60 | 77 | 80 | 97 |
| | 10 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 16.853 | 0.2 | 0.9 | | 60 | 66 | 78 | 84 |
| | 10 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 16.803 | 0.2 | 0.9 | | 60 | 66 | 78 | 84 |
| | 10 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 22.084 | 0.2 | 0.9 | | 60 | 62 | 74 | 76 |
| | 10 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 16.392 | 0.2 | 0.9 | | 60 | 66 | 78 | 84 |
| | 10 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 13.993 | 0.2 | 0.9 | | 60 | 74 | 81 | 95 |
| | 10 | 30 | 30 | 0 | 0 | | 6 | 8.32 | 2.43 | 24.51 | 0.2 | 0.9 | | 60 | 63 | 72 | 75 |
| | 10 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 14.308 | 0.2 | 0.9 | | 60 | 66 | 81 | 87 |
| | 10 | 30 | 30 | 0 | 0 | | 3 | 24.89 | 3.68 | 16.569 | 0.2 | 0.9 | | 60 | 75 | 78 | 93 |
| | 10 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 20.33 | 0.2 | 0.9 | | 60 | 69 | 75 | 84 |
| | 10 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 14.605 | 0.2 | 0.9 | | 60 | 67 | 81 | 88 |
| | 10 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 16.342 | 0.2 | 0.9 | | 60 | 66 | 78 | 84 |
| | 10 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 23.762 | 0.2 | 0.9 | | 60 | 62 | 73 | 75 |
| | 10 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 20.448 | 0.2 | 0.9 | | 60 | 62 | 75 | 77 |
| | 10 | 30 | 30 | 0 | 0 | | 3 | 24.89 | 3.68 | 18.717 | 0.2 | 0.9 | | 60 | 73 | 76 | 89 |
| | 10 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 17.69 | 0.2 | 0.9 | | 60 | 66 | 77 | 83 |
| | 10 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 27.799 | 0.2 | 0.9 | | 60 | 62 | 71 | 73 |
| | 10 | 30 | 30 | 0 | 0 | | 6 | 8.32 | 2.43 | 22.424 | 0.2 | 0.9 | | 60 | 64 | 73 | 77 |
| | 30 | 30 | 30 | 0.05 | 0 | | 6 | 8.32 | 2.43 | 20.131 | 0.2 | 0.9 | | 60 | 72 | 105 | 117 |
| | 30 | 30 | 30 | 0.05 | 0 | | 3 | 24.89 | 3.68 | 16.712 | 0.2 | 0.9 | | 60 | 105 | 114 | 159 |
| | 30 | 30 | 30 | 0.05 | 0 | | 5 | 19.27 | 2.44 | 12.954 | 0.2 | 0.9 | | 60 | 105 | 129 | 174 |
| | 30 | 30 | 30 | 0.05 | 0 | | 5 | 19.27 | 2.44 | 15.809 | 0.2 | 0.9 | | 60 | 97 | 117 | 154 |
| | 30 | 30 | 30 | 0.05 | 0 | | 6 | 8.32 | 2.43 | 20.999 | 0.2 | 0.9 | | 60 | 72 | 103 | 115 |
| | 30 | 30 | 30 | 0.05 | 0 | | 2 | 10.19 | 2.44 | 13.408 | 0.2 | 0.9 | | 60 | 83 | 127 | 150 |
| | 30 | 30 | 30 | 0.05 | 0 | | 6 | 8.32 | 2.43 | 16.728 | 0.2 | 0.9 | | 60 | 75 | 114 | 129 |
| | 30 | 30 | 30 | 0.05 | 0 | | 2 | 10.19 | 2.44 | 17.452 | 0.2 | 0.9 | | 60 | 78 | 112 | 130 |
| | 30 | 30 | 30 | 0.05 | 0 | | 5 | 19.27 | 2.44 | 17.285 | 0.2 | 0.9 | | 60 | 93 | 112 | 145 |
| | 30 | 30 | 30 | 0.05 | 0 | | 1 | 4.57 | 2.16 | 18.628 | 0.2 | 0.9 | | 60 | 67 | 108 | 115 |
| | 30 | 30 | 30 | 0.05 | 0 | | 3 | 24.89 | 3.68 | 12.769 | 0.2 | 0.9 | | 60 | 118 | 130 | 188 |
| | 30 | 30 | 30 | 0.05 | 0 | | 2 | 10.19 | 2.44 | 19.735 | 0.2 | 0.9 | | 60 | 75 | 106 | 121 |
| | 30 | 30 | 30 | 0.05 | 0 | | 4 | 9.08 | 2.59 | 12.369 | 0.2 | 0.9 | | 60 | 82 | 133 | 155 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 11.633 | 0.2 | 0.9 | 60 | 124 | 137 | 201 |
| 30 | 30 | 30 | 0.05 | 0 | 5 | 19.27 | 2.44 | 19.513 | 0.2 | 0.9 | 60 | 90 | 106 | 136 |
| 30 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 12.449 | 0.2 | 0.9 | 60 | 120 | 132 | 192 |
| 30 | 30 | 30 | 0.05 | 0 | 5 | 19.27 | 2.44 | 17.056 | 0.2 | 0.9 | 60 | 94 | 113 | 147 |
| 30 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 15.276 | 0.2 | 0.9 | 60 | 80 | 119 | 139 |
| 30 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 15.422 | 0.2 | 0.9 | 60 | 108 | 118 | 166 |
| 60 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 22.394 | 0.2 | 0.9 | 60 | 72 | 140 | 152 |
| 60 | 30 | 30 | 0.05 | 0 | 5 | 19.27 | 2.44 | 12.759 | 0.2 | 0.9 | 60 | 151 | 201 | 292 |
| 60 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 14.935 | 0.2 | 0.9 | 60 | 78 | 181 | 199 |
| 60 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 15.999 | 0.2 | 0.9 | 60 | 153 | 173 | 266 |
| 60 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 12.477 | 0.2 | 0.9 | 60 | 180 | 204 | 324 |
| 60 | 30 | 30 | 0.05 | 0 | 5 | 19.27 | 2.44 | 22.264 | 0.2 | 0.9 | 60 | 112 | 141 | 193 |
| 60 | 30 | 30 | 0.05 | 0 | 4 | 9.08 | 2.59 | 13.372 | 0.2 | 0.9 | 60 | 101 | 195 | 236 |
| 60 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 26.649 | 0.2 | 0.9 | 60 | 70 | 128 | 138 |
| 60 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 21.553 | 0.2 | 0.9 | 60 | 73 | 144 | 157 |
| 60 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 21.196 | 0.2 | 0.9 | 60 | 89 | 145 | 174 |
| 60 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 26.606 | 0.2 | 0.9 | 60 | 70 | 128 | 138 |
| 60 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 25.257 | 0.2 | 0.9 | 60 | 80 | 131 | 151 |
| 60 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 23.748 | 0.2 | 0.9 | 60 | 81 | 136 | 157 |
| 60 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 22.864 | 0.2 | 0.9 | 60 | 82 | 139 | 161 |
| 60 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 16.822 | 0.2 | 0.9 | 60 | 96 | 167 | 203 |
| 60 | 30 | 30 | 0.05 | 0 | 4 | 9.08 | 2.59 | 16.231 | 0.2 | 0.9 | 60 | 94 | 171 | 205 |
| 60 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 22.864 | 0.2 | 0.9 | 60 | 87 | 139 | 166 |
| 60 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 19.418 | 0.2 | 0.9 | 60 | 74 | 153 | 167 |
| 60 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 22.81 | 0.2 | 0.9 | 60 | 72 | 139 | 151 |
| 60 | 30 | 30 | 0.05 | 0 | 5 | 19.27 | 2.44 | 13.223 | 0.2 | 0.9 | 60 | 147 | 196 | 283 |
| 120 | 30 | 30 | 0.05 | 0 | 5 | 19.27 | 2.44 | 24.435 | 0.2 | 0.9 | 60 | 155 | 207 | 302 |
| 120 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 24.663 | 0.2 | 0.9 | 60 | 110 | 206 | 256 |
| 120 | 30 | 30 | 0.05 | 0 | 4 | 9.08 | 2.59 | 12.376 | 0.2 | 0.9 | 60 | 148 | 351 | 439 |
| 120 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 18.902 | 0.2 | 0.9 | 60 | 113 | 250 | 303 |
| 120 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 14.94 | 0.2 | 0.9 | 60 | 142 | 301 | 383 |
| 120 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 17.309 | 0.2 | 0.9 | 60 | 118 | 268 | 326 |
| 120 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 15.263 | 0.2 | 0.9 | 60 | 140 | 296 | 376 |
| 120 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 16.996 | 0.2 | 0.9 | 60 | 132 | 272 | 344 |
| 120 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 25.769 | 0.2 | 0.9 | 60 | 81 | 200 | 221 |
| 120 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 16.751 | 0.2 | 0.9 | 60 | 238 | 275 | 453 |
| 120 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 11.261 | 0.2 | 0.9 | 60 | 325 | 380 | 0 |
| 120 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 15.66 | 0.2 | 0.9 | 60 | 124 | 290 | 354 |
| 120 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 16.278 | 0.2 | 0.9 | 60 | 243 | 281 | 464 |
| 120 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 16.205 | 0.2 | 0.9 | 60 | 244 | 282 | 466 |
| 120 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 13.861 | 0.2 | 0.9 | 60 | 275 | 320 | 535 |
| 120 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 17.327 | 0.2 | 0.9 | 60 | 232 | 268 | 440 |
| 120 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 20.934 | 0.2 | 0.9 | 60 | 108 | 232 | 280 |
| 120 | 30 | 30 | 0.05 | 0 | 5 | 19.27 | 2.44 | 14.693 | 0.2 | 0.9 | 60 | 217 | 305 | 462 |
| 120 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 22.832 | 0.2 | 0.9 | 60 | 84 | 218 | 242 |
| 120 | 30 | 30 | 0.05 | 0 | 4 | 9.08 | 2.59 | 18.208 | 0.2 | 0.9 | 60 | 120 | 258 | 318 |
| 120 | 30 | 30 | 0.05 | 0 | 3 | 24.89 | 3.68 | 19.308 | 0.2 | 0.9 | 60 | 215 | 246 | 401 |
| 120 | 30 | 30 | 0.05 | 0 | 1 | 4.57 | 2.16 | 24.417 | 0.2 | 0.9 | 60 | 82 | 207 | 229 |
| 120 | 30 | 30 | 0.05 | 0 | 2 | 10.19 | 2.44 | 13.256 | 0.2 | 0.9 | 60 | 152 | 332 | 424 |
| 120 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 25.447 | 0.2 | 0.9 | 60 | 99 | 201 | 240 |
| 120 | 30 | 30 | 0.05 | 0 | 6 | 8.32 | 2.43 | 23.108 | 0.2 | 0.9 | 60 | 103 | 216 | 259 |
| 120 | 30 | 30 | 0.04 | 0 | 4 | 9.08 | 2.59 | 12.229 | 0.2 | 0.9 | 60 | 149 | 354 | 443 |
| 120 | 30 | 30 | 0.04 | 0 | 6 | 8.32 | 2.43 | 15.647 | 0.2 | 0.9 | 60 | 124 | 290 | 354 |
| 120 | 30 | 30 | 0.04 | 0 | 2 | 10.19 | 2.44 | 17.62 | 0.2 | 0.9 | 60 | 129 | 264 | 333 |
| 120 | 30 | 30 | 0.04 | 0 | 2 | 10.19 | 2.44 | 15.45 | 0.2 | 0.9 | 60 | 139 | 293 | 372 |
| 120 | 30 | 30 | 0.04 | 0 | 3 | 24.89 | 3.68 | 15.014 | 0.2 | 0.9 | 60 | 259 | 300 | 499 |
| 120 | 30 | 30 | 0.04 | 0 | 3 | 24.89 | 3.68 | 14.135 | 0.2 | 0.9 | 60 | 271 | 315 | 526 |
| 120 | 30 | 30 | 0.04 | 0 | 3 | 24.89 | 3.68 | 15.081 | 0.2 | 0.9 | 60 | 258 | 299 | 497 |
| 120 | 30 | 30 | 0.04 | 0 | 4 | 9.08 | 2.59 | 10.196 | 0.2 | 0.9 | 60 | 167 | 413 | 520 |
| 120 | 30 | 30 | 0.04 | 0 | 2 | 10.19 | 2.44 | 22.05 | 0.2 | 0.9 | 60 | 115 | 223 | 278 |
| 120 | 30 | 30 | 0.04 | 0 | 1 | 4.57 | 2.16 | 27.472 | 0.2 | 0.9 | 60 | 80 | 191 | 211 |
| 120 | 30 | 30 | 0.04 | 0 | 3 | 24.89 | 3.68 | 14.54 | 0.2 | 0.9 | 60 | 265 | 308 | 513 |
| 120 | 30 | 30 | 0.04 | 0 | 3 | 24.89 | 3.68 | 12.338 | 0.2 | 0.9 | 60 | 302 | 352 | 594 |
| 120 | 30 | 30 | 0.04 | 0 | 5 | 19.27 | 2.44 | 23.078 | 0.2 | 0.9 | 60 | 160 | 216 | 316 |
| 120 | 30 | 30 | 0.04 | 0 | 4 | 9.08 | 2.59 | 13.44 | 0.2 | 0.9 | 60 | 141 | 328 | 409 |
| 120 | 30 | 30 | 0.04 | 0 | 6 | 8.32 | 2.43 | 13.741 | 0.2 | 0.9 | 60 | 133 | 322 | 395 |
| 120 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 12.51 | 0.2 | 0.9 | 60 | 299 | 348 | 587 |
| 120 | 30 | 30 | 0.02 | 0 | 2 | 10.19 | 2.44 | 19.876 | 0.2 | 0.9 | 60 | 122 | 241 | 303 |
| 120 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 12.933 | 0.2 | 0.9 | 60 | 291 | 338 | 569 |
| 120 | 30 | 30 | 0.02 | 0 | 4 | 9.08 | 2.59 | 19.527 | 0.2 | 0.9 | 60 | 116 | 244 | 300 |
| 120 | 30 | 30 | 0.02 | 0 | 1 | 4.57 | 2.16 | 17.538 | 0.2 | 0.9 | 60 | 91 | 265 | 296 |
| 120 | 30 | 30 | 0.02 | 0 | 6 | 8.32 | 2.43 | 23.205 | 0.2 | 0.9 | 60 | 103 | 215 | 258 |
| 120 | 30 | 30 | 0.02 | 0 | 1 | 4.57 | 2.16 | 22.136 | 0.2 | 0.9 | 60 | 85 | 223 | 248 |
| 120 | 30 | 30 | 0.02 | 0 | 1 | 4.57 | 2.16 | 19.548 | 0.2 | 0.9 | 60 | 88 | 244 | 272 |
| 120 | 30 | 30 | 0.02 | 0 | 2 | 10.19 | 2.44 | 21.813 | 0.2 | 0.9 | 60 | 116 | 225 | 281 |
| 120 | 30 | 30 | 0.02 | 0 | 6 | 8.32 | 2.43 | 19.645 | 0.2 | 0.9 | 60 | 111 | 243 | 294 |
| 120 | 30 | 30 | 0.02 | 0 | 5 | 19.27 | 2.44 | 18.785 | 0.2 | 0.9 | 60 | 183 | 252 | 375 |
| 120 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 14.038 | 0.2 | 0.9 | 60 | 273 | 316 | 529 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 120 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 19.181 | 0.2 | 0.9 | 60 | 216 | 248 | 404 |
| 120 | 30 | 30 | 0.02 | 0 | 6 | 8.32 | 2.43 | 25.328 | 0.2 | 0.9 | 60 | 99 | 202 | 241 |
| 120 | 30 | 30 | 0.02 | 0 | 6 | 8.32 | 2.43 | 19.839 | 0.2 | 0.9 | 60 | 110 | 241 | 291 |
| 120 | 30 | 30 | 0.02 | 0 | 4 | 9.08 | 2.59 | 10.418 | 0.2 | 0.9 | 60 | 165 | 406 | 511 |
| 120 | 30 | 30 | 0.02 | 0 | 2 | 10.19 | 2.44 | 21.111 | 0.2 | 0.9 | 60 | 118 | 231 | 289 |
| 120 | 30 | 30 | 0.02 | 0 | 6 | 8.32 | 2.43 | 14.592 | 0.2 | 0.9 | 60 | 128 | 307 | 375 |
| 120 | 30 | 30 | 0.02 | 0 | 2 | 10.19 | 2.44 | 14.042 | 0.2 | 0.9 | 60 | 147 | 316 | 403 |
| 120 | 30 | 30 | 0.02 | 0 | 1 | 4.57 | 2.16 | 26.903 | 0.2 | 0.9 | 60 | 80 | 194 | 214 |
| 120 | 30 | 30 | 0.02 | 0.5 | 4 | 9.08 | 2.59 | 15.251 | 0.2 | 0.9 | 60 | 131 | 296 | 367 |
| 120 | 30 | 30 | 0.02 | 0.5 | 6 | 8.32 | 2.43 | 18.751 | 0.2 | 0.9 | 60 | 113 | 252 | 305 |
| 120 | 30 | 30 | 0.02 | 0.5 | 5 | 19.27 | 2.44 | 23.297 | 0.2 | 0.9 | 60 | 159 | 215 | 314 |
| 120 | 30 | 30 | 0.02 | 0.5 | 2 | 10.19 | 2.44 | 14.92 | 0.2 | 0.9 | 60 | 142 | 301 | 383 |
| 120 | 30 | 30 | 0.02 | 0.5 | 5 | 19.27 | 2.44 | 24.22 | 0.2 | 0.9 | 60 | 155 | 209 | 304 |
| 120 | 30 | 30 | 0.02 | 0.5 | 1 | 4.57 | 2.16 | 21.315 | 0.2 | 0.9 | 60 | 86 | 229 | 255 |
| 120 | 30 | 30 | 0.02 | 0.5 | 2 | 10.19 | 2.44 | 18.196 | 0.2 | 0.9 | 60 | 127 | 258 | 325 |
| 120 | 30 | 30 | 0.02 | 0.5 | 3 | 24.89 | 3.68 | 13.951 | 0.2 | 0.9 | 60 | 274 | 318 | 532 |
| 120 | 30 | 30 | 0.02 | 0.5 | 1 | 4.57 | 2.16 | 15.979 | 0.2 | 0.9 | 60 | 94 | 285 | 319 |
| 120 | 30 | 30 | 0.02 | 0.5 | 2 | 10.19 | 2.44 | 14.194 | 0.2 | 0.9 | 60 | 146 | 314 | 400 |
| 120 | 30 | 30 | 0.02 | 0.5 | 1 | 4.57 | 2.16 | 28.248 | 0.2 | 0.9 | 60 | 79 | 187 | 206 |
| 120 | 30 | 30 | 0.02 | 0.5 | 1 | 4.57 | 2.16 | 15.053 | 0.2 | 0.9 | 60 | 96 | 299 | 335 |
| 120 | 30 | 30 | 0.02 | 0.5 | 3 | 24.89 | 3.68 | 16.392 | 0.2 | 0.9 | 60 | 242 | 280 | 462 |
| 120 | 30 | 30 | 0.02 | 0.5 | 3 | 24.89 | 3.68 | 20.085 | 0.2 | 0.9 | 60 | 209 | 239 | 388 |
| 120 | 30 | 30 | 0.02 | 0.5 | 2 | 10.19 | 2.44 | 17.05 | 0.2 | 0.9 | 60 | 132 | 271 | 343 |
| 120 | 30 | 30 | 0.02 | 0.5 | 3 | 24.89 | 3.68 | 16.986 | 0.2 | 0.9 | 60 | 236 | 272 | 448 |
| 120 | 30 | 30 | 0.02 | 0.5 | 4 | 9.08 | 2.59 | 17.275 | 0.2 | 0.9 | 60 | 123 | 268 | 331 |
| 120 | 30 | 30 | 0.02 | 0.5 | 2 | 10.19 | 2.44 | 14.374 | 0.2 | 0.9 | 60 | 145 | 310 | 395 |
| 120 | 30 | 30 | 0.02 | 0.5 | 6 | 8.32 | 2.43 | 13.464 | 0.2 | 0.9 | 60 | 134 | 327 | 401 |
| 120 | 30 | 30 | 0.02 | 0.5 | 5 | 19.27 | 2.44 | 16.414 | 0.2 | 0.9 | 60 | 201 | 279 | 420 |
| 120 | 30 | 30 | 0.02 | 0.25 | 6 | 8.32 | 2.43 | 14.34 | 0.2 | 0.9 | 60 | 130 | 311 | 381 |
| 120 | 30 | 30 | 0.02 | 0.25 | 2 | 10.19 | 2.44 | 24.056 | 0.2 | 0.9 | 60 | 111 | 210 | 261 |
| 120 | 30 | 30 | 0.02 | 0.25 | 4 | 9.08 | 2.59 | 10.492 | 0.2 | 0.9 | 60 | 164 | 403 | 507 |
| 120 | 30 | 30 | 0.02 | 0.25 | 5 | 19.27 | 2.44 | 12.951 | 0.2 | 0.9 | 60 | 239 | 338 | 517 |
| 120 | 30 | 30 | 0.02 | 0.25 | 2 | 10.19 | 2.44 | 22.251 | 0.2 | 0.9 | 60 | 115 | 222 | 277 |
| 120 | 30 | 30 | 0.02 | 0.25 | 1 | 4.57 | 2.16 | 20.022 | 0.2 | 0.9 | 60 | 87 | 240 | 267 |
| 120 | 30 | 30 | 0.02 | 0.25 | 1 | 4.57 | 2.16 | 17.152 | 0.2 | 0.9 | 60 | 92 | 270 | 302 |
| 120 | 30 | 30 | 0.02 | 0.25 | 3 | 24.89 | 3.68 | 18.613 | 0.2 | 0.9 | 60 | 220 | 253 | 413 |
| 120 | 30 | 30 | 0.02 | 0.25 | 1 | 4.57 | 2.16 | 27.735 | 0.2 | 0.9 | 60 | 80 | 190 | 210 |
| 120 | 30 | 30 | 0.02 | 0.25 | 5 | 19.27 | 2.44 | 24.52 | 0.2 | 0.9 | 60 | 154 | 207 | 301 |
| 120 | 30 | 30 | 0.02 | 0.25 | 2 | 10.19 | 2.44 | 16.486 | 0.2 | 0.9 | 60 | 134 | 278 | 352 |
| 120 | 30 | 30 | 0.02 | 0.25 | 4 | 9.08 | 2.59 | 15.606 | 0.2 | 0.9 | 60 | 130 | 291 | 361 |
| 120 | 30 | 30 | 0.02 | 0.25 | 1 | 4.57 | 2.16 | 18.642 | 0.2 | 0.9 | 60 | 89 | 253 | 282 |
| 120 | 30 | 30 | 0.02 | 0.25 | 4 | 9.08 | 2.59 | 16.919 | 0.2 | 0.9 | 60 | 124 | 273 | 337 |
| 120 | 30 | 30 | 0.02 | 0.25 | 1 | 4.57 | 2.16 | 18.844 | 0.2 | 0.9 | 60 | 89 | 251 | 280 |
| 120 | 30 | 30 | 0.02 | 0.25 | 2 | 10.19 | 2.44 | 18.695 | 0.2 | 0.9 | 60 | 125 | 253 | 318 |
| 120 | 30 | 30 | 0.02 | 0.25 | 3 | 24.89 | 3.68 | 14.653 | 0.2 | 0.9 | 60 | 264 | 306 | 510 |
| 120 | 30 | 30 | 0.02 | 0.25 | 2 | 10.19 | 2.44 | 25.219 | 0.2 | 0.9 | 60 | 108 | 203 | 251 |
| 120 | 30 | 30 | 0.02 | 0.25 | 4 | 9.08 | 2.59 | 18.434 | 0.2 | 0.9 | 60 | 119 | 255 | 314 |
| 120 | 30 | 30 | 0.02 | 0.25 | 4 | 9.08 | 2.59 | 17.635 | 0.2 | 0.9 | 60 | 122 | 264 | 326 |
| 120 | 30 | 5 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 18.2 | 0.2 | 0.9 | 60 | 224 | 258 | 422 |
| 120 | 30 | 5 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 16.728 | 0.2 | 0.9 | 60 | 198 | 275 | 413 |
| 120 | 30 | 5 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 16.648 | 0.2 | 0.9 | 60 | 125 | 276 | 341 |
| 120 | 30 | 5 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 11.198 | 0.2 | 0.9 | 60 | 157 | 381 | 478 |
| 120 | 30 | 5 | 0.01 | 0.25 | 6 | 8.32 | 2.43 | 13.711 | 0.2 | 0.9 | 60 | 133 | 323 | 396 |
| 120 | 30 | 5 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 21.519 | 0.2 | 0.9 | 60 | 167 | 227 | 334 |
| 120 | 30 | 5 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 18.999 | 0.2 | 0.9 | 60 | 182 | 249 | 371 |
| 120 | 30 | 5 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 16.733 | 0.2 | 0.9 | 60 | 239 | 275 | 454 |
| 120 | 30 | 5 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 10.532 | 0.2 | 0.9 | 60 | 163 | 402 | 505 |
| 120 | 30 | 5 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 16.849 | 0.2 | 0.9 | 60 | 125 | 274 | 339 |
| 120 | 30 | 5 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 17.355 | 0.2 | 0.9 | 60 | 193 | 267 | 400 |
| 120 | 30 | 5 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 19.401 | 0.2 | 0.9 | 60 | 179 | 246 | 365 |
| 120 | 30 | 5 | 0.01 | 0.25 | 1 | 4.57 | 2.16 | 28.547 | 0.2 | 0.9 | 60 | 79 | 186 | 205 |
| 120 | 30 | 5 | 0.01 | 0.25 | 1 | 4.57 | 2.16 | 26.62 | 0.2 | 0.9 | 60 | 81 | 195 | 216 |
| 120 | 30 | 5 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 16.655 | 0.2 | 0.9 | 60 | 239 | 276 | 455 |
| 120 | 30 | 5 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 13.744 | 0.2 | 0.9 | 60 | 228 | 322 | 490 |
| 120 | 30 | 5 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 16.84 | 0.2 | 0.9 | 60 | 125 | 274 | 339 |
| 120 | 30 | 5 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 18.459 | 0.2 | 0.9 | 60 | 119 | 255 | 314 |
| 120 | 30 | 5 | 0.01 | 0.25 | 2 | 10.19 | 2.44 | 24.201 | 0.2 | 0.9 | 60 | 111 | 209 | 260 |
| 120 | 30 | 5 | 0.01 | 0.25 | 6 | 8.32 | 2.43 | 23.454 | 0.2 | 0.9 | 60 | 103 | 213 | 256 |
| 120 | 30 | 30 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 19.959 | 0.2 | 0.9 | 60 | 176 | 240 | 356 |
| 120 | 30 | 30 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 17.796 | 0.2 | 0.9 | 60 | 190 | 262 | 392 |
| 120 | 30 | 30 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 13.9 | 0.2 | 0.9 | 60 | 138 | 319 | 397 |
| 120 | 30 | 30 | 0.01 | 0.25 | 6 | 8.32 | 2.43 | 16.341 | 0.2 | 0.9 | 60 | 121 | 280 | 341 |
| 120 | 30 | 30 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 16.47 | 0.2 | 0.9 | 60 | 241 | 279 | 460 |
| 120 | 30 | 30 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 13.217 | 0.2 | 0.9 | 60 | 142 | 332 | 414 |
| 120 | 30 | 30 | 0.01 | 0.25 | 2 | 10.19 | 2.44 | 16.705 | 0.2 | 0.9 | 60 | 133 | 276 | 349 |
| 120 | 30 | 30 | 0.01 | 0.25 | 2 | 10.19 | 2.44 | 18.648 | 0.2 | 0.9 | 60 | 126 | 253 | 319 |
| 120 | 30 | 30 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 13.588 | 0.2 | 0.9 | 60 | 230 | 325 | 495 |
| 120 | 30 | 30 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 13.75 | 0.2 | 0.9 | 60 | 139 | 322 | 401 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 120 | 30 | 30 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 16.829 | 0.2 | 0.9 | 60 | 237 | 274 | 451 |
| 120 | 30 | 30 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 13.107 | 0.2 | 0.9 | 60 | 143 | 335 | 418 |
| 120 | 30 | 30 | 0.01 | 0.25 | 6 | 8.32 | 2.43 | 16.8 | 0.2 | 0.9 | 60 | 119 | 274 | 333 |
| 120 | 30 | 30 | 0.01 | 0.25 | 5 | 19.27 | 2.44 | 17.302 | 0.2 | 0.9 | 60 | 194 | 268 | 402 |
| 120 | 30 | 30 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 17.915 | 0.2 | 0.9 | 60 | 121 | 261 | 322 |
| 120 | 30 | 30 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 20.081 | 0.2 | 0.9 | 60 | 114 | 239 | 293 |
| 120 | 30 | 30 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 14.001 | 0.2 | 0.9 | 60 | 273 | 317 | 530 |
| 120 | 30 | 30 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 13.677 | 0.2 | 0.9 | 60 | 278 | 323 | 541 |
| 120 | 30 | 30 | 0.01 | 0.25 | 4 | 9.08 | 2.59 | 14.131 | 0.2 | 0.9 | 60 | 137 | 315 | 392 |
| 120 | 30 | 30 | 0.01 | 0.25 | 3 | 24.89 | 3.68 | 12.323 | 0.2 | 0.9 | 60 | 302 | 352 | 594 |

# Table B.2 Detection Comparison: Threshold Detection Results

| Threshold | N | γ | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 | Threshold DiffTEn1 | DiffTEx2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2 | 1 | 1 | 61 | 115 | 15 | 1 | 61 | 115 | 30 | 1 | 61 | 115 | 0.0333 | 0.0000 |
| | 2 | 1 | 1 | 61 | 153 | 15 | 1 | 61 | 153 | 30 | 1 | 61 | 153 | 0.0333 | 0.0000 |
| | 2 | 1 | 1 | 61 | 139 | 15 | 1 | 61 | 139 | 30 | 1 | 61 | 139 | 0.0333 | 0.0000 |
| | 2 | 1 | 1 | 61 | 107 | 15 | 1 | 61 | 107 | 30 | 1 | 61 | 107 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 143 | 15 | 1 | 61 | 143 | 30 | 1 | 61 | 143 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 153 | 15 | 1 | 61 | 153 | 30 | 1 | 61 | 153 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 140 | 15 | 1 | 61 | 140 | 30 | 1 | 61 | 140 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 126 | 15 | 1 | 61 | 126 | 30 | 1 | 61 | 126 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 114 | 15 | 1 | 61 | 114 | 30 | 1 | 61 | 114 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 148 | 15 | 1 | 61 | 148 | 30 | 1 | 61 | 148 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 133 | 15 | 1 | 61 | 133 | 30 | 1 | 61 | 133 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 150 | 15 | 1 | 61 | 150 | 30 | 1 | 61 | 150 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 174 | 15 | 1 | 61 | 174 | 30 | 1 | 61 | 174 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 123 | 15 | 1 | 61 | 123 | 30 | 1 | 61 | 123 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 168 | 15 | 1 | 61 | 168 | 30 | 1 | 61 | 168 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 150 | 15 | 1 | 61 | 150 | 30 | 1 | 61 | 150 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 112 | 15 | 1 | 61 | 112 | 30 | 1 | 61 | 112 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 147 | 15 | 1 | 61 | 147 | 30 | 1 | 61 | 147 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 116 | 15 | 1 | 61 | 116 | 30 | 1 | 61 | 116 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 155 | 15 | 1 | 61 | 155 | 30 | 1 | 61 | 155 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 254 | 15 | 1 | 61 | 254 | 30 | 1 | 61 | 254 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 141 | 15 | 1 | 61 | 141 | 30 | 1 | 61 | 141 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 148 | 15 | 1 | 61 | 148 | 30 | 1 | 61 | 148 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 153 | 15 | 1 | 61 | 153 | 30 | 1 | 61 | 153 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 188 | 15 | 1 | 61 | 188 | 30 | 1 | 61 | 188 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 246 | 15 | 1 | 61 | 246 | 30 | 1 | 61 | 246 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 167 | 15 | 1 | 61 | 167 | 30 | 1 | 61 | 167 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 176 | 15 | 1 | 61 | 176 | 30 | 1 | 61 | 176 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 204 | 15 | 1 | 61 | 204 | 30 | 1 | 61 | 204 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 191 | 15 | 1 | 61 | 191 | 30 | 1 | 61 | 191 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 191 | 15 | 1 | 61 | 191 | 30 | 1 | 61 | 191 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 171 | 15 | 1 | 61 | 171 | 30 | 1 | 61 | 171 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 224 | 15 | 1 | 61 | 224 | 30 | 1 | 61 | 224 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 175 | 15 | 1 | 61 | 175 | 30 | 1 | 61 | 175 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 163 | 15 | 1 | 61 | 163 | 30 | 1 | 61 | 163 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 195 | 15 | 1 | 61 | 195 | 30 | 1 | 61 | 195 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 185 | 15 | 1 | 61 | 185 | 30 | 1 | 61 | 185 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 189 | 15 | 1 | 61 | 189 | 30 | 1 | 61 | 189 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 181 | 15 | 1 | 61 | 181 | 30 | 1 | 61 | 181 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 199 | 15 | 1 | 61 | 199 | 30 | 1 | 61 | 199 | 0.0333 | 0.0000 |
| | 1 | 1 | 1 | 61 | 98 | 15 | 1 | 61 | 98 | 30 | 1 | 61 | 98 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 80 | 15 | 1 | 61 | 80 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 97 | 15 | 1 | 61 | 97 | 30 | 1 | 61 | 97 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 76 | 15 | 1 | 61 | 76 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 95 | 15 | 1 | 61 | 95 | 30 | 1 | 61 | 95 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 75 | 15 | 1 | 61 | 75 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 87 | 15 | 1 | 61 | 87 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 93 | 15 | 1 | 61 | 93 | 30 | 1 | 61 | 93 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 88 | 15 | 1 | 61 | 88 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 75 | 15 | 1 | 61 | 75 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 77 | 15 | 1 | 61 | 77 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 89 | 15 | 1 | 61 | 89 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 83 | 15 | 1 | 61 | 83 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 73 | 15 | 1 | 61 | 73 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 1 | 61 | 77 | 15 | 1 | 61 | 77 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 82 | 8 | 10 | 15 | 7 | 41 | 259 | 30 | 3 | 41 | 259 | 1.7333 | 3.5667 |
| | 1 | 1 | 123 | 98 | 117 | 15 | 2 | 2 | 515 | 30 | 2 | 2 | 515 | 1.2667 | 1.4000 |
| | 1 | 1 | 32 | 114 | 121 | 15 | 12 | 78 | 165 | 30 | 2 | 78 | 165 | 1.8000 | 1.7667 |
| | 1 | 1 | 77 | ## | 394 | 15 | 10 | 461 | 571 | 30 | 4 | 461 | 571 | 11.0000 | 8.0000 |
| | 2 | 1 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 2.0000 | 3.8333 |
| | 2 | 1 | 8 | 85 | 88 | 15 | 1 | 75 | 116 | 30 | 1 | 75 | 116 | 0.8333 | 2.0667 |
| | 2 | 1 | 1 | 75 | 76 | 15 | 1 | 75 | 76 | 30 | 0 | 0 | 0 | 0.5000 | 1.7667 |
| | 2 | 1 | 2 | 80 | 81 | 15 | 2 | 80 | 81 | 30 | 0 | 0 | 0 | 0.6667 | 1.6333 |
| | 2 | 1 | 31 | 84 | 87 | 50 | 3 | 291 | 467 | 30 | 2 | 291 | 467 | 0.8000 | 1.9333 |
| | 2 | 1 | 36 | 88 | 90 | 50 | 3 | 180 | 461 | 30 | 2 | 180 | 461 | 0.9333 | 0.8333 |
| | 2 | 1 | 16 | 127 | 131 | 50 | 1 | 81 | 164 | 30 | 1 | 81 | 164 | 2.2333 | 1.9000 |
| | 2 | 1 | 2 | 75 | 76 | 50 | 1 | 75 | 94 | 30 | 0 | 0 | 0 | 0.5000 | 1.5000 |
| | 2 | 1 | 10 | 89 | 92 | 50 | 1 | 68 | 145 | 30 | 1 | 68 | 145 | 0.9667 | 2.1000 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 15 | 138 | 141 | 25 | 1 | 91 | 172 | 30 | 1 | 91 | 172 | 2.6000 | 2.0000 |
| 2 | 1 | 15 | 102 | 105 | 25 | 4 | 76 | 114 | 30 | 1 | 76 | 114 | 1.4000 | 1.0333 |
| 2 | 1 | 20 | 140 | 144 | 25 | 1 | 89 | 161 | 30 | 1 | 89 | 161 | 2.6667 | 1.6000 |
| 2 | 1 | 12 | 91 | 94 | 25 | 1 | 79 | 138 | 30 | 1 | 79 | 138 | 1.0333 | 1.7667 |
| 2 | 1 | 9 | 117 | 119 | 25 | 1 | 80 | 126 | 30 | 1 | 80 | 126 | 1.9000 | 0.6667 |
| 2 | 1 | 7 | 107 | 109 | 25 | 1 | 92 | 123 | 30 | 1 | 92 | 123 | 1.5667 | 1.9000 |
| 2 | 1 | 1 | 78 | 79 | 25 | 1 | 78 | 79 | 30 | 0 | 0 | 0 | 0.3000 | 1.2167 |
| 2 | 1 | 19 | 163 | 167 | 25 | 1 | 113 | 240 | 30 | 1 | 113 | 240 | 1.7167 | 2.0833 |
| 2 | 1 | 50 | 2 | 4 | 25 | 7 | 136 | 353 | 30 | 4 | 136 | 353 | 0.9667 | 3.2500 |
| 2 | 1 | 87 | 142 | 165 | 25 | 3 | 3 | 505 | 30 | 1 | 3 | 505 | 1.3667 | 1.6833 |
| 2 | 1 | 32 | 182 | 188 | 25 | 1 | 115 | 271 | 30 | 1 | 115 | 271 | 2.0333 | 2.2667 |
| 2 | 1 | 24 | 107 | 109 | 25 | 1 | 70 | 179 | 30 | 1 | 70 | 179 | 0.7833 | 1.4000 |
| 2 | 1 | 109 | 129 | 133 | 25 | 1 | 4 | 566 | 30 | 1 | 4 | 566 | 1.1500 | 1.7167 |
| 2 | 1 | 0 | 0 | 0 | 25 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 1.0000 | 2.3000 |
| 2 | 1 | 86 | 15 | 18 | 25 | 2 | 237 | 580 | 30 | 2 | 237 | 580 | 0.7500 | 2.3167 |
| 2 | 1 | 1 | 140 | 141 | 25 | 1 | 140 | 141 | 30 | 0 | 0 | 0 | 1.3333 | 0.5500 |
| 2 | 1 | 63 | ## | 292 | 25 | 3 | 339 | 567 | 30 | 3 | 339 | 567 | 3.8167 | 2.5667 |
| 2 | 1 | 7 | 98 | 99 | 25 | 1 | 98 | 132 | 30 | 1 | 98 | 132 | 0.6333 | 0.8667 |
| 2 | 1 | 130 | 8 | 15 | 25 | 1 | 2 | 576 | 30 | 1 | 2 | 576 | 0.8667 | 2.3667 |
| 2 | 1 | 4 | 93 | 94 | 25 | 2 | 129 | 138 | 30 | 0 | 0 | 0 | 0.5500 | 1.1167 |
| 2 | 1 | 7 | 157 | 159 | 25 | 1 | 97 | 162 | 30 | 1 | 97 | 162 | 1.6167 | 0.7333 |
| 2 | 1 | 0 | 0 | 0 | 25 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 1.0000 | 3.4167 |
| 2 | 1 | 9 | 93 | 94 | 25 | 1 | 93 | 149 | 30 | 1 | 93 | 149 | 0.5500 | 1.2000 |
| 2 | 1 | 23 | 9 | 11 | 25 | 7 | 148 | 219 | 30 | 4 | 148 | 219 | 0.8500 | 2.6000 |
| 2 | 1 | 63 | 6 | 8 | 25 | 2 | 298 | 568 | 30 | 2 | 298 | 568 | 0.9000 | 2.3833 |
| 2 | 1 | 11 | 131 | 132 | 25 | 1 | 131 | 217 | 30 | 1 | 131 | 217 | 1.1833 | 2.5167 |
| 2 | 1 | 47 | 140 | 146 | 25 | 1 | 82 | 281 | 30 | 1 | 82 | 281 | 0.6667 | 1.3000 |
| 2 | 1 | 0 | 0 | 0 | 25 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 2.1333 |
| 2 | 1 | 21 | 132 | 134 | 25 | 3 | 275 | 367 | 30 | 2 | 275 | 367 | 0.6000 | 2.5417 |
| 2 | 1 | 9 | 146 | 149 | 25 | 4 | 180 | 220 | 30 | 1 | 180 | 220 | 0.7167 | 1.2833 |
| 2 | 1 | 15 | 147 | 149 | 25 | 2 | 147 | 295 | 30 | 1 | 147 | 295 | 0.7250 | 1.9500 |
| 2 | 1 | 3 | 126 | 127 | 25 | 3 | 126 | 127 | 30 | 0 | 0 | 0 | 0.5500 | 1.6583 |
| 2 | 1 | 4 | 183 | 184 | 25 | 2 | 183 | 206 | 30 | 0 | 0 | 0 | 1.0250 | 1.6000 |
| 2 | 1 | 18 | 108 | 109 | 25 | 2 | 174 | 297 | 30 | 2 | 174 | 297 | 0.4000 | 1.9583 |
| 2 | 1 | 85 | 140 | 144 | 25 | 1 | 6 | 581 | 30 | 1 | 6 | 581 | 0.6667 | 0.6417 |
| 2 | 1 | 51 | ## | 230 | 25 | 2 | 165 | 393 | 30 | 1 | 165 | 393 | 1.3583 | 1.8583 |
| 2 | 1 | 79 | 321 | 326 | 25 | 1 | 154 | 519 | 30 | 1 | 154 | 519 | 2.1750 | 2.7167 |
| 2 | 1 | 37 | 195 | 198 | 25 | 2 | 123 | 323 | 30 | 1 | 123 | 323 | 1.1250 | 1.3000 |
| 2 | 1 | 57 | ## | 334 | 25 | 1 | 146 | 395 | 30 | 1 | 146 | 395 | 2.2333 | 1.0833 |
| 2 | 1 | 67 | ## | 253 | 25 | 1 | 103 | 421 | 30 | 1 | 103 | 421 | 1.5667 | 1.7750 |
| 2 | 1 | 101 | ## | 259 | 25 | 3 | 82 | 500 | 30 | 3 | 82 | 500 | 1.5583 | 2.3000 |
| 2 | 1 | 111 | 221 | 247 | 25 | 1 | 2 | 580 | 30 | 1 | 2 | 580 | 1.3417 | 1.6083 |
| 2 | 1 | 11 | 100 | 102 | 25 | 3 | 156 | 213 | 30 | 1 | 156 | 213 | 0.3333 | 1.4833 |
| 2 | 1 | 16 | ## | 300 | 25 | 1 | 182 | 341 | 30 | 1 | 182 | 341 | 1.9833 | 1.3500 |
| 2 | 1 | 0 | 0 | 0 | 25 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 2.0167 |
| 2 | 1 | 43 | 135 | 139 | 25 | 3 | 63 | 369 | 30 | 1 | 63 | 369 | 0.6250 | 1.4917 |
| 2 | 1 | 86 | 211 | 218 | 25 | 4 | 6 | 453 | 30 | 1 | 6 | 453 | 1.2583 | 1.5250 |
| 2 | 1 | 0 | 0 | 0 | 25 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 1.9083 |
| 2 | 1 | 0 | 0 | 0 | 25 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 3.5333 |
| 2 | 1 | 8 | 96 | 97 | 25 | 3 | 142 | 183 | 30 | 1 | 142 | 183 | 0.3000 | 1.1917 |
| 2 | 1 | 0 | 0 | 0 | 25 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 2.1583 |
| 2 | 1 | 25 | 121 | 122 | 25 | 3 | 260 | 377 | 30 | 2 | 260 | 377 | 0.5083 | 2.6750 |
| 2 | 1 | 13 | ## | 207 | 25 | 2 | 196 | 286 | 30 | 2 | 196 | 286 | 1.2083 | 1.2250 |
| 2 | 1 | 2 | ## | 206 | 25 | 1 | 205 | 217 | 30 | 0 | 0 | 0 | 1.2083 | 1.0583 |
| 2 | 1 | 38 | 128 | 130 | 25 | 2 | 192 | 325 | 30 | 2 | 192 | 325 | 0.5667 | 2.0167 |
| 2 | 1 | 94 | ## | 312 | 25 | 3 | 9 | 508 | 30 | 1 | 9 | 508 | 1.6583 | 1.5583 |
| 2 | 1 | 86 | ## | 352 | 25 | 2 | 9 | 460 | 30 | 2 | 9 | 460 | 1.4667 | 1.4500 |
| 2 | 1 | 80 | ## | 291 | 25 | 1 | 107 | 459 | 30 | 1 | 107 | 459 | 1.8250 | 1.7167 |
| 2 | 1 | 107 | 111 | 117 | 25 | 1 | 4 | 561 | 30 | 1 | 4 | 561 | 0.4250 | 3.3583 |
| 2 | 1 | 29 | 115 | 118 | 25 | 1 | 84 | 256 | 30 | 1 | 84 | 256 | 0.4583 | 1.3333 |
| 2 | 1 | 1 | 104 | 105 | 25 | 1 | 104 | 105 | 30 | 0 | 0 | 0 | 0.3667 | 0.8833 |
| 2 | 1 | 52 | ## | 292 | 25 | 2 | 190 | 426 | 30 | 1 | 190 | 426 | 1.8333 | 1.8417 |
| 2 | 1 | 77 | ## | 346 | 25 | 3 | 108 | 577 | 30 | 1 | 108 | 577 | 1.8667 | 2.0667 |
| 2 | 1 | 39 | ## | 226 | 25 | 1 | 91 | 296 | 30 | 1 | 91 | 296 | 1.3500 | 0.7500 |
| 2 | 1 | 72 | 91 | 95 | 25 | 3 | 72 | 430 | 30 | 1 | 72 | 430 | 0.2583 | 2.6167 |
| 2 | 1 | 110 | 155 | 159 | 25 | 1 | 9 | 581 | 30 | 1 | 9 | 581 | 0.7917 | 1.9667 |
| 2 | 1 | 49 | 217 | 430 | 25 | 1 | 104 | 553 | 30 | 1 | 104 | 553 | 1.3083 | 1.3083 |
| 2 | 1 | 74 | 154 | 168 | 25 | 4 | 71 | 309 | 30 | 4 | 71 | 309 | 0.7833 | 1.1250 |
| 2 | 1 | 48 | 180 | 443 | 25 | 1 | 76 | 537 | 30 | 1 | 76 | 537 | 1.0000 | 1.0500 |
| 2 | 1 | 36 | 178 | 181 | 25 | 1 | 77 | 272 | 30 | 1 | 77 | 272 | 0.9833 | 0.9917 |
| 2 | 1 | 19 | 103 | 105 | 25 | 2 | 149 | 266 | 30 | 1 | 149 | 266 | 0.3583 | 1.5917 |
| 2 | 1 | 26 | 189 | 195 | 25 | 1 | 91 | 234 | 30 | 1 | 91 | 234 | 1.0750 | 0.5250 |
| 2 | 1 | 10 | 96 | 97 | 25 | 3 | 96 | 144 | 30 | 1 | 96 | 144 | 0.3000 | 1.2583 |
| 2 | 1 | 51 | 95 | 99 | 25 | 4 | 63 | 209 | 30 | 3 | 63 | 209 | 0.2917 | 1.4417 |
| 2 | 1 | 45 | 161 | 166 | 25 | 1 | 67 | 263 | 30 | 1 | 67 | 263 | 0.8417 | 0.9583 |
| 2 | 1 | 29 | 117 | 119 | 25 | 2 | 93 | 210 | 30 | 1 | 93 | 210 | 0.4750 | 1.4583 |
| 2 | 1 | 42 | ## | 225 | 25 | 1 | 75 | 354 | 30 | 1 | 75 | 354 | 1.2167 | 1.2500 |
| 2 | 1 | 44 | ## | 390 | 25 | 1 | 102 | 492 | 30 | 1 | 102 | 492 | 1.1917 | 1.1583 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 31 | 156 | 310 | 25 | 1 | 84 | 373 | 30 | 1 | 84 | 373 | 0.8000 | 0.7833 |
| 2 | 1 | 32 | 115 | 118 | 25 | 1 | 84 | 220 | 30 | 1 | 84 | 220 | 0.4583 | 1.0250 |
| 2 | 1 | 93 | 187 | 192 | 25 | 1 | 8 | 581 | 30 | 1 | 8 | 581 | 1.0583 | 0.8250 |
| 2 | 1 | 89 | 312 | 317 | 25 | 1 | 76 | 468 | 30 | 1 | 76 | 468 | 2.1000 | 1.6167 |
| 2 | 1 | 34 | 106 | 108 | 25 | 1 | 98 | 259 | 30 | 1 | 98 | 259 | 0.3833 | 1.5083 |
| 2 | 1 | 6 | 185 | 186 | 25 | 2 | 185 | 227 | 30 | 1 | 185 | 227 | 1.0417 | 1.5750 |
| 2 | 1 | 63 | ## | 310 | 25 | 1 | 94 | 387 | 30 | 1 | 94 | 387 | 2.0000 | 0.7750 |
| 2 | 1 | 19 | 138 | 140 | 25 | 1 | 76 | 200 | 30 | 1 | 76 | 200 | 0.6500 | 0.6167 |
| 2 | 1 | 1 | 570 | 571 | 25 | 1 | 570 | 571 | 30 | 0 | 0 | 0 | 4.2500 | 1.7000 |
| 2 | 1 | 55 | 112 | 114 | 25 | 5 | 91 | 309 | 30 | 3 | 91 | 309 | 0.4333 | 1.5917 |
| 2 | 1 | 50 | ## | 248 | 25 | 2 | 29 | 423 | 30 | 1 | 29 | 423 | 1.5250 | 0.5500 |
| 2 | 1 | 23 | 144 | 146 | 25 | 9 | 332 | 398 | 30 | 2 | 332 | 398 | 0.7000 | 1.9750 |
| 2 | 1 | 53 | 183 | 187 | 25 | 5 | 85 | 264 | 30 | 3 | 85 | 264 | 1.0250 | 0.9750 |
| 2 | 1 | 2 | 129 | 130 | 25 | 2 | 129 | 130 | 30 | 0 | 0 | 0 | 0.5750 | 1.0417 |
| 2 | 1 | 13 | ## | 255 | 25 | 5 | 217 | 278 | 30 | 2 | 217 | 278 | 1.6083 | 0.5833 |
| 2 | 1 | 14 | ## | 272 | 25 | 3 | 244 | 348 | 30 | 1 | 244 | 348 | 1.7500 | 2.1667 |
| 2 | 1 | 57 | ## | 244 | 25 | 3 | 50 | 369 | 30 | 3 | 50 | 369 | 1.5167 | 0.6250 |
| 2 | 1 | 90 | 121 | 124 | 25 | 1 | 5 | 574 | 30 | 1 | 5 | 574 | 0.5083 | 2.3000 |
| 2 | 1 | 33 | ## | 287 | 25 | 9 | 407 | 498 | 30 | 5 | 407 | 498 | 1.8750 | 0.6750 |
| 2 | 1 | 46 | 117 | 119 | 25 | 8 | 25 | 119 | 30 | 5 | 25 | 119 | 0.4750 | 1.8000 |
| 2 | 1 | 83 | ## | 250 | 25 | 3 | 101 | 532 | 30 | 1 | 101 | 532 | 1.5000 | 1.7667 |
| 2 | 1 | 39 | ## | 232 | 25 | 4 | 183 | 328 | 30 | 1 | 183 | 328 | 1.3750 | 1.3000 |
| 2 | 1 | 2 | 291 | 292 | 25 | 2 | 291 | 292 | 30 | 0 | 0 | 0 | 1.9250 | 0.4250 |
| 2 | 1 | 82 | ## | 245 | 25 | 3 | 86 | 491 | 30 | 1 | 86 | 491 | 1.4417 | 1.6917 |
| 2 | 1 | 64 | 558 | 562 | 25 | 6 | 104 | 291 | 30 | 3 | 104 | 291 | 4.1500 | 1.9250 |
| 2 | 1 | 12 | ## | 411 | 25 | 7 | 149 | 185 | 30 | 1 | 149 | 185 | 2.9083 | 0.1333 |
| 2 | 1 | 18 | ## | 372 | 25 | 5 | 269 | 403 | 30 | 1 | 269 | 403 | 2.5750 | 0.2417 |
| 2 | 1 | 81 | ## | 256 | 25 | 4 | 84 | 446 | 30 | 2 | 84 | 446 | 1.6000 | 1.3667 |
| 2 | 1 | 69 | ## | 228 | 25 | 4 | 29 | 371 | 30 | 2 | 29 | 371 | 1.3750 | 1.2750 |
| 2 | 1 | 8 | 183 | 185 | 25 | 3 | 171 | 207 | 30 | 1 | 171 | 207 | 1.0250 | 0.6333 |
| 2 | 1 | 27 | ## | 282 | 25 | 5 | 126 | 241 | 30 | 3 | 126 | 241 | 1.8250 | 1.8750 |
| 2 | 1 | 1 | ## | 279 | 25 | 1 | 278 | 279 | 30 | 0 | 0 | 0 | 1.8167 | 1.9833 |
| 2 | 1 | 16 | 116 | 118 | 25 | 3 | 88 | 250 | 30 | 1 | 88 | 250 | 0.4667 | 1.3250 |
| 2 | 1 | 41 | 106 | 108 | 25 | 8 | 381 | 490 | 30 | 4 | 381 | 490 | 0.3833 | 1.3250 |
| 2 | 1 | 15 | 31 | 32 | 25 | 7 | 148 | 206 | 30 | 2 | 148 | 206 | 0.2417 | 2.2500 |
| 2 | 1 | 64 | 173 | 190 | 25 | 3 | 84 | 461 | 30 | 1 | 84 | 461 | 0.9417 | 1.8583 |
| 2 | 1 | 33 | 33 | 36 | 25 | 10 | 15 | 89 | 30 | 3 | 15 | 89 | 0.2250 | 1.4500 |
| 2 | 1 | 31 | 145 | 148 | 25 | 1 | 114 | 266 | 30 | 1 | 114 | 266 | 0.7083 | 1.2750 |
| 2 | 1 | 69 | 121 | 125 | 25 | 7 | 64 | 332 | 30 | 1 | 64 | 332 | 0.5083 | 1.8917 |
| 2 | 1 | 9 | ## | 222 | 25 | 6 | 287 | 304 | 30 | 0 | 0 | 0 | 1.3333 | 1.1583 |
| 2 | 1 | 129 | 164 | 169 | 25 | 1 | 2 | 581 | 30 | 1 | 2 | 581 | 0.8667 | 0.9417 |
| 2 | 1 | 7 | 100 | 101 | 25 | 4 | 212 | 223 | 30 | 0 | 0 | 0 | 0.3333 | 1.9667 |
| 2 | 1 | 53 | 69 | 71 | 25 | 6 | 21 | 269 | 30 | 3 | 21 | 269 | 0.0750 | 1.7417 |
| 2 | 1 | 22 | 137 | 139 | 25 | 4 | 98 | 270 | 30 | 1 | 98 | 270 | 0.6417 | 1.4917 |
| 2 | 1 | 52 | 251 | 257 | 25 | 4 | 169 | 392 | 30 | 1 | 169 | 392 | 1.5917 | 2.1083 |
| 2 | 1 | 6 | 159 | 161 | 25 | 2 | 103 | 161 | 30 | 1 | 103 | 161 | 0.8250 | 0.7500 |
| 2 | 1 | 54 | ## | 245 | 25 | 5 | 35 | 309 | 30 | 3 | 35 | 309 | 1.5000 | 0.5750 |
| 2 | 1 | 72 | 145 | 148 | 25 | 4 | 63 | 393 | 30 | 2 | 63 | 393 | 0.7083 | 1.4833 |
| 2 | 1 | 25 | 80 | 401 | 25 | 1 | 77 | 559 | 30 | 1 | 77 | 559 | 0.1667 | 0.1750 |
| 2 | 1 | 46 | 80 | 387 | 25 | 2 | 48 | 578 | 30 | 1 | 48 | 578 | 0.1667 | 0.2167 |
| 2 | 1 | 38 | 85 | 321 | 25 | 2 | 67 | 501 | 30 | 1 | 67 | 501 | 0.2083 | 0.1667 |
| 2 | 1 | 19 | 111 | 437 | 25 | 2 | 88 | 480 | 30 | 1 | 88 | 480 | 0.4250 | 0.3417 |
| 2 | 1 | 14 | 112 | 345 | 25 | 2 | 88 | 378 | 30 | 1 | 88 | 378 | 0.4333 | 0.4250 |
| 2 | 1 | 27 | 75 | 317 | 25 | 2 | 60 | 447 | 30 | 1 | 60 | 447 | 0.1250 | 0.1417 |
| 2 | 1 | 57 | 73 | 361 | 25 | 2 | 29 | 581 | 30 | 1 | 29 | 581 | 0.1083 | 0.0833 |
| 2 | 1 | 28 | 101 | 420 | 25 | 2 | 88 | 533 | 30 | 1 | 88 | 533 | 0.3417 | 0.2833 |
| 2 | 1 | 14 | 132 | 444 | 25 | 1 | 106 | 481 | 30 | 1 | 106 | 481 | 0.6000 | 0.5083 |
| 2 | 1 | 32 | 79 | 313 | 25 | 5 | 61 | 327 | 30 | 3 | 61 | 327 | 0.1583 | 0.2167 |
| 2 | 1 | 44 | 81 | 377 | 25 | 2 | 50 | 577 | 30 | 1 | 50 | 577 | 0.1750 | 0.1917 |
| 2 | 1 | 33 | 77 | 349 | 25 | 2 | 48 | 455 | 30 | 2 | 48 | 455 | 0.1417 | 0.1333 |
| 2 | 1 | 45 | 69 | 193 | 25 | 5 | 65 | 330 | 30 | 2 | 65 | 330 | 0.0750 | 0.1000 |
| 2 | 1 | 56 | 73 | 204 | 25 | 5 | 2 | 383 | 30 | 2 | 2 | 383 | 0.1083 | 0.1000 |
| 2 | 1 | 32 | 84 | 429 | 25 | 1 | 75 | 572 | 30 | 1 | 75 | 572 | 0.2000 | 0.2167 |
| 2 | 1 | 18 | 128 | 415 | 25 | 2 | 113 | 474 | 30 | 1 | 113 | 474 | 0.5667 | 0.6250 |
| 2 | 1 | 44 | 77 | 318 | 25 | 2 | 67 | 540 | 30 | 1 | 67 | 540 | 0.1417 | 0.1750 |
| 2 | 1 | 27 | 78 | 299 | 25 | 2 | 71 | 487 | 30 | 1 | 71 | 487 | 0.1500 | 0.1250 |
| 2 | 1 | 22 | 78 | 245 | 25 | 1 | 72 | 394 | 30 | 1 | 72 | 394 | 0.1500 | 0.1250 |
| 2 | 1 | 38 | 76 | 250 | 25 | 4 | 60 | 404 | 30 | 2 | 60 | 404 | 0.1333 | 0.0500 |
| 2 | 1 | 67 | ## | 227 | 25 | 5 | 77 | 344 | 30 | 3 | 77 | 344 | 1.2333 | 1.0750 |
| 2 | 1 | 73 | 151 | 318 | 25 | 1 | 2 | 578 | 30 | 1 | 2 | 578 | 0.7583 | 0.6167 |
| 2 | 1 | 19 | ## | 242 | 25 | 3 | 119 | 300 | 30 | 1 | 119 | 300 | 1.4917 | 1.2917 |
| 2 | 1 | 44 | 192 | 195 | 25 | 6 | 108 | 311 | 30 | 1 | 108 | 311 | 1.1000 | 1.2167 |
| 2 | 1 | 55 | 155 | 360 | 25 | 2 | 52 | 547 | 30 | 1 | 52 | 547 | 0.7917 | 0.8333 |
| 2 | 1 | 26 | 157 | 160 | 25 | 5 | 191 | 260 | 30 | 2 | 191 | 260 | 0.8083 | 2.1167 |
| 2 | 1 | 90 | 146 | 154 | 25 | 4 | 76 | 341 | 30 | 4 | 76 | 341 | 0.7167 | 1.6250 |
| 2 | 1 | 26 | 145 | 147 | 25 | 2 | 99 | 292 | 30 | 2 | 99 | 292 | 0.7083 | 1.4333 |
| 2 | 1 | 69 | ## | 250 | 25 | 5 | 92 | 449 | 30 | 2 | 92 | 449 | 1.4417 | 2.0417 |
| 2 | 1 | 13 | 126 | 127 | 25 | 6 | 126 | 176 | 30 | 1 | 126 | 176 | 0.5500 | 2.2833 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2 | 1 | 54 | 135 | 373 | 25 | 1 | 24 | 575 | 30 | 1 | 24 | 575 | 0.6250 | 0.6500 |
| | 2 | 1 | 76 | ## | 238 | 25 | 4 | 66 | 393 | 30 | 2 | 66 | 393 | 1.4417 | 1.5000 |
| | 2 | 1 | 103 | 123 | 131 | 25 | 1 | 9 | 581 | 30 | 1 | 9 | 581 | 0.5250 | 1.6833 |
| | 2 | 1 | 65 | ## | 275 | 25 | 3 | 41 | 420 | 30 | 2 | 41 | 420 | 1.4667 | 1.0583 |
| | 2 | 1 | 6 | 112 | 113 | 25 | 3 | 206 | 228 | 30 | 0 | 0 | 0 | 0.4333 | 1.7417 |
| | 2 | 1 | 77 | 196 | 202 | 25 | 5 | 80 | 331 | 30 | 4 | 80 | 331 | 1.1333 | 0.7583 |
| | 2 | 1 | 45 | 180 | 391 | 25 | 2 | 80 | 503 | 30 | 1 | 80 | 503 | 1.0000 | 1.1583 |
| | 2 | 1 | 44 | 192 | 402 | 25 | 2 | 85 | 502 | 30 | 1 | 85 | 502 | 1.1000 | 1.1583 |
| | 2 | 1 | 3 | 152 | 153 | 25 | 3 | 152 | 153 | 30 | 0 | 0 | 0 | 0.7667 | 1.9917 |
| | 2 | 1 | 48 | ## | 436 | 25 | 2 | 119 | 580 | 30 | 1 | 119 | 580 | 1.4500 | 1.3167 |

# Table B.3 Detection Comparison: Duration Detection Results

| Duration | N | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 | Duration DiffDEn1 | DiffDEx2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2 | 1 | 66 | 110 | 15 | 1 | 66 | 110 | 30 | 1 | 66 | 110 | 0.2000 | 0.1667 |
| | 2 | 1 | 68 | 146 | 15 | 1 | 68 | 146 | 30 | 1 | 68 | 146 | 0.2667 | 0.2333 |
| | 2 | 1 | 72 | 128 | 15 | 1 | 72 | 128 | 30 | 1 | 72 | 128 | 0.4000 | 0.3667 |
| | 2 | 1 | 64 | 104 | 15 | 1 | 64 | 104 | 30 | 1 | 64 | 104 | 0.1333 | 0.1000 |
| | 1 | 1 | 66 | 138 | 15 | 1 | 66 | 138 | 30 | 1 | 66 | 138 | 0.2000 | 0.1667 |
| | 1 | 1 | 64 | 150 | 15 | 1 | 64 | 150 | 30 | 1 | 64 | 150 | 0.1333 | 0.1000 |
| | 1 | 1 | 63 | 138 | 15 | 1 | 63 | 138 | 30 | 1 | 63 | 138 | 0.1000 | 0.0667 |
| | 1 | 1 | 63 | 124 | 15 | 1 | 63 | 124 | 30 | 1 | 63 | 124 | 0.1000 | 0.0667 |
| | 1 | 1 | 63 | 112 | 15 | 1 | 63 | 112 | 30 | 1 | 63 | 112 | 0.1000 | 0.0667 |
| | 1 | 1 | 63 | 146 | 15 | 1 | 63 | 146 | 30 | 1 | 63 | 146 | 0.1000 | 0.0667 |
| | 1 | 1 | 63 | 131 | 15 | 1 | 63 | 131 | 30 | 1 | 63 | 131 | 0.1000 | 0.0667 |
| | 1 | 1 | 64 | 147 | 15 | 1 | 64 | 147 | 30 | 1 | 64 | 147 | 0.1333 | 0.1000 |
| | 1 | 1 | 65 | 170 | 15 | 1 | 65 | 170 | 30 | 1 | 65 | 170 | 0.1667 | 0.1333 |
| | 1 | 1 | 63 | 121 | 15 | 1 | 63 | 121 | 30 | 1 | 63 | 121 | 0.1000 | 0.0667 |
| | 1 | 1 | 64 | 165 | 15 | 1 | 64 | 165 | 30 | 1 | 64 | 165 | 0.1333 | 0.1000 |
| | 1 | 1 | 68 | 143 | 15 | 1 | 68 | 143 | 30 | 1 | 68 | 143 | 0.2667 | 0.2333 |
| | 1 | 1 | 64 | 109 | 15 | 1 | 64 | 109 | 30 | 1 | 64 | 109 | 0.1333 | 0.1000 |
| | 1 | 1 | 63 | 145 | 15 | 1 | 63 | 145 | 30 | 1 | 63 | 145 | 0.1000 | 0.0667 |
| | 1 | 1 | 63 | 114 | 15 | 1 | 63 | 114 | 30 | 1 | 63 | 114 | 0.1000 | 0.0667 |
| | 1 | 1 | 64 | 152 | 15 | 1 | 64 | 152 | 30 | 1 | 64 | 152 | 0.1333 | 0.1000 |
| | 1 | 1 | 68 | 247 | 15 | 1 | 68 | 247 | 30 | 1 | 68 | 247 | 0.2667 | 0.2333 |
| | 1 | 1 | 64 | 138 | 15 | 1 | 64 | 138 | 30 | 1 | 64 | 138 | 0.1333 | 0.1000 |
| | 1 | 1 | 64 | 145 | 15 | 1 | 64 | 145 | 30 | 1 | 64 | 145 | 0.1333 | 0.1000 |
| | 1 | 1 | 67 | 147 | 15 | 1 | 67 | 147 | 30 | 1 | 67 | 147 | 0.2333 | 0.2000 |
| | 1 | 1 | 71 | 178 | 15 | 1 | 71 | 178 | 30 | 1 | 71 | 178 | 0.3667 | 0.3333 |
| | 1 | 1 | 71 | 236 | 15 | 1 | 71 | 236 | 30 | 1 | 71 | 236 | 0.3667 | 0.3333 |
| | 1 | 1 | 69 | 159 | 15 | 1 | 69 | 159 | 30 | 1 | 69 | 159 | 0.3000 | 0.2667 |
| | 1 | 1 | 67 | 170 | 15 | 1 | 67 | 170 | 30 | 1 | 67 | 170 | 0.2333 | 0.2000 |
| | 1 | 1 | 69 | 196 | 15 | 1 | 69 | 196 | 30 | 1 | 69 | 196 | 0.3000 | 0.2667 |
| | 1 | 1 | 66 | 186 | 15 | 1 | 66 | 186 | 30 | 1 | 66 | 186 | 0.2000 | 0.1667 |
| | 1 | 1 | 65 | 187 | 15 | 1 | 65 | 187 | 30 | 1 | 65 | 187 | 0.1667 | 0.1333 |
| | 1 | 1 | 68 | 164 | 15 | 1 | 68 | 164 | 30 | 1 | 68 | 164 | 0.2667 | 0.2333 |
| | 1 | 1 | 72 | 213 | 15 | 1 | 72 | 213 | 30 | 1 | 72 | 213 | 0.4000 | 0.3667 |
| | 1 | 1 | 67 | 169 | 15 | 1 | 67 | 169 | 30 | 1 | 67 | 169 | 0.2333 | 0.2000 |
| | 1 | 1 | 69 | 155 | 15 | 1 | 69 | 155 | 30 | 1 | 69 | 155 | 0.3000 | 0.2667 |
| | 1 | 1 | 68 | 188 | 15 | 1 | 68 | 188 | 30 | 1 | 68 | 188 | 0.2667 | 0.2333 |
| | 1 | 1 | 66 | 180 | 15 | 1 | 66 | 180 | 30 | 1 | 66 | 180 | 0.2000 | 0.1667 |
| | 1 | 1 | 65 | 185 | 15 | 1 | 65 | 185 | 30 | 1 | 65 | 185 | 0.1667 | 0.1333 |
| | 1 | 1 | 68 | 174 | 15 | 1 | 68 | 174 | 30 | 1 | 68 | 174 | 0.2667 | 0.2333 |
| | 1 | 1 | 68 | 192 | 15 | 1 | 68 | 192 | 30 | 1 | 68 | 192 | 0.2667 | 0.2333 |
| | 1 | 1 | 62 | 97 | 15 | 1 | 62 | 97 | 30 | 1 | 62 | 97 | 0.2000 | 0.1000 |
| | 1 | 1 | 62 | 79 | 15 | 1 | 62 | 79 | 30 | 0 | 0 | 0 | 0.2000 | 0.1000 |
| | 1 | 1 | 62 | 96 | 15 | 1 | 62 | 96 | 30 | 1 | 62 | 96 | 0.2000 | 0.1000 |
| | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 61 | 76 | 15 | 1 | 61 | 76 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 63 | 93 | 15 | 1 | 63 | 93 | 30 | 1 | 63 | 93 | 0.3000 | 0.2000 |
| | 1 | 1 | 61 | 75 | 15 | 1 | 61 | 75 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 61 | 87 | 15 | 1 | 61 | 87 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 62 | 92 | 15 | 1 | 62 | 92 | 30 | 1 | 62 | 92 | 0.2000 | 0.1000 |
| | 1 | 1 | 62 | 83 | 15 | 1 | 62 | 83 | 30 | 0 | 0 | 0 | 0.2000 | 0.1000 |
| | 1 | 1 | 62 | 87 | 15 | 1 | 62 | 87 | 30 | 0 | 0 | 0 | 0.2000 | 0.1000 |
| | 1 | 1 | 61 | 84 | 15 | 1 | 61 | 84 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 61 | 75 | 15 | 1 | 61 | 75 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 61 | 77 | 15 | 1 | 61 | 77 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 62 | 88 | 15 | 1 | 62 | 88 | 30 | 0 | 0 | 0 | 0.2000 | 0.1000 |
| | 1 | 1 | 61 | 83 | 15 | 1 | 61 | 83 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 61 | 73 | 15 | 1 | 61 | 73 | 30 | 0 | 0 | 0 | 0.1000 | 0.0000 |
| | 1 | 1 | 62 | 76 | 15 | 1 | 62 | 76 | 30 | 0 | 0 | 0 | 0.2000 | 0.1000 |
| | 1 | 7 | 69 | 70 | 15 | 1 | 69 | 101 | 30 | 1 | 69 | 101 | 0.3000 | 1.5667 |
| | 1 | 121 | 57 | 146 | 15 | 1 | 2 | 596 | 30 | 1 | 2 | 596 | 0.1000 | 0.4333 |
| | 1 | 137 | 93 | 135 | 15 | 1 | 2 | 595 | 30 | 1 | 2 | 595 | 1.1000 | 1.3000 |
| | 1 | 15 | 81 | 83 | 15 | 2 | 81 | 142 | 30 | 1 | 81 | 142 | 0.7000 | 2.3667 |
| | 2 | ## | 97 | 105 | 15 | 1 | 2 | 595 | 30 | 1 | 2 | 595 | 1.2333 | 0.3333 |
| | 2 | ## | 428 | 442 | 15 | 1 | 2 | 596 | 30 | 1 | 2 | 596 | 12.2667 | 9.7333 |
| | 2 | 94 | 81 | 85 | 15 | 8 | ## | 413 | 30 | 3 | ## | 413 | 0.7000 | 1.4667 |
| | 2 | ## | 105 | 111 | 15 | 3 | 2 | 460 | 30 | 2 | 2 | 460 | 1.5000 | 0.6333 |
| | 3 | 18 | 101 | 103 | 15 | 1 | 68 | 137 | 30 | 1 | 68 | 137 | 1.3667 | 1.4000 |
| | 3 | ## | 77 | 86 | 15 | 1 | 2 | 596 | 30 | 1 | 2 | 596 | 0.5667 | 0.9667 |
| | 3 | 55 | 105 | 112 | 15 | 12 | 69 | 189 | 30 | 2 | 69 | 189 | 1.5000 | 2.5333 |
| | 3 | 112 | 77 | 82 | 15 | 1 | 2 | 595 | 30 | 1 | 2 | 595 | 0.5667 | 1.3000 |
| | 3 | 107 | 81 | 86 | 15 | 6 | 53 | 381 | 30 | 4 | 53 | 381 | 0.7000 | 2.3000 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 16 | 138 | 141 | 15 | 3 | 106 | 147 | 30 | 1 | 106 | 147 | 2.6000 | 2.0000 |
| 3 | 8 | 98 | 100 | 15 | 1 | 79 | 114 | 30 | 1 | 79 | 114 | 1.2667 | 1.2000 |
| 3 | 72 | 135 | 144 | 15 | 11 | 76 | 183 | 30 | 4 | 76 | 183 | 2.5000 | 1.6000 |
| 3 | ## | 88 | 96 | 15 | 1 | 6 | 586 | 30 | 1 | 6 | 586 | 0.9333 | 1.7000 |
| 3 | 75 | 116 | 119 | 15 | 11 | ## | 544 | 30 | 4 | ## | 544 | 1.8667 | 0.6667 |
| 3 | 12 | 117 | 120 | 15 | 1 | 92 | 137 | 30 | 1 | 92 | 137 | 1.9000 | 1.5333 |
| 3 | 4 | 78 | 79 | 15 | 2 | 125 | 140 | 30 | 0 | 0 | 0 | 0.3000 | 1.2167 |
| 3 | 117 | 157 | 167 | 15 | 2 | 2 | 541 | 30 | 2 | 2 | 541 | 1.6167 | 2.0833 |
| 3 | 152 | 136 | 146 | 15 | 1 | 2 | 596 | 30 | 1 | 2 | 596 | 1.2667 | 0.8833 |
| 3 | 65 | 144 | 165 | 15 | 11 | 57 | 249 | 30 | 1 | 57 | 249 | 1.4000 | 1.6833 |
| 3 | 38 | 182 | 188 | 15 | 1 | 115 | 279 | 30 | 1 | 115 | 279 | 2.0333 | 2.2667 |
| 3 | 92 | 107 | 112 | 15 | 9 | 8 | 229 | 30 | 4 | 8 | 229 | 0.7833 | 1.3500 |
| 3 | ## | 161 | 173 | 15 | 1 | 4 | 594 | 30 | 1 | 4 | 594 | 1.6833 | 1.0500 |
| 3 | ## | 539 | 548 | 15 | 1 | 8 | 591 | 30 | 1 | 8 | 591 | 7.9833 | 6.8333 |
| 3 | 8 | 109 | 111 | 15 | 3 | 72 | 111 | 30 | 1 | 72 | 111 | 0.8167 | 0.7667 |
| 3 | 11 | 71 | 73 | 15 | 3 | 71 | 113 | 30 | 1 | 71 | 113 | 0.1833 | 1.6833 |
| 3 | 127 | 308 | 313 | 15 | 3 | ## | 594 | 30 | 3 | ## | 594 | 4.1333 | 2.9167 |
| 3 | 17 | 81 | 83 | 15 | 9 | 65 | 83 | 30 | 0 | 0 | 0 | 0.3500 | 1.1333 |
| 3 | ## | 8 | 15 | 15 | 2 | 2 | 576 | 30 | 1 | 2 | 576 | 0.8667 | 2.3667 |
| 3 | 154 | 76 | 86 | 15 | 1 | 2 | 596 | 30 | 1 | 2 | 596 | 0.2667 | 1.2500 |
| 3 | 6 | 97 | 98 | 15 | 3 | 142 | 162 | 30 | 0 | 0 | 0 | 0.6167 | 1.7500 |
| 3 | 6 | 107 | 109 | 15 | 1 | 84 | 117 | 30 | 1 | 84 | 117 | 0.7833 | 1.6000 |
| 3 | 119 | 38 | 42 | 15 | 3 | 2 | 363 | 30 | 3 | 2 | 363 | 0.3667 | 2.0667 |
| 3 | 99 | 94 | 99 | 15 | 5 | 195 | 432 | 30 | 5 | 195 | 432 | 0.5667 | 1.1333 |
| 3 | ## | 27 | 34 | 15 | 2 | 2 | 532 | 30 | 2 | 2 | 532 | 0.5500 | 1.9500 |
| 3 | 5 | 145 | 146 | 15 | 3 | 145 | 166 | 30 | 0 | 0 | 0 | 1.4167 | 2.2833 |
| 3 | 79 | 140 | 146 | 15 | 12 | 81 | 289 | 30 | 1 | 81 | 289 | 0.6667 | 1.3000 |
| 3 | ## | 105 | 112 | 15 | 1 | 2 | 596 | 30 | 1 | 2 | 596 | 0.3750 | 1.2000 |
| 3 | ## | 327 | 331 | 15 | 8 | 56 | 308 | 30 | 5 | 56 | 308 | 2.2250 | 0.9000 |
| 3 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 2.5250 |
| 3 | 77 | 171 | 176 | 15 | 8 | 24 | 370 | 30 | 2 | 24 | 370 | 0.9250 | 1.7250 |
| 3 | 91 | 107 | 110 | 15 | 7 | 65 | 343 | 30 | 3 | 65 | 343 | 0.3917 | 1.8000 |
| 3 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 3.1333 |
| 3 | 125 | 191 | 199 | 15 | 1 | 2 | 592 | 30 | 1 | 2 | 592 | 1.0917 | 1.2083 |
| 3 | ## | 104 | 109 | 15 | 1 | 4 | 594 | 30 | 1 | 4 | 594 | 0.3667 | 0.9333 |
| 3 | 40 | 238 | 244 | 15 | 1 | 165 | 355 | 30 | 1 | 165 | 355 | 1.4833 | 1.7417 |
| 3 | 66 | 321 | 326 | 15 | 3 | ## | 507 | 30 | 1 | ## | 507 | 2.1750 | 2.7167 |
| 3 | 50 | 123 | 126 | 15 | 6 | 123 | 323 | 30 | 1 | 123 | 323 | 0.5250 | 1.9000 |
| 3 | 44 | 290 | 295 | 15 | 1 | 146 | 373 | 30 | 1 | 146 | 373 | 1.9167 | 1.4083 |
| 3 | 45 | 261 | 264 | 15 | 4 | 174 | 333 | 30 | 1 | 174 | 333 | 1.6750 | 1.6833 |
| 3 | 56 | 358 | 363 | 15 | 2 | ## | 434 | 30 | 2 | ## | 434 | 2.4833 | 1.4333 |
| 3 | 45 | 263 | 268 | 15 | 1 | 140 | 354 | 30 | 1 | 140 | 354 | 1.6917 | 1.4333 |
| 3 | ## | 137 | 143 | 15 | 1 | 2 | 592 | 30 | 1 | 2 | 592 | 0.6417 | 1.1417 |
| 3 | ## | 325 | 332 | 15 | 6 | 67 | 466 | 30 | 2 | 67 | 466 | 2.2083 | 1.0833 |
| 3 | 110 | 114 | 117 | 15 | 4 | 5 | 309 | 30 | 3 | 5 | 309 | 0.4500 | 1.0417 |
| 3 | ## | 142 | 148 | 15 | 1 | 8 | 596 | 30 | 1 | 8 | 596 | 0.6833 | 1.4167 |
| 3 | 45 | 138 | 142 | 15 | 2 | 138 | 344 | 30 | 1 | 138 | 344 | 0.6500 | 2.1583 |
| 3 | 54 | 35 | 37 | 15 | 17 | ## | 411 | 30 | 3 | ## | 411 | 0.2083 | 1.6000 |
| 3 | 16 | 228 | 230 | 15 | 5 | ## | 284 | 30 | 1 | ## | 284 | 1.4000 | 1.6167 |
| 3 | 37 | 80 | 82 | 15 | 13 | 125 | 183 | 30 | 1 | 125 | 183 | 0.1667 | 1.3167 |
| 3 | ## | 104 | 111 | 15 | 2 | ## | 595 | 30 | 2 | ## | 595 | 0.3667 | 1.2333 |
| 3 | 113 | 372 | 377 | 15 | 2 | 3 | 445 | 30 | 2 | 3 | 445 | 2.6000 | 0.5500 |
| 3 | ## | 144 | 152 | 15 | 1 | 5 | 596 | 30 | 1 | 5 | 596 | 0.7000 | 1.6833 |
| 3 | 84 | 161 | 165 | 15 | 8 | 124 | 255 | 30 | 5 | 124 | 255 | 0.8417 | 1.4000 |
| 3 | 12 | 124 | 125 | 15 | 4 | ## | 295 | 30 | 1 | ## | 295 | 0.5333 | 2.0583 |
| 3 | 44 | 242 | 247 | 15 | 3 | 187 | 375 | 30 | 1 | 187 | 375 | 1.5167 | 2.1000 |
| 3 | 69 | 253 | 259 | 15 | 1 | 140 | 439 | 30 | 1 | 140 | 439 | 1.6083 | 2.2250 |
| 3 | 42 | 279 | 284 | 15 | 2 | 185 | 376 | 30 | 1 | 185 | 376 | 1.8250 | 1.7750 |
| 3 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 4.3333 |
| 3 | 59 | 97 | 100 | 15 | 12 | 84 | 269 | 30 | 2 | 84 | 269 | 0.3083 | 1.4833 |
| 3 | 87 | 154 | 159 | 15 | 7 | 78 | 315 | 30 | 5 | 78 | 315 | 0.7833 | 0.4333 |
| 3 | 39 | 280 | 292 | 15 | 2 | 190 | 392 | 30 | 1 | 190 | 392 | 1.8333 | 1.8417 |
| 3 | 77 | 297 | 303 | 15 | 3 | 166 | 462 | 30 | 1 | 166 | 462 | 1.9750 | 2.4250 |
| 3 | 69 | 151 | 156 | 15 | 9 | 78 | 307 | 30 | 2 | 78 | 307 | 0.7583 | 1.3333 |
| 3 | 44 | 138 | 141 | 15 | 2 | 126 | 379 | 30 | 1 | 126 | 379 | 0.6500 | 2.2333 |
| 3 | 41 | 281 | 284 | 15 | 3 | 96 | 262 | 30 | 2 | 96 | 262 | 1.8417 | 0.9250 |
| 3 | 9 | 328 | 330 | 15 | 2 | ## | 330 | 30 | 1 | ## | 330 | 2.2333 | 2.1417 |
| 3 | 43 | 157 | 165 | 15 | 2 | 94 | 269 | 30 | 1 | 94 | 269 | 0.8083 | 1.1500 |
| 3 | 41 | 259 | 359 | 15 | 1 | 176 | 464 | 30 | 1 | 176 | 464 | 1.6583 | 1.7500 |
| 3 | 39 | 137 | 141 | 15 | 1 | 77 | 272 | 30 | 1 | 77 | 272 | 0.6417 | 1.3250 |
| 3 | 10 | 241 | 243 | 15 | 4 | 189 | 204 | 30 | 0 | 0 | 0 | 1.5083 | 0.4417 |
| 3 | 20 | 189 | 195 | 15 | 1 | 98 | 229 | 30 | 1 | 98 | 229 | 1.0750 | 0.5250 |
| 3 | 84 | 139 | 144 | 15 | 9 | ## | 595 | 30 | 5 | ## | 595 | 0.6583 | 0.8667 |
| 3 | 31 | 116 | 120 | 15 | 3 | 68 | 154 | 30 | 2 | 68 | 154 | 0.4667 | 1.2667 |
| 3 | 38 | 161 | 166 | 15 | 1 | 83 | 257 | 30 | 1 | 83 | 257 | 0.8417 | 0.9583 |
| 3 | 7 | 186 | 188 | 15 | 6 | ## | 210 | 30 | 0 | 0 | 0 | 1.0500 | 0.8833 |
| 3 | 39 | 196 | 205 | 15 | 1 | 121 | 294 | 30 | 1 | 121 | 294 | 1.1333 | 1.4167 |
| 3 | 41 | 222 | 354 | 15 | 1 | 131 | 454 | 30 | 1 | 131 | 454 | 1.3500 | 1.4583 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 36 | 193 | 290 | 15 | 1 | 111 | 346 | 30 | 1 | 111 | 346 | 1.1083 | 0.9500 |
| 3 | 30 | 115 | 118 | 15 | 1 | 84 | 206 | 30 | 1 | 84 | 206 | 0.4583 | 1.0250 |
| 3 | 45 | 98 | 102 | 15 | 2 | 70 | 263 | 30 | 1 | 70 | 263 | 0.3167 | 1.5750 |
| 3 | 64 | 163 | 166 | 15 | 1 | 131 | 447 | 30 | 1 | 131 | 447 | 0.8583 | 2.8750 |
| 3 | 26 | 106 | 108 | 15 | 3 | 157 | 240 | 30 | 2 | 157 | 240 | 0.3833 | 1.5083 |
| 3 | 35 | 226 | 230 | 15 | 5 | 111 | 246 | 30 | 1 | 111 | 246 | 1.3833 | 1.2083 |
| 3 | 44 | 134 | 137 | 15 | 1 | 116 | 353 | 30 | 1 | 116 | 353 | 0.6167 | 2.2167 |
| 3 | 7 | 95 | 96 | 15 | 2 | 95 | 153 | 30 | 1 | 95 | 153 | 0.2917 | 0.9833 |
| 3 | 54 | 104 | 106 | 15 | 13 | 173 | 271 | 30 | 3 | 173 | 271 | 0.3667 | 2.1750 |
| 3 | 47 | 112 | 114 | 15 | 12 | 181 | 258 | 30 | 3 | 181 | 258 | 0.4333 | 1.5917 |
| 3 | 59 | 243 | 248 | 15 | 9 | 82 | 252 | 30 | 2 | 82 | 252 | 1.5250 | 0.5500 |
| 3 | 56 | 89 | 91 | 15 | 12 | ## | 313 | 30 | 3 | ## | 313 | 0.2417 | 2.4333 |
| 3 | 39 | 183 | 187 | 15 | 8 | 98 | 233 | 30 | 2 | 98 | 233 | 1.0250 | 0.9750 |
| 3 | 31 | 8 | 9 | 15 | 13 | 98 | 133 | 30 | 1 | 98 | 133 | 0.4333 | 2.0500 |
| 3 | 42 | 253 | 256 | 15 | 13 | ## | 278 | 30 | 2 | ## | 278 | 1.6083 | 0.5750 |
| 3 | 50 | 283 | 290 | 15 | 7 | 197 | 411 | 30 | 1 | 197 | 411 | 1.8583 | 2.0167 |
| 3 | 65 | 239 | 244 | 15 | 8 | 114 | 269 | 30 | 5 | 114 | 269 | 1.4917 | 0.6250 |
| 3 | 101 | 205 | 210 | 15 | 8 | 101 | 368 | 30 | 3 | 101 | 368 | 1.2083 | 1.5833 |
| 3 | 50 | 92 | 94 | 15 | 15 | 321 | 382 | 30 | 2 | 321 | 382 | 0.2667 | 0.9333 |
| 3 | 51 | 117 | 119 | 15 | 14 | 51 | 119 | 30 | 3 | 51 | 119 | 0.4750 | 1.8000 |
| 3 | 46 | 244 | 248 | 15 | 8 | 199 | 354 | 30 | 2 | 199 | 354 | 1.5333 | 1.7833 |
| 3 | 40 | 225 | 232 | 15 | 5 | 183 | 263 | 30 | 2 | 183 | 263 | 1.3750 | 1.3000 |
| 3 | 64 | 134 | 136 | 15 | 14 | ## | 535 | 30 | 5 | ## | 535 | 0.6167 | 1.7250 |
| 3 | 48 | 271 | 276 | 15 | 9 | 219 | 332 | 30 | 1 | 219 | 332 | 1.7583 | 1.4333 |
| 3 | 48 | 104 | 106 | 15 | 11 | ## | 475 | 30 | 3 | ## | 475 | 0.3667 | 1.8750 |
| 3 | 32 | 259 | 261 | 15 | 15 | 111 | 153 | 30 | 2 | 111 | 153 | 1.6583 | 1.1167 |
| 3 | 19 | 369 | 372 | 15 | 12 | 127 | 154 | 30 | 0 | 0 | 0 | 2.5750 | 0.2417 |
| 3 | 58 | 178 | 180 | 15 | 11 | 171 | 282 | 30 | 3 | 171 | 282 | 0.9833 | 2.0000 |
| 3 | 77 | 225 | 230 | 15 | 7 | 29 | 186 | 30 | 3 | 29 | 186 | 1.3750 | 1.2583 |
| 3 | 40 | 133 | 137 | 15 | 11 | 90 | 213 | 30 | 1 | 90 | 213 | 0.6083 | 1.0333 |
| 3 | 24 | 279 | 282 | 15 | 13 | 318 | 363 | 30 | 1 | 318 | 363 | 1.8250 | 1.8750 |
| 3 | 47 | 281 | 284 | 15 | 6 | 189 | 407 | 30 | 1 | 189 | 407 | 1.8417 | 1.9417 |
| 3 | 107 | 116 | 122 | 15 | 5 | 78 | 254 | 30 | 5 | 78 | 254 | 0.4667 | 1.2917 |
| 3 | ## | 69 | 72 | 15 | 5 | 149 | 467 | 30 | 4 | 149 | 467 | 0.0750 | 1.6250 |
| 3 | 117 | 203 | 208 | 15 | 2 | ## | 596 | 30 | 2 | ## | 596 | 1.1917 | 0.7833 |
| 3 | 51 | 228 | 235 | 15 | 3 | 122 | 334 | 30 | 1 | 122 | 334 | 1.4000 | 1.4833 |
| 3 | 36 | 33 | 36 | 15 | 14 | 54 | 95 | 30 | 1 | 54 | 95 | 0.2250 | 1.4500 |
| 3 | 33 | 145 | 148 | 15 | 3 | 114 | 233 | 30 | 1 | 114 | 233 | 0.7083 | 1.2750 |
| 3 | ## | 184 | 192 | 15 | 5 | 52 | 444 | 30 | 2 | 52 | 444 | 1.0333 | 1.3333 |
| 3 | ## | 97 | 101 | 15 | 5 | 5 | 347 | 30 | 3 | 5 | 347 | 0.3083 | 2.1667 |
| 3 | 97 | 254 | 258 | 15 | 5 | 2 | 329 | 30 | 4 | 2 | 329 | 1.6167 | 0.2000 |
| 3 | ## | 220 | 226 | 15 | 1 | 2 | 591 | 30 | 1 | 2 | 591 | 1.3333 | 0.9250 |
| 3 | 75 | 120 | 126 | 15 | 12 | 93 | 269 | 30 | 2 | 93 | 269 | 0.5000 | 1.2833 |
| 3 | 43 | 136 | 139 | 15 | 10 | 98 | 270 | 30 | 1 | 98 | 270 | 0.6333 | 1.4917 |
| 3 | 45 | 251 | 257 | 15 | 1 | 198 | 385 | 30 | 1 | 198 | 385 | 1.5917 | 2.1083 |
| 3 | 50 | 102 | 106 | 15 | 9 | 26 | 257 | 30 | 1 | 26 | 257 | 0.3500 | 1.2083 |
| 3 | 52 | 240 | 245 | 15 | 12 | 89 | 265 | 30 | 1 | 89 | 265 | 1.5000 | 0.5750 |
| 3 | 25 | 190 | 193 | 15 | 7 | 181 | 268 | 30 | 2 | 181 | 268 | 1.0833 | 1.1083 |
| 3 | 14 | 127 | 359 | 15 | 1 | 112 | 390 | 30 | 1 | 112 | 390 | 0.5583 | 0.5250 |
| 3 | 11 | 115 | 363 | 15 | 2 | 101 | 380 | 30 | 1 | 101 | 380 | 0.4583 | 0.4167 |
| 3 | 8 | 89 | 311 | 15 | 3 | 87 | 323 | 30 | 1 | 87 | 323 | 0.2417 | 0.2500 |
| 3 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 3.9833 |
| 3 | 20 | 102 | 359 | 15 | 7 | 85 | 387 | 30 | 1 | 85 | 387 | 0.3500 | 0.3083 |
| 3 | 11 | 105 | 290 | 15 | 1 | 98 | 330 | 30 | 1 | 98 | 330 | 0.3750 | 0.3667 |
| 3 | 13 | 113 | 339 | 15 | 2 | 100 | 372 | 30 | 1 | 100 | 372 | 0.4417 | 0.2667 |
| 3 | 9 | 140 | 386 | 15 | 1 | 118 | 414 | 30 | 1 | 118 | 414 | 0.6667 | 0.5667 |
| 3 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 30 | 0 | 0 | 0 | 0.5000 | 4.2083 |
| 3 | 14 | 86 | 306 | 15 | 5 | 80 | 327 | 30 | 1 | 80 | 327 | 0.2167 | 0.2750 |
| 3 | 14 | 112 | 350 | 15 | 2 | 98 | 388 | 30 | 1 | 98 | 388 | 0.4333 | 0.4167 |
| 3 | 15 | 108 | 319 | 15 | 2 | 95 | 364 | 30 | 1 | 95 | 364 | 0.4000 | 0.3833 |
| 3 | 17 | 133 | 193 | 15 | 8 | 67 | 199 | 30 | 1 | 67 | 199 | 0.6083 | 0.1000 |
| 3 | 69 | 73 | 205 | 15 | 7 | 34 | 383 | 30 | 2 | 34 | 383 | 0.1083 | 0.0917 |
| 3 | 9 | 137 | 381 | 15 | 1 | 124 | 396 | 30 | 1 | 124 | 396 | 0.6417 | 0.6167 |
| 3 | 17 | 136 | 415 | 15 | 3 | 113 | 456 | 30 | 1 | 113 | 456 | 0.6333 | 0.6250 |
| 3 | 18 | 90 | 306 | 15 | 5 | 81 | 320 | 30 | 1 | 81 | 320 | 0.2500 | 0.2750 |
| 3 | 11 | 93 | 287 | 15 | 6 | 79 | 296 | 30 | 1 | 79 | 296 | 0.2750 | 0.2250 |
| 3 | 5 | 85 | 237 | 15 | 2 | 79 | 244 | 30 | 1 | 79 | 244 | 0.2083 | 0.1917 |
| 3 | 15 | 81 | 233 | 15 | 5 | 76 | 249 | 30 | 1 | 76 | 249 | 0.1750 | 0.1917 |
| 3 | 50 | 245 | 253 | 15 | 5 | 116 | 320 | 30 | 1 | 116 | 320 | 1.5417 | 0.8583 |
| 3 | 52 | 218 | 229 | 15 | 6 | 147 | 340 | 30 | 1 | 147 | 340 | 1.3167 | 1.3583 |
| 3 | 23 | 239 | 242 | 15 | 10 | ## | 276 | 30 | 2 | ## | 276 | 1.4917 | 1.2917 |
| 3 | 23 | 192 | 195 | 15 | 8 | ## | 283 | 30 | 1 | ## | 283 | 1.1000 | 1.2167 |
| 3 | 47 | 170 | 330 | 15 | 5 | 105 | 438 | 30 | 1 | 105 | 438 | 0.9167 | 1.0833 |
| 3 | 71 | 157 | 160 | 15 | 7 | 103 | 382 | 30 | 1 | 103 | 382 | 0.8083 | 2.1167 |
| 3 | 60 | 155 | 159 | 15 | 8 | 105 | 319 | 30 | 2 | 105 | 319 | 0.7917 | 1.5833 |
| 3 | 88 | 153 | 160 | 15 | 10 | 88 | 340 | 30 | 1 | 88 | 340 | 0.7750 | 1.3250 |
| 3 | 64 | 233 | 250 | 15 | 6 | 154 | 425 | 30 | 1 | 154 | 425 | 1.4417 | 2.0417 |
| 3 | 68 | 231 | 235 | 15 | 5 | 97 | 363 | 30 | 2 | 97 | 363 | 1.4250 | 1.3833 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 46 | 176 | 349 | 15 | 5 | 101 | 441 | 30 | 1 | 101 | 441 | 0.9667 | 0.8500 |
| 3 | 47 | 148 | 151 | 15 | 8 | 131 | 351 | 30 | 1 | 131 | 351 | 0.7333 | 2.2250 |
| 3 | 63 | 123 | 128 | 15 | 10 | 94 | 381 | 30 | 1 | 94 | 381 | 0.5250 | 1.7083 |
| 3 | 43 | 185 | 191 | 15 | 3 | 154 | 344 | 30 | 1 | 154 | 344 | 1.0417 | 1.7583 |
| 3 | 21 | 205 | 207 | 15 | 6 | 191 | 228 | 30 | 2 | 191 | 228 | 1.2083 | 0.9583 |
| 3 | 44 | 197 | 201 | 15 | 9 | 98 | 240 | 30 | 1 | 98 | 240 | 1.1417 | 0.7667 |
| 3 | 47 | 205 | 381 | 15 | 3 | 111 | 501 | 30 | 1 | 111 | 501 | 1.2083 | 1.2417 |
| 3 | 6 | 323 | 325 | 15 | 3 | ## | 325 | 30 | 0 | 0 | 0 | 2.1917 | 1.8000 |
| 3 | 19 | 136 | 138 | 15 | 7 | ## | 309 | 30 | 2 | ## | 309 | 0.6333 | 2.1167 |
| 3 | 48 | 241 | 382 | 15 | 4 | 147 | 515 | 30 | 1 | 147 | 515 | 1.5083 | 1.7667 |

# Table B.4 Detection Comparison: FAPI Detection Results

| FAPI | N | γ | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 | FAPI DiffFEx1 | DiffFEx2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 1 | 62 | 164 | 15 | 1 | 62 | 164 | 50 | 1 | 62 | 164 | 0.0667 | 1.6333 |
| | 2 | 1 | 1 | 63 | 199 | 15 | 1 | 63 | 199 | 50 | 1 | 63 | 199 | 0.1000 | 1.5333 |
| | 2 | 1 | 1 | 62 | 189 | 15 | 1 | 62 | 189 | 50 | 1 | 62 | 189 | 0.0667 | 1.6667 |
| | 2 | 1 | 1 | 62 | 149 | 15 | 1 | 62 | 149 | 50 | 1 | 62 | 149 | 0.0667 | 1.4000 |
| | 1 | 1 | 1 | 63 | 188 | 15 | 1 | 63 | 188 | 50 | 1 | 63 | 188 | 0.1000 | 1.5000 |
| | 1 | 1 | 1 | 63 | 199 | 15 | 1 | 63 | 199 | 50 | 1 | 63 | 199 | 0.1000 | 1.5333 |
| | 1 | 1 | 1 | 62 | 186 | 15 | 1 | 62 | 186 | 50 | 1 | 62 | 186 | 0.0667 | 1.5333 |
| | 1 | 1 | 1 | 62 | 174 | 15 | 1 | 62 | 174 | 50 | 1 | 62 | 174 | 0.0667 | 1.6000 |
| | 1 | 1 | 1 | 62 | 163 | 15 | 1 | 62 | 163 | 50 | 1 | 62 | 163 | 0.0667 | 1.6333 |
| | 1 | 1 | 1 | 62 | 194 | 15 | 1 | 62 | 194 | 50 | 1 | 62 | 194 | 0.0667 | 1.5333 |
| | 1 | 1 | 1 | 62 | 180 | 15 | 1 | 62 | 180 | 50 | 1 | 62 | 180 | 0.0667 | 1.5667 |
| | 1 | 1 | 1 | 62 | 206 | 15 | 1 | 62 | 206 | 50 | 1 | 62 | 206 | 0.0667 | 1.8667 |
| | 1 | 1 | 1 | 62 | 228 | 15 | 1 | 62 | 228 | 50 | 1 | 62 | 228 | 0.0667 | 1.8000 |
| | 1 | 1 | 1 | 62 | 171 | 15 | 1 | 62 | 171 | 50 | 1 | 62 | 171 | 0.0667 | 1.6000 |
| | 1 | 1 | 1 | 63 | 213 | 15 | 1 | 63 | 213 | 50 | 1 | 63 | 213 | 0.1000 | 1.5000 |
| | 1 | 1 | 1 | 62 | 199 | 15 | 1 | 62 | 199 | 50 | 1 | 62 | 199 | 0.0667 | 1.6333 |
| | 1 | 1 | 1 | 62 | 160 | 15 | 1 | 62 | 160 | 50 | 1 | 62 | 160 | 0.0667 | 1.6000 |
| | 1 | 1 | 1 | 62 | 193 | 15 | 1 | 62 | 193 | 50 | 1 | 62 | 193 | 0.0667 | 1.5333 |
| | 1 | 1 | 1 | 62 | 165 | 15 | 1 | 62 | 165 | 50 | 1 | 62 | 165 | 0.0667 | 1.6333 |
| | 1 | 1 | 1 | 63 | 201 | 15 | 1 | 63 | 201 | 50 | 1 | 63 | 201 | 0.1000 | 1.5333 |
| | 1 | 1 | 1 | 63 | 291 | 15 | 1 | 63 | 291 | 50 | 1 | 63 | 291 | 0.1000 | 1.2333 |
| | 1 | 1 | 1 | 63 | 175 | 15 | 1 | 63 | 175 | 50 | 1 | 63 | 175 | 0.1000 | 1.1333 |
| | 1 | 1 | 1 | 63 | 182 | 15 | 1 | 63 | 182 | 50 | 1 | 63 | 182 | 0.1000 | 1.1333 |
| | 1 | 1 | 1 | 64 | 181 | 15 | 1 | 64 | 181 | 50 | 1 | 64 | 181 | 0.1333 | 0.9333 |
| | 1 | 1 | 1 | 64 | 223 | 15 | 1 | 64 | 223 | 50 | 1 | 64 | 223 | 0.1333 | 1.1667 |
| | 1 | 1 | 1 | 64 | 276 | 15 | 1 | 64 | 276 | 50 | 1 | 64 | 276 | 0.1333 | 1.0000 |
| | 1 | 1 | 1 | 63 | 204 | 15 | 1 | 63 | 204 | 50 | 1 | 63 | 204 | 0.1000 | 1.2333 |
| | 1 | 1 | 1 | 64 | 210 | 15 | 1 | 64 | 210 | 50 | 1 | 64 | 210 | 0.1333 | 1.1333 |
| | 1 | 1 | 1 | 64 | 236 | 15 | 1 | 64 | 236 | 50 | 1 | 64 | 236 | 0.1333 | 1.0667 |
| | 1 | 1 | 1 | 64 | 223 | 15 | 1 | 64 | 223 | 50 | 1 | 64 | 223 | 0.1333 | 1.0667 |
| | 1 | 1 | 1 | 63 | 232 | 15 | 1 | 63 | 232 | 50 | 1 | 63 | 232 | 0.1000 | 1.3667 |
| | 1 | 1 | 1 | 64 | 199 | 15 | 1 | 64 | 199 | 50 | 1 | 64 | 199 | 0.1333 | 0.9333 |
| | 1 | 1 | 1 | 64 | 255 | 15 | 1 | 64 | 255 | 50 | 1 | 64 | 255 | 0.1333 | 1.0333 |
| | 1 | 1 | 1 | 64 | 209 | 15 | 1 | 64 | 209 | 50 | 1 | 64 | 209 | 0.1333 | 1.1333 |
| | 1 | 1 | 1 | 63 | 200 | 15 | 1 | 63 | 200 | 50 | 1 | 63 | 200 | 0.1000 | 1.2333 |
| | 1 | 1 | 1 | 64 | 227 | 15 | 1 | 64 | 227 | 50 | 1 | 64 | 227 | 0.1333 | 1.0667 |
| | 1 | 1 | 1 | 64 | 217 | 15 | 1 | 64 | 217 | 50 | 1 | 64 | 217 | 0.1333 | 1.0667 |
| | 1 | 1 | 1 | 63 | 230 | 15 | 1 | 63 | 230 | 50 | 1 | 63 | 230 | 0.1000 | 1.3667 |
| | 1 | 1 | 1 | 64 | 214 | 15 | 1 | 64 | 214 | 50 | 1 | 64 | 214 | 0.1333 | 1.1000 |
| | 1 | 1 | 1 | 64 | 231 | 15 | 1 | 64 | 231 | 50 | 1 | 64 | 231 | 0.1333 | 1.0667 |
| | 1 | 1 | 1 | 62 | 159 | 15 | 1 | 62 | 159 | 50 | 1 | 62 | 159 | 0.2000 | 6.1000 |
| | 1 | 1 | 1 | 62 | 128 | 15 | 1 | 62 | 128 | 50 | 1 | 62 | 128 | 0.2000 | 4.8000 |
| | 1 | 1 | 1 | 62 | 157 | 15 | 1 | 62 | 157 | 50 | 1 | 62 | 157 | 0.2000 | 6.0000 |
| | 1 | 1 | 1 | 62 | 134 | 15 | 1 | 62 | 134 | 50 | 1 | 62 | 134 | 0.2000 | 5.0000 |
| | 1 | 1 | 1 | 62 | 134 | 15 | 1 | 62 | 134 | 50 | 1 | 62 | 134 | 0.2000 | 5.0000 |
| | 1 | 1 | 1 | 62 | 117 | 15 | 1 | 62 | 117 | 50 | 1 | 62 | 117 | 0.2000 | 4.1000 |
| | 1 | 1 | 1 | 62 | 133 | 15 | 1 | 62 | 133 | 50 | 1 | 62 | 133 | 0.2000 | 4.9000 |
| | 1 | 1 | 1 | 62 | 149 | 15 | 1 | 62 | 149 | 50 | 1 | 62 | 149 | 0.2000 | 5.4000 |
| | 1 | 1 | 1 | 62 | 122 | 15 | 1 | 62 | 122 | 50 | 1 | 62 | 122 | 0.2000 | 4.7000 |
| | 1 | 1 | 1 | 62 | 137 | 15 | 1 | 62 | 137 | 50 | 1 | 62 | 137 | 0.2000 | 5.0000 |
| | 1 | 1 | 1 | 62 | 154 | 15 | 1 | 62 | 154 | 50 | 1 | 62 | 154 | 0.2000 | 6.1000 |
| | 1 | 1 | 1 | 62 | 139 | 15 | 1 | 62 | 139 | 50 | 1 | 62 | 139 | 0.2000 | 5.5000 |
| | 1 | 1 | 1 | 62 | 138 | 15 | 1 | 62 | 138 | 50 | 1 | 62 | 138 | 0.2000 | 5.0000 |
| | 1 | 1 | 1 | 62 | 133 | 15 | 1 | 62 | 133 | 50 | 1 | 62 | 133 | 0.2000 | 4.9000 |
| | 1 | 1 | 1 | 62 | 116 | 15 | 1 | 62 | 116 | 50 | 1 | 62 | 116 | 0.2000 | 4.1000 |
| | 1 | 1 | 1 | 62 | 119 | 15 | 1 | 62 | 119 | 50 | 1 | 62 | 119 | 0.2000 | 4.2000 |
| | 1 | 1 | 1 | 62 | 150 | 15 | 1 | 62 | 150 | 50 | 1 | 62 | 150 | 0.2000 | 6.1000 |
| | 1 | 1 | 1 | 62 | 133 | 15 | 1 | 62 | 133 | 50 | 1 | 62 | 133 | 0.2000 | 5.0000 |
| | 1 | 1 | 1 | 62 | 113 | 15 | 1 | 62 | 113 | 50 | 1 | 62 | 113 | 0.2000 | 4.0000 |
| | 1 | 1 | 1 | 62 | 124 | 15 | 1 | 62 | 124 | 50 | 1 | 62 | 124 | 0.2000 | 4.7000 |
| | 1 | 1 | 38 | 109 | 118 | 15 | 8 | 64 | 134 | 50 | 2 | 64 | 134 | 1.6333 | 0.0333 |
| | 1 | 1 | 29 | 66 | 162 | 15 | 8 | 66 | 225 | 50 | 1 | 66 | 225 | 0.2000 | 0.1000 |
| | 1 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 13.5667 |
| | 1 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 14.2333 |
| | 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 15.5333 |
| | 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 14.3667 |
| | 2 | 1 | 18 | 75 | 101 | 15 | 7 | 69 | 135 | 50 | 1 | 69 | 135 | 0.5000 | 0.9333 |
| | 2 | 1 | 13 | 105 | 116 | 15 | 2 | 72 | 126 | 50 | 1 | 72 | 126 | 1.5000 | 0.4667 |
| | 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 14.5333 |
| | 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 15.5333 |
| | 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 13.1000 |
| | 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 15.3333 |
| | 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 14.2000 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 12.6667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 14.8333 |
| 2 | 1 | 30 | 90 | 192 | 15 | 9 | 81 | 206 | 50 | 1 | 81 | 206 | 1.0000 | 0.0000 |
| 2 | 1 | 30 | 80 | 158 | 15 | 9 | 69 | 208 | 50 | 1 | 69 | 208 | 0.6667 | 0.3667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 14.7333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 1.9333 | 13.8333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 7.1500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 4.8167 |
| 2 | 1 | 37 | 77 | 117 | 15 | 8 | 67 | 218 | 50 | 4 | 67 | 218 | 0.2833 | 1.3667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 5.2500 |
| 2 | 1 | 33 | 94 | 315 | 15 | 9 | 70 | 337 | 50 | 2 | 70 | 337 | 0.5667 | 0.1500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 6.4667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 5.7500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 7.3833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 7.0667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 6.7833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 7.3833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 7.1667 |
| 2 | 1 | 105 | 189 | 202 | 15 | 6 | 49 | 264 | 50 | 5 | 49 | 264 | 2.1500 | 0.7500 |
| 2 | 1 | 15 | 88 | 90 | 15 | 7 | 82 | 108 | 50 | 0 | 0 | 0 | 0.4667 | 1.1833 |
| 2 | 1 | 7 | 114 | 124 | 15 | 2 | 147 | 172 | 50 | 0 | 0 | 0 | 0.9000 | 1.3167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 6.2667 |
| 2 | 1 | 18 | 21 | 27 | 15 | 4 | 103 | 153 | 50 | 1 | 103 | 153 | 0.6500 | 2.3167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 6.9000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.9667 | 7.1667 |
| 2 | 1 | 12 | 144 | 221 | 15 | 1 | 116 | 262 | 50 | 1 | 116 | 262 | 1.4000 | 1.0333 |
| 2 | 1 | 40 | 120 | 286 | 15 | 9 | 89 | 301 | 50 | 1 | 89 | 301 | 0.5000 | 0.1333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.7083 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.1833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.3167 |
| 2 | 1 | 60 | 169 | 188 | 15 | 5 | 58 | 330 | 50 | 1 | 58 | 330 | 0.9083 | 1.6250 |
| 2 | 1 | 15 | 118 | 313 | 15 | 6 | 76 | 319 | 50 | 1 | 76 | 319 | 0.4833 | 0.1083 |
| 2 | 1 | 40 | 149 | 189 | 15 | 6 | 100 | 350 | 50 | 1 | 100 | 350 | 0.7417 | 1.5583 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.9750 |
| 2 | 1 | 32 | 142 | 157 | 15 | 6 | 109 | 255 | 50 | 1 | 109 | 255 | 0.6833 | 0.5333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.0667 |
| 2 | 1 | 16 | 205 | 535 | 15 | 3 | 178 | 580 | 50 | 1 | 178 | 580 | 1.2083 | 4.4583 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.8917 |
| 2 | 1 | 9 | 87 | 446 | 15 | 5 | 87 | 467 | 50 | 1 | 87 | 467 | 0.2250 | 0.1500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.9583 |
| 2 | 1 | 21 | 85 | 484 | 15 | 3 | 85 | 580 | 50 | 1 | 85 | 580 | 0.2083 | 0.4250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.1750 |
| 2 | 1 | 80 | 132 | 172 | 15 | 8 | 39 | 281 | 50 | 2 | 39 | 281 | 0.6000 | 0.9000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.9917 |
| 2 | 1 | 35 | 50 | 167 | 15 | 7 | 6 | 228 | 50 | 2 | 6 | 228 | 0.0833 | 0.6250 |
| 2 | 1 | 15 | 117 | 281 | 15 | 5 | 87 | 314 | 50 | 1 | 87 | 314 | 0.4750 | 0.3083 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5000 |
| 2 | 1 | 2 | 111 | 112 | 15 | 1 | 111 | 114 | 50 | 0 | 0 | 0 | 0.4250 | 0.9750 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.3083 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.8417 |
| 2 | 1 | 8 | 143 | 180 | 15 | 3 | 133 | 185 | 50 | 1 | 133 | 185 | 0.6917 | 0.6583 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.1500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.8917 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.0667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.7417 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.6833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.4583 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.7000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.5083 |
| 2 | 1 | 54 | 99 | 268 | 15 | 3 | 49 | 345 | 50 | 2 | 49 | 345 | 0.3250 | 0.0833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 3.0833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.5667 |
| 2 | 1 | 13 | 123 | 534 | 15 | 3 | 121 | 551 | 50 | 1 | 121 | 551 | 0.5250 | 0.5000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.2083 |
| 2 | 1 | 11 | 72 | 407 | 15 | 4 | 72 | 481 | 50 | 1 | 72 | 481 | 0.1000 | 0.0167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.0500 |
| 2 | 1 | 23 | 71 | 323 | 15 | 1 | 2 | 579 | 50 | 1 | 2 | 579 | 0.0917 | 0.1667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.1000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.3417 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.3750 |
| 2 | 1 | 22 | 82 | 264 | 15 | 9 | 70 | 264 | 50 | 2 | 70 | 264 | 0.1833 | 0.0500 |
| 2 | 1 | 44 | 136 | 156 | 15 | 6 | 78 | 238 | 50 | 1 | 78 | 238 | 0.6333 | 0.7667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.5750 |
| 2 | 1 | 10 | 90 | 273 | 15 | 5 | 71 | 287 | 50 | 1 | 71 | 287 | 0.2500 | 0.0667 |
| 2 | 1 | 13 | 103 | 282 | 15 | 5 | 73 | 299 | 50 | 1 | 73 | 299 | 0.3583 | 0.1000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.7167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.4333 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.4750 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.8333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.4167 |
| 2 | 1 | 16 | 86 | 479 | 15 | 6 | 82 | 497 | 50 | 1 | 82 | 497 | 0.2167 | 0.2667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.4333 |
| 2 | 1 | 17 | 72 | 370 | 15 | 4 | 72 | 389 | 50 | 2 | 72 | 389 | 0.1000 | 0.0417 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.4833 |
| 2 | 1 | 32 | 67 | 241 | 15 | 5 | 30 | 260 | 50 | 3 | 30 | 260 | 0.0583 | 0.2250 |
| 2 | 1 | 11 | 117 | 144 | 15 | 6 | 111 | 165 | 50 | 1 | 111 | 165 | 0.4750 | 1.8583 |
| 2 | 1 | 17 | 100 | 173 | 15 | 5 | 92 | 295 | 50 | 1 | 92 | 295 | 0.3333 | 1.1000 |
| 2 | 1 | 31 | 86 | 287 | 15 | 7 | 53 | 321 | 50 | 2 | 53 | 321 | 0.2167 | 0.2250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.6500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.3083 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.7167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.1333 |
| 2 | 1 | 8 | 169 | 483 | 15 | 1 | 140 | 544 | 50 | 1 | 140 | 544 | 0.9083 | 0.4083 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.1833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5083 |
| 2 | 1 | 26 | 451 | 488 | 15 | 8 | 21 | 135 | 50 | 3 | 21 | 135 | 3.2583 | 2.3500 |
| 2 | 1 | 48 | 168 | 248 | 15 | 10 | 156 | 333 | 50 | 4 | 156 | 333 | 0.9000 | 0.7250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.9917 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.6083 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.9833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.1083 |
| 2 | 1 | 36 | 95 | 315 | 15 | 1 | 5 | 579 | 50 | 1 | 5 | 579 | 0.2917 | 0.1333 |
| 2 | 1 | 13 | 304 | 358 | 15 | 3 | 260 | 358 | 50 | 2 | ## | 358 | 2.0333 | 0.3083 |
| 2 | 1 | 17 | 145 | 152 | 15 | 6 | 372 | 406 | 50 | 0 | 0 | 0 | 0.7083 | 2.0750 |
| 2 | 1 | 18 | 116 | 398 | 15 | 4 | 93 | 403 | 50 | 2 | 93 | 403 | 0.4667 | 0.1833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.6667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.6667 |
| 2 | 1 | 35 | 156 | 470 | 15 | 4 | 120 | 578 | 50 | 1 | 120 | 578 | 0.8000 | 0.3083 |
| 2 | 1 | 10 | 184 | 450 | 15 | 3 | 135 | 450 | 50 | 1 | 135 | 450 | 1.0333 | 0.5583 |
| 2 | 1 | 17 | 106 | 222 | 15 | 6 | 89 | 312 | 50 | 1 | 89 | 312 | 0.3833 | 0.4583 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.6167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.3250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.4000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 3.0917 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.3333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.9083 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.8333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.4917 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.0333 |
| 2 | 1 | 25 | 53 | 259 | 15 | 2 | 39 | 578 | 50 | 1 | 39 | 578 | 0.0583 | 0.1750 |
| 2 | 1 | 20 | 95 | 340 | 15 | 5 | 36 | 356 | 50 | 1 | 36 | 356 | 0.2917 | 0.1833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.5917 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.7500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.2250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.1250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.3250 |
| 2 | 1 | 6 | 65 | 542 | 15 | 2 | 65 | 568 | 50 | 1 | 65 | 568 | 0.0417 | 1.0750 |
| 2 | 1 | 7 | 53 | 370 | 15 | 2 | 53 | 513 | 50 | 1 | 53 | 513 | 0.0583 | 0.2417 |
| 2 | 1 | 14 | 70 | 498 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.0833 | 0.1667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5417 |
| 2 | 1 | 7 | 61 | 478 | 15 | 3 | 61 | 511 | 50 | 1 | 61 | 511 | 0.0083 | 1.2000 |
| 2 | 1 | 2 | 49 | 527 | 15 | 1 | 49 | 548 | 50 | 1 | 49 | 548 | 0.0917 | 1.3000 |
| 2 | 1 | 5 | 77 | 539 | 15 | 2 | 77 | 564 | 50 | 1 | 77 | 564 | 0.1417 | 0.7083 |
| 2 | 1 | 6 | 83 | 508 | 15 | 3 | 83 | 536 | 50 | 1 | 83 | 536 | 0.1917 | 0.0250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.0167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5083 |
| 2 | 1 | 11 | 67 | 552 | 15 | 2 | 43 | 581 | 50 | 1 | 43 | 581 | 0.0583 | 1.5583 |
| 2 | 1 | 20 | 70 | 231 | 15 | 9 | 70 | 280 | 50 | 2 | 70 | 280 | 0.0833 | 0.2167 |
| 2 | 1 | 26 | 57 | 311 | 15 | 4 | 3 | 362 | 50 | 3 | 3 | 362 | 0.0250 | 0.7917 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.0500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.7583 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.0167 |
| 2 | 1 | 10 | 64 | 358 | 15 | 3 | 64 | 578 | 50 | 1 | 64 | 578 | 0.0333 | 0.3667 |
| 2 | 1 | 27 | 67 | 391 | 15 | 4 | 7 | 427 | 50 | 2 | 7 | 427 | 0.0583 | 1.0917 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.7083 |
| 2 | 1 | 9 | 102 | 339 | 15 | 2 | 95 | 365 | 50 | 1 | 95 | 365 | 0.3500 | 0.1417 |
| 2 | 1 | 17 | 78 | 416 | 15 | 3 | 46 | 493 | 50 | 2 | 46 | 493 | 0.1500 | 0.2000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.0000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.0083 |
| 2 | 1 | 19 | 99 | 393 | 15 | 6 | 73 | 460 | 50 | 1 | 73 | 460 | 0.3250 | 0.1750 |
| 2 | 1 | 23 | 97 | 331 | 15 | 5 | 73 | 360 | 50 | 1 | 73 | 360 | 0.3083 | 0.1500 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.1833 |
| 2 | 1 | 17 | 101 | 467 | 15 | 5 | 94 | 494 | 50 | 1 | 94 | 494 | 0.3417 | 0.2333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5000 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.0833 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.3583 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.0667 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.4917 |
| 2 | 1 | 2 | 95 | 308 | 15 | 2 | 95 | 308 | 50 | 1 | 95 | 308 | 0.2917 | 0.1167 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 2.4000 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.4250 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 0.3333 |
| 2 | 1 | 1 | 2 | 581 | 15 | 1 | 2 | 581 | 50 | 1 | 2 | 581 | 0.4833 | 1.5750 |
| 2 | 1 | 8 | 132 | 540 | 15 | 2 | 123 | 568 | 50 | 1 | 123 | 568 | 0.6000 | 0.4500 |

# Table B.5 Analysis Comparison (w/Threshold Detect): Sensor and Target Information

| S Info | S Rate | Pixel X | Pixel Y | Noise | Jitter | T Info | # | Length | Width | Speed | Ib k | ITg t | 0 Noise | En 1 | En 2 | Ex 1 | Ex 2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 120 | 30 | 30 | 0.01 | 0 | | 4 | 9.08 | 2.59 | 16.357 4 | 0.2 | 0.9 | | 60 | 127 | 280 | 347 |
| | 120 | 30 | 30 | 0.01 | 0 | | 5 | 19.27 | 2.44 | 16.521 6 | 0.2 | 0.9 | | 60 | 200 | 278 | 418 |
| | 120 | 30 | 30 | 0.01 | 0 | | 5 | 19.27 | 2.44 | 20.501 9 | 0.2 | 0.9 | | 60 | 173 | 236 | 349 |
| | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 16.614 7 | 0.2 | 0.9 | | 60 | 134 | 277 | 351 |
| | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 18.783 9 | 0.2 | 0.9 | | 60 | 125 | 252 | 317 |
| | 120 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 20.132 1 | 0.2 | 0.9 | | 60 | 87 | 239 | 266 |
| | 120 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 19.082 1 | 0.2 | 0.9 | | 60 | 181 | 249 | 370 |
| | 120 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 27.225 5 | 0.2 | 0.9 | | 60 | 80 | 192 | 212 |
| | 120 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 14.254 2 | 0.2 | 0.9 | | 60 | 136 | 313 | 389 |
| | 120 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 28.584 8 | 0.2 | 0.9 | | 60 | 79 | 186 | 205 |
| | 120 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 21.713 2 | 0.2 | 0.9 | | 60 | 166 | 226 | 332 |
| | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 16.710 4 | 0.2 | 0.9 | | 60 | 133 | 275 | 348 |
| | 120 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 16.378 3 | 0.2 | 0.9 | | 60 | 201 | 280 | 421 |
| | 120 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 17.041 5 | 0.2 | 0.9 | | 60 | 92 | 271 | 303 |
| | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 22.369 | 0.2 | 0.9 | | 60 | 115 | 221 | 276 |
| | 120 | 30 | 30 | 0 | 0 | | 1 | 4.57 | 2.16 | 21.706 2 | 0.2 | 0.9 | | 60 | 85 | 226 | 251 |
| | 120 | 30 | 30 | 0 | 0 | | 5 | 19.27 | 2.44 | 15.759 6 | 0.2 | 0.9 | | 60 | 207 | 288 | 435 |
| | 120 | 30 | 30 | 0 | 0 | | 4 | 9.08 | 2.59 | 17.420 9 | 0.2 | 0.9 | | 60 | 123 | 267 | 330 |
| | 120 | 30 | 30 | 0 | 0 | | 2 | 10.19 | 2.44 | 14.234 2 | 0.2 | 0.9 | | 60 | 146 | 313 | 399 |
| | 120 | 30 | 30 | 0.04 | 0 | | 2 | 10.19 | 2.44 | 21.810 4 | 0.2 | 0.9 | | 60 | 116 | 225 | 281 |
| | 120 | 30 | 30 | 0.04 | 0 | | 6 | 8.32 | 2.43 | 20.940 3 | 0.2 | 0.9 | | 60 | 108 | 232 | 280 |
| | 120 | 30 | 30 | 0.04 | 0 | | 5 | 19.27 | 2.44 | 18.386 | 0.2 | 0.9 | | 60 | 186 | 256 | 382 |
| | 120 | 30 | 30 | 0.04 | 0 | | 4 | 9.08 | 2.59 | 15.235 8 | 0.2 | 0.9 | | 60 | 132 | 296 | 368 |
| | 120 | 30 | 30 | 0.04 | 0 | | 4 | 9.08 | 2.59 | 15.543 1 | 0.2 | 0.9 | | 60 | 130 | 292 | 362 |
| | 120 | 30 | 5 | 0.04 | 0 | | 4 | 9.08 | 2.59 | 11.749 | 0.2 | 0.9 | | 60 | 153 | 366 | 459 |
| | 120 | 30 | 5 | 0.04 | 0 | | 2 | 10.19 | 2.44 | 15.645 5 | 0.2 | 0.9 | | 60 | 138 | 290 | 368 |
| | 120 | 30 | 5 | 0.04 | 0 | | 2 | 10.19 | 2.44 | 20.122 7 | 0.2 | 0.9 | | 60 | 121 | 239 | 300 |
| | 120 | 30 | 5 | 0.04 | 0 | | 4 | 9.08 | 2.59 | 15.552 6 | 0.2 | 0.9 | | 60 | 130 | 291 | 361 |
| | 120 | 30 | 5 | 0.04 | 0 | | 6 | 8.32 | 2.43 | 23.284 1 | 0.2 | 0.9 | | 60 | 103 | 215 | 258 |
| | 120 | 30 | 5 | 0.04 | 0 | | 4 | 9.08 | 2.59 | 18.955 5 | 0.2 | 0.9 | | 60 | 117 | 250 | 307 |
| | 120 | 30 | 5 | 0.04 | 0 | | 6 | 8.32 | 2.43 | 18.052 2 | 0.2 | 0.9 | | 60 | 115 | 259 | 314 |
| | 120 | 30 | 5 | 0.04 | 0 | | 2 | 10.19 | 2.44 | 24.508 2 | 0.2 | 0.9 | | 60 | 110 | 207 | 257 |
| | 120 | 30 | 5 | 0.04 | 0 | | 5 | 19.27 | 2.44 | 15.052 1 | 0.2 | 0.9 | | 60 | 214 | 299 | 453 |
| | 120 | 30 | 5 | 0.04 | 0 | | 1 | 4.57 | 2.16 | 26.334 5 | 0.2 | 0.9 | | 60 | 81 | 197 | 218 |
| | 120 | 30 | 5 | 0.04 | 0 | | 6 | 8.32 | 2.43 | 13.837 7 | 0.2 | 0.9 | | 60 | 132 | 320 | 392 |
| | 120 | 30 | 5 | 0.04 | 0 | | 4 | 9.08 | 2.59 | 15.643 5 | 0.2 | 0.9 | | 60 | 130 | 290 | 360 |
| | 120 | 30 | 5 | 0.04 | 0 | | 3 | 24.89 | 3.68 | 15.305 6 | 0.2 | 0.9 | | 60 | 255 | 295 | 490 |
| | 120 | 30 | 5 | 0.04 | 0 | | 2 | 10.19 | 2.44 | 18.564 2 | 0.2 | 0.9 | | 60 | 126 | 254 | 320 |
| | 90 | 30 | 5 | 0.02 | 0 | | 3 | 24.89 | 3.68 | 15.262 | 0.2 | 0.9 | | 60 | 207 | 237 | 384 |
| | 90 | 30 | 5 | 0.02 | 0 | | 1 | 4.57 | 2.16 | 20.895 9 | 0.2 | 0.9 | | 60 | 80 | 189 | 209 |
| | 90 | 30 | 5 | 0.02 | 0 | | 3 | 24.89 | 3.68 | 15.983 9 | 0.2 | 0.9 | | 60 | 200 | 229 | 369 |
| | 90 | 30 | 5 | 0.02 | 0 | | 1 | 4.57 | 2.16 | 22.602 5 | 0.2 | 0.9 | | 60 | 78 | 179 | 197 |
| | 90 | 30 | 5 | 0.02 | 0 | | 1 | 4.57 | 2.16 | 20.752 3 | 0.2 | 0.9 | | 60 | 80 | 190 | 210 |
| | 90 | 30 | 5 | 0.02 | 0 | | 6 | 8.32 | 2.43 | 22.509 8 | 0.2 | 0.9 | | 60 | 93 | 180 | 213 |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 90 | 30 | 5 | 0.02 | 0 | 3 | 24.89 | 3.68 | 16.013 2 | 0.2 | 0.9 | 60 | 200 | 229 | 369 |
| 90 | 30 | 5 | 0.02 | 0 | 2 | 10.19 | 2.44 | 15.448 5 | 0.2 | 0.9 | 60 | 119 | 235 | 294 |
| 90 | 30 | 5 | 0.02 | 0 | 6 | 8.32 | 2.43 | 16.011 1 | 0.2 | 0.9 | 60 | 107 | 229 | 276 |
| 90 | 30 | 5 | 0.02 | 0 | 3 | 24.89 | 3.68 | 10.565 2 | 0.2 | 0.9 | 60 | 272 | 316 | 528 |
| 90 | 30 | 10 | 0.02 | 0 | 4 | 9.08 | 2.59 | 13.783 9 | 0.2 | 0.9 | 60 | 119 | 256 | 315 |
| 90 | 30 | 10 | 0.02 | 0 | 3 | 24.89 | 3.68 | 18.380 7 | 0.2 | 0.9 | 60 | 182 | 207 | 329 |
| 90 | 30 | 10 | 0.02 | 0 | 4 | 9.08 | 2.59 | 19.937 7 | 0.2 | 0.9 | 60 | 101 | 195 | 236 |
| 90 | 30 | 10 | 0.02 | 0 | 3 | 24.89 | 3.68 | 14.683 7 | 0.2 | 0.9 | 60 | 214 | 244 | 398 |
| 90 | 30 | 10 | 0.02 | 0 | 5 | 19.27 | 2.44 | 20.277 8 | 0.2 | 0.9 | 60 | 146 | 193 | 279 |
| 90 | 30 | 10 | 0.02 | 0 | 6 | 8.32 | 2.43 | 20.509 5 | 0.2 | 0.9 | 60 | 97 | 192 | 229 |
| 90 | 30 | 10 | 0.02 | 0 | 5 | 19.27 | 2.44 | 13.352 4 | 0.2 | 0.9 | 60 | 190 | 262 | 392 |
| 90 | 30 | 10 | 0.02 | 0 | 6 | 8.32 | 2.43 | 19.858 6 | 0.2 | 0.9 | 60 | 98 | 196 | 234 |
| 90 | 30 | 10 | 0.02 | 0 | 4 | 9.08 | 2.59 | 11.593 8 | 0.2 | 0.9 | 60 | 130 | 293 | 363 |
| 90 | 30 | 10 | 0.02 | 0 | 4 | 9.08 | 2.59 | 12.806 3 | 0.2 | 0.9 | 60 | 124 | 271 | 335 |
| 90 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 19.318 | 0.2 | 0.9 | 60 | 176 | 200 | 316 |
| 90 | 30 | 30 | 0.02 | 0 | 6 | 8.32 | 2.43 | 18.449 9 | 0.2 | 0.9 | 60 | 101 | 206 | 247 |
| 90 | 30 | 30 | 0.02 | 0 | 5 | 19.27 | 2.44 | 17.600 2 | 0.2 | 0.9 | 60 | 159 | 213 | 312 |
| 90 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 12.661 1 | 0.2 | 0.9 | 60 | 237 | 273 | 450 |
| 90 | 30 | 30 | 0.02 | 0 | 6 | 8.32 | 2.43 | 24.497 7 | 0.2 | 0.9 | 60 | 91 | 170 | 201 |
| 90 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 11.789 7 | 0.2 | 0.9 | 60 | 250 | 289 | 479 |
| 90 | 30 | 30 | 0.02 | 0 | 1 | 4.57 | 2.16 | 19.450 7 | 0.2 | 0.9 | 60 | 81 | 199 | 220 |
| 90 | 30 | 30 | 0.02 | 0 | 5 | 19.27 | 2.44 | 19.727 9 | 0.2 | 0.9 | 60 | 148 | 197 | 285 |
| 90 | 30 | 30 | 0.02 | 0 | 3 | 24.89 | 3.68 | 10.488 8 | 0.2 | 0.9 | 60 | 274 | 317 | 531 |
| 90 | 30 | 30 | 0.02 | 0 | 1 | 4.57 | 2.16 | 18.959 | 0.2 | 0.9 | 60 | 82 | 202 | 224 |

## Table B.6 Analysis Comparison (w/ Threshold Detection): PolyFit and FermiFit Results

| Threshold | N | γ | #T | FAR2 | #T | En1 | Ex2 | Poly | En1 | En2 | Ex1 | Ex2 | Fermi | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 3 | 1 | 51 | 30 | 1 | 87 | 328 | | 62.9 | 131 | 289 | 345 | | 62.9 | 138 | 270 | 345 |
| | 3 | 1 | 25 | 30 | 1 | 101 | 378 | | 58.7 | 196 | 276 | 419 | | 58.7 | 184 | 284 | 419 |
| | 3 | 1 | 24 | 30 | 1 | 73 | 336 | | 49.9 | 172 | 239 | 352 | | 49.9 | 160 | 242 | 352 |
| | 3 | 1 | 1 | 30 | 1 | 107 | 305 | | 60.7 | 136 | 275 | 350 | | 61 | 144 | 267 | 350 |
| | 3 | 1 | 1 | 30 | 1 | 101 | 277 | | 60.1 | 127 | 250 | 317 | | 60 | 136 | 242 | 317 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 60.1 | 127 | 250 | 317 | | 60 | 133 | 244 | 317 |
| | 3 | 1 | 1 | 30 | 1 | 100 | 331 | | 60.8 | 173 | 254 | 369 | | 61 | 177 | 252 | 369 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 60.8 | 173 | 254 | 369 | | 61 | 155 | 275 | 369 |
| | 3 | 1 | 1 | 30 | 1 | 111 | 339 | | 56.4 | 141 | 307 | 393 | | 56 | 157 | 292 | 393 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 56.4 | 141 | 307 | 393 | | 56 | 155 | 294 | 393 |
| | 3 | 1 | 1 | 30 | 1 | 95 | 298 | | 60.6 | 158 | 230 | 331 | | 61 | 162 | 225 | 331 |
| | 3 | 1 | 1 | 30 | 1 | 106 | 303 | | 60.7 | 135 | 273 | 347 | | 61 | 143 | 265 | 347 |
| | 3 | 1 | 1 | 30 | 1 | 107 | 375 | | 61.6 | 191 | 286 | 419 | | 62 | 196 | 284 | 419 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 61.6 | 191 | 286 | 419 | | 62 | 173 | 308 | 419 |
| | 3 | 1 | 1 | 30 | 1 | 95 | 242 | | 60.1 | 117 | 219 | 276 | | 60 | 123 | 213 | 276 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 60.1 | 117 | 219 | 276 | | 60 | 120 | 216 | 276 |
| | 3 | 1 | 1 | 30 | 1 | 109 | 387 | | 61.6 | 197 | 295 | 433 | | 62 | 201 | 292 | 433 |
| | 3 | 1 | 1 | 30 | 1 | 102 | 289 | | 57.2 | 127 | 263 | 332 | | 57 | 137 | 252 | 332 |
| | 3 | 1 | 1 | 30 | 1 | 114 | 346 | | 61.5 | 148 | 311 | 397 | | 62 | 154 | 300 | 397 |
| | 2 | 1 | 132 | 30 | 2 | 23 | 405 | | 0.76 | 157 | 325 | 341 | | 1 | 117 | 225 | 341 |
| | 2 | 1 | 140 | 30 | 0 | 0 | 0 | | 0.76 | 157 | 325 | 341 | | 1 | | 242 | 341 |
| | 2 | 1 | 125 | 30 | 2 | 20 | 505 | | 28.7 | 228 | 408 | 517 | | 29 | 206 | 340 | 517 |
| | 2 | 1 | 127 | 30 | 0 | 0 | 0 | | 28.7 | 228 | 408 | 517 | | 29 | 183 | 363 | 517 |
| | 2 | 1 | 121 | 30 | 3 | 21 | 395 | | 49.1 | 119 | 276 | 385 | | 49 | 166 | 268 | 385 |
| | 2 | 1 | 53 | 30 | 1 | 22 | 477 | | 30.4 | 211 | 292 | 493 | | 30 | 185 | 345 | 493 |
| | 2 | 1 | 71 | 30 | 2 | 20 | 381 | | 33.7 | 195 | 253 | 392 | | 34 | 164 | 262 | 392 |
| | 2 | 1 | 84 | 30 | 1 | 23 | 408 | | 31.6 | 178 | 330 | 416 | | 32 | 169 | 288 | 416 |
| | 2 | 1 | 78 | 30 | 0 | 0 | 0 | | 31.6 | 178 | 330 | 416 | | 32 | 159 | 288 | 416 |
| | 2 | 1 | 90 | 30 | 2 | 21 | 420 | | 33.3 | 132 | 278 | 425 | | 33 | 162 | 296 | 425 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 86 | 30 | 2 | 60 | 325 | 44.1 | 138 | 221 | 339 | 44 | 149 | 242 | 339 |
| 2 | 1 | 85 | 30 | 2 | 66 | 331 | 52.5 | 128 | 233 | 340 | 53 | 158 | 234 | 340 |
| 2 | 1 | 57 | 30 | 1 | 73 | 250 | 58.4 | 106 | 208 | 258 | 58 | 118 | 198 | 258 |
| 2 | 1 | 53 | 30 | 1 | 79 | 456 | 56.8 | 206 | 215 | 482 | 57 | 202 | 338 | 482 |
| 2 | 1 | 101 | 30 | 4 | 61 | 210 | 58.9 | 90.3 | 191 | 214 | 59 | 99.2 | 174 | 214 |
| 2 | 1 | 66 | 30 | 1 | 61 | 385 | 45.6 | 143 | 312 | 408 | 46 | 153 | 301 | 408 |
| 2 | 1 | 74 | 30 | 2 | 63 | 352 | 44.5 | 139 | 283 | 365 | 44 | 162 | 247 | 365 |
| 2 | 1 | 40 | 30 | 1 | 58 | 488 | 30.4 | 339 | 212 | 518 | 30 | 198 | 350 | 518 |
| 2 | 1 | 81 | 30 | 2 | 75 | 299 | 59.3 | 124 | 254 | 313 | 59 | 133 | 239 | 313 |
| 2 | 1 | 64 | 30 | 5 | 22 | 388 | 33.7 | 322 | 251 | 387 | 34 | 162 | 259 | 387 |
| 2 | 1 | 109 | 30 | 3 | 58 | 209 | 57.5 | 84.3 | 188 | 207 | 58 | 84.1 | 181 | 207 |
| 2 | 1 | 53 | 30 | 3 | 22 | 403 | 34.7 | 295 | 152 | 390 | 35 | 157 | 268 | 390 |
| 2 | 1 | 104 | 30 | 5 | 52 | 204 | 56.5 | 85.5 | 177 | 200 | 56 | 88.2 | 168 | 200 |
| 2 | 1 | 83 | 30 | 1 | 50 | 211 | 53.3 | 84.4 | 194 | 209 | 53 | 86.5 | 176 | 209 |
| 2 | 1 | 101 | 30 | 5 | 61 | 246 | 51.9 | 127 | 237 | 241 | 52 | 118 | 175 | 241 |
| 2 | 1 | 53 | 30 | 1 | 62 | 368 | 58.7 | 188 | 243 | 370 | 59 | 171 | 258 | 370 |
| 2 | 1 | 77 | 30 | 2 | 64 | 308 | 38.7 | 165 | 324 | 301 | 39 | 132 | 208 | 301 |
| 2 | 1 | 87 | 30 | 1 | 55 | 282 | 57 | 111 | 226 | 277 | 57 | 106 | 229 | 277 |
| 2 | 1 | 29 | 30 | 1 | 61 | 524 | 58.5 | 253 | 335 | 530 | 59 | 225 | 363 | 530 |
| 2 | 1 | 68 | 30 | 1 | 80 | 314 | 66.7 | 120 | 264 | 312 | 67 | 139 | 240 | 312 |
| 2 | 1 | 68 | 30 | 2 | 63 | 338 | 61.6 | 169 | 200 | 337 | 62 | 159 | 240 | 337 |
| 2 | 1 | 105 | 30 | 5 | 62 | 243 | 60.7 | 99.7 | 197 | 235 | 61 | 107 | 189 | 235 |
| 2 | 1 | 68 | 30 | 4 | 70 | 392 | 60.7 | 197 | 260 | 397 | 61 | 182 | 276 | 397 |
| 2 | 1 | 97 | 30 | 2 | 64 | 287 | 58.3 | 142 | 187 | 283 | 58 | 137 | 204 | 283 |
| 2 | 1 | 84 | 30 | 2 | 63 | 228 | 60.8 | 98.5 | 195 | 229 | 61 | 99.3 | 191 | 229 |
| 2 | 1 | 65 | 30 | 1 | 40 | 387 | 48.4 | 208 | 267 | 390 | 48 | 173 | 265 | 390 |
| 2 | 1 | 107 | 30 | 3 | 20 | 232 | 22.7 | 144 | 212 | 233 | 23 | 65.9 | 191 | 233 |
| 2 | 1 | 58 | 30 | 1 | 83 | 361 | 65.3 | 135 | 296 | 359 | 65 | 146 | 278 | 359 |
| 2 | 1 | 75 | 30 | 1 | 38 | 327 | 44.7 | 125 | 268 | 331 | 45 | 126 | 250 | 331 |
| 2 | 1 | 91 | 30 | 3 | 71 | 309 | 64 | 151 | 225 | 312 | 64 | 155 | 221 | 312 |
| 2 | 1 | 112 | 30 | 2 | 174 | 229 | 196 | 183 | 250 | 80.6 | 196 | 94.2 | 183 | 81 |
| 2 | 1 | 105 | 30 | 3 | 90 | 285 | 62.7 | 173 | 269 | 287 | 63 | 121 | 229 | 287 |
| 2 | 1 | 75 | 30 | 3 | 21 | 456 | 30.3 | 372 | 289 | 453 | 30 | 191 | 292 | 453 |
| 2 | 1 | 134 | 30 | 3 | 70 | 138 | 63.3 | 77.8 | 125 | 188 | 63 | 87.4 | 163 | 188 |
| 2 | 1 | 73 | 30 | 2 | 64 | 488 | 51.6 | 228 | 323 | 475 | 52 | 224 | 303 | 475 |
| 2 | 1 | 131 | 30 | 4 | 92 | 183 | 70.9 | 66.2 | 175 | 189 | 71 | 76.4 | 184 | 189 |
| 2 | 1 | 109 | 30 | 2 | 22 | 280 | 35.9 | 424 | 207 | 273 | 36 | 102 | 207 | 273 |
| 2 | 1 | 54 | 30 | 1 | 52 | 515 | 46.7 | 257 | 345 | 533 | 47 | 223 | 357 | 533 |
| 2 | 1 | 125 | 30 | 3 | 99 | 196 | 91.2 | 115 | 183 | 196 | 91 | 99.2 | 188 | 196 |

# Table B.7 Analysis Comparison (w/ Duration Detect): Sensor and Target Information

| S Info | S Rate | Pixel X | Pixel Y | Noise | Jitter | Targ | # | Length | Width | Speed | Ibk | ITgt | 0 Noise | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 120 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 19.2456 | 0.2 | 0.9 | | 60 | 117 | 247 | 304 |
| | 120 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 16.9973 | 0.2 | 0.9 | | 60 | 196 | 272 | 408 |
| | 120 | 30 | 30 | 0.01 | 0.05 | | 2 | 10.19 | 2.44 | 24.8741 | 0.2 | 0.9 | | 60 | 109 | 205 | 254 |
| | 120 | 30 | 30 | 0.01 | 0.05 | | 3 | 24.89 | 3.68 | 12.998 | 0.2 | 0.9 | | 60 | 290 | 337 | 567 |
| | 120 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 13.5824 | 0.2 | 0.9 | | 60 | 230 | 325 | 495 |
| | 120 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 17.0304 | 0.2 | 0.9 | | 60 | 196 | 271 | 407 |
| | 120 | 30 | 30 | 0.01 | 0.05 | | 3 | 24.89 | 3.68 | 14.4398 | 0.2 | 0.9 | | 60 | 267 | 309 | 516 |
| | 120 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 22.7559 | 0.2 | 0.9 | | 60 | 162 | 218 | 320 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 15.9028 | 0.2 | 0.9 | | 60 | 65 | 79 | 84 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 20.2498 | 0.2 | 0.9 | | 60 | 62 | 75 | 77 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 2 | 10.19 | 2.44 | 18.4967 | 0.2 | 0.9 | | 60 | 66 | 76 | 82 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 17.6384 | 0.2 | 0.9 | | 60 | 63 | 77 | 80 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 17.4494 | 0.2 | 0.9 | | 60 | 65 | 77 | 82 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 16.282 | 0.2 | 0.9 | | 60 | 66 | 78 | 84 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 14.4018 | 0.2 | 0.9 | | 60 | 73 | 81 | 94 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 14.2904 | 0.2 | 0.9 | | 60 | 66 | 81 | 87 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 18.2569 | 0.2 | 0.9 | | 60 | 65 | 76 | 81 |
| | 10 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 15.5979 | 0.2 | 0.9 | | 60 | 66 | 79 | 85 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 12.1187 | 0.2 | 0.9 | | 60 | 82 | 134 | 156 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 25.3133 | 0.2 | 0.9 | | 60 | 70 | 96 | 106 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 2 | 10.19 | 2.44 | 15.3386 | 0.2 | 0.9 | | 60 | 80 | 119 | 139 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 22.1056 | 0.2 | 0.9 | | 60 | 66 | 101 | 107 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 25.3133 | 0.2 | 0.9 | | 60 | 70 | 96 | 106 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 12.1187 | 0.2 | 0.9 | | 60 | 82 | 134 | 156 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 25.3133 | 0.2 | 0.9 | | 60 | 70 | 96 | 106 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 2 | 10.19 | 2.44 | 15.3386 | 0.2 | 0.9 | | 60 | 80 | 119 | 139 |
| | 30 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 22.1056 | 0.2 | 0.9 | | 60 | 66 | 101 | 107 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 3 | 24.89 | 3.68 | 18.8432 | 0.2 | 0.9 | | 60 | 139 | 156 | 235 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 19.5218 | 0.2 | 0.9 | | 60 | 88 | 152 | 180 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 3 | 24.89 | 3.68 | 12.8738 | 0.2 | 0.9 | | 60 | 176 | 200 | 316 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 16.5197 | 0.2 | 0.9 | | 60 | 93 | 169 | 202 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 25.1346 | 0.2 | 0.9 | | 60 | 71 | 132 | 143 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 3 | 24.89 | 3.68 | 15.0369 | 0.2 | 0.9 | | 60 | 159 | 180 | 279 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 16.4862 | 0.2 | 0.9 | | 60 | 77 | 169 | 186 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 24.562 | 0.2 | 0.9 | | 60 | 107 | 133 | 180 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 14.4115 | 0.2 | 0.9 | | 60 | 98 | 185 | 223 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 19.3816 | 0.2 | 0.9 | | 60 | 88 | 153 | 181 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 18.2653 | 0.2 | 0.9 | | 60 | 75 | 159 | 174 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 23.1095 | 0.2 | 0.9 | | 60 | 110 | 138 | 188 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 3 | 24.89 | 3.68 | 16.3089 | 0.2 | 0.9 | | 60 | 152 | 170 | 262 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 24.9397 | 0.2 | 0.9 | | 60 | 80 | 132 | 152 |
| | 60 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 21.3418 | 0.2 | 0.9 | | 60 | 83 | 144 | 167 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 2 | 10.19 | 2.44 | 22.5002 | 0.2 | 0.9 | | 60 | 101 | 180 | 221 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 2 | 10.19 | 2.44 | 16.9493 | 0.2 | 0.9 | | 60 | 114 | 219 | 273 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 3 | 24.89 | 3.68 | 17.4372 | 0.2 | 0.9 | | 60 | 188 | 215 | 343 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 15.2469 | 0.2 | 0.9 | | 60 | 109 | 237 | 286 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 2 | 10.19 | 2.44 | 20.5342 | 0.2 | 0.9 | | 60 | 105 | 191 | 236 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 1 | 4.57 | 2.16 | 25.5661 | 0.2 | 0.9 | | 60 | 76 | 166 | 182 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 14.1915 | 0.2 | 0.9 | | 60 | 182 | 250 | 372 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 5 | 19.27 | 2.44 | 23.63 | 0.2 | 0.9 | | 60 | 133 | 174 | 247 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 4 | 9.08 | 2.59 | 17.1668 | 0.2 | 0.9 | | 60 | 108 | 217 | 265 |
| | 90 | 30 | 30 | 0.01 | 0.05 | | 6 | 8.32 | 2.43 | 23.8287 | 0.2 | 0.9 | | 60 | 91 | 173 | 204 |

# Table B.8 Analysis Comparison (w/ Duration Detect): PolyFit and FermiFit Results

| Duration | N | #T | FAR2 | #T | En1 | Ex2 | Poly | En1 | En2 | Ex1 | Ex2 | Fermi | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 3 | 45 | 30 | 1 | 91 | 276 | | 57.7 | 107 | 247 | 303 | | 58 | 124 | 237 | 303 |
| | 3 | 21 | 30 | 1 | 130 | 344 | | 70.6 | 193 | 300 | 400 | | 71 | 191 | 280 | 400 |
| | 3 | 27 | 30 | 1 | 97 | 234 | | 15.8 | 145 | 198 | 253 | | 16 | 59.1 | 210 | 253 |
| | 3 | 22 | 30 | 1 | 137 | 492 | | 71.1 | 266 | 351 | 558 | | 71 | 249 | 380 | 558 |
| | 3 | 30 | 30 | 1 | 115 | 432 | | 73.1 | 249 | 388 | 502 | | 73 | 231 | 344 | 502 |
| | 3 | 26 | 30 | 1 | 132 | 340 | | 54.7 | 226 | 286 | 403 | | 55 | 179 | 279 | 403 |
| | 3 | 19 | 30 | 1 | 130 | 447 | | 65.1 | 254 | 331 | 514 | | 65 | 231 | 348 | 514 |
| 23 | 3 | 17 | 30 | 1 | 113 | 276 | | 67.4 | 171 | 226 | 332 | | 67 | 154 | 245 | 332 |
| | 3 | 5 | 30 | 0 | 0 | 0 | | 67.4 | 171 | 226 | 332 | | 67 | 148 | 245 | 332 |
| | 3 | 3 | 30 | 0 | 0 | 0 | | 67.4 | 171 | 226 | 332 | | 67 | 148 | 251 | 332 |
| | 3 | 5 | 0 | 1 | 63 | 77 | | 57.8 | 65.1 | 78 | 82.1 | | 60 | 67.4 | 77 | 82 |
| | 3 | 4 | 0 | 1 | 63 | 77 | | 50.4 | 70.3 | 74.7 | 83.6 | | 60 | 67 | 76.5 | 80 |
| | 3 | 4 | 0 | 1 | 65 | 81 | | 61.1 | 65.2 | 78.4 | 81.4 | | 60 | 67.5 | 76.5 | 82 |
| | 3 | 2 | 0 | 1 | 67 | 77 | | 60.4 | 64.6 | 77.7 | 86.1 | | 60 | 65.6 | 76.4 | 84 |
| | 3 | 1 | 0 | 1 | 68 | 84 | | 58.9 | 73.5 | 100 | 92.8 | | 60 | 69.2 | 84.8 | 94 |
| | 3 | 3 | 0 | 1 | 71 | 79 | | 58.9 | 73.4 | 102 | 84.9 | | 60 | 67.8 | 79.2 | 87 |
| | 3 | 3 | 0 | 1 | 66 | 73 | | 59.8 | 66.6 | 74.6 | 82.1 | | 60 | 65.9 | 75.1 | 81 |
| 3.3 | 3 | 3 | 0 | 1 | 65 | 82 | | 58.8 | 67.8 | 79 | 85.5 | | 60 | 68.4 | 76.6 | 85 |
| | 3 | 14 | 0 | 1 | 75 | 137 | | 57 | 87.6 | 132 | 167 | | 60 | 86.4 | 130 | 156 |
| | 3 | 11 | 0 | 1 | 68 | 105 | | 60.2 | 76 | 89.3 | 116 | | 60 | 69.6 | 96.4 | 106 |
| | 3 | 9 | 0 | 1 | 79 | 119 | | 56.3 | 81.1 | 120 | 141 | | 60 | 73.3 | 126 | 139 |
| | 3 | 5 | 0 | 1 | 65 | 97 | | 58.6 | 78.4 | 110 | 113 | | 60 | 68.5 | 98.5 | 107 |
| | 3 | 11 | 0 | 1 | 68 | 105 | | 60.2 | 76 | 89.3 | 116 | | 60 | 69.6 | 96.4 | 106 |
| | 3 | 14 | 0 | 1 | 75 | 137 | | 57 | 87.6 | 132 | 167 | | 60 | 86.4 | 130 | 156 |
| | 3 | 11 | 0 | 1 | 68 | 105 | | 60.2 | 76 | 89.3 | 116 | | 60 | 69.6 | 96.4 | 106 |
| | 3 | 9 | 0 | 1 | 79 | 119 | | 56.3 | 81.1 | 120 | 141 | | 60 | 73 | 126 | 139 |
| 9.75 | 3 | 5 | 0 | 1 | 65 | 97 | | 58.6 | 78.4 | 110 | 113 | | 60 | 68.5 | 98.5 | 107 |
| | 3 | 3 | 0 | 1 | 92 | 204 | | 48.1 | 173 | 150 | 240 | | 60 | 103 | 191 | 235 |
| | 3 | 20 | 0 | 1 | 80 | 164 | | 55.8 | 89.4 | 140 | 187 | | 60 | 86.7 | 143 | 180 |
| | 3 | 13 | 0 | 1 | 97 | 278 | | 45.5 | 224 | 171 | 327 | | 60 | 156 | 220 | 316 |
| | 3 | 24 | 0 | 5 | 66 | 204 | | 53.5 | 92.9 | 168 | 207 | | 60 | 100 | 162 | 202 |
| | 3 | 13 | 0 | 1 | 69 | 135 | | 58.8 | 71.2 | 134 | 146 | | 60 | 82.9 | 120 | 143 |
| | 3 | 6 | 0 | 1 | 95 | 242 | | 50.4 | 182 | 146 | 295 | | 60 | 125 | 214 | 279 |
| | 3 | 29 | 0 | 5 | 67 | 117 | | 57.4 | 74.3 | 173 | 183 | | 60 | 87.5 | 158 | 186 |
| | 3 | 10 | 0 | 1 | 90 | 151 | | 53.9 | 121 | 100 | 180 | | 60 | 87.4 | 153 | 180 |
| | 3 | 32 | 0 | 1 | 85 | 198 | | 52.3 | 110 | 170 | 232 | | 60 | 89.5 | 194 | 223 |
| | 3 | 27 | 0 | 1 | 72 | 168 | | 57.6 | 90.7 | 142 | 190 | | 60 | 88.5 | 153 | 181 |
| | 3 | 16 | 0 | 1 | 71 | 159 | | 44.6 | 95.9 | 165 | 175 | | 60 | 78.4 | 156 | 174 |
| | 3 | 10 | 0 | 1 | 82 | 172 | | 54.3 | 114 | 268 | 225 | | 60 | 97.7 | 150 | 188 |
| | 3 | 7 | 0 | 1 | 88 | 228 | | 52.3 | 163 | 134 | 276 | | 60 | 114 | 208 | 262 |
| | 3 | 15 | 0 | 1 | 68 | 145 | | 54.5 | 87 | 135 | 155 | | 60 | 89.8 | 122 | 152 |
| 15.73333333 | 3 | 11 | 0 | 1 | 80 | 143 | | 45.4 | 86.6 | 139 | 172 | | 60 | 91.7 | 135 | 167 |
| | 3 | 27 | 0 | 1 | 84 | 210 | | 54.5 | 101 | 174 | 231 | | 60 | 107 | 174 | 221 |
| | 3 | 30 | 0 | 1 | 79 | 240 | | 48.1 | 117 | 198 | 290 | | 60 | 121 | 212 | 273 |
| | 3 | 13 | 0 | 1 | 116 | 288 | | 48.6 | 218 | 198 | 352 | | 60 | 167 | 236 | 343 |
| | 3 | 29 | 0 | 2 | 170 | 241 | | 51.5 | 117 | 230 | 298 | | 60 | 131 | 215 | 286 |
| | 3 | 26 | 0 | 1 | 88 | 208 | | 54.7 | 102 | 192 | 243 | | 60 | 105 | 191 | 236 |
| | 3 | 16 | 0 | 1 | 69 | 169 | | 46.3 | 90.9 | 165 | 191 | | 60 | 86.3 | 156 | 182 |
| | 3 | 24 | 0 | 1 | 125 | 318 | | 44.5 | 199 | 177 | 382 | | 60 | 183 | 249 | 372 |
| | 3 | 12 | 0 | 1 | 105 | 205 | | 47.3 | 191 | 159 | 256 | | 60 | 106 | 201 | 247 |
| | 3 | 24 | 0 | 1 | 97 | 227 | | 48.9 | 125 | 193 | 281 | | 60 | 117 | 208 | 265 |
| 21.8 | 3 | 17 | 0 | 2 | 126 | 169 | | 57.3 | 93.3 | 162 | 205 | | 60 | 101 | 163 | 204 |

## Table B.9 Analysis Comparison (w/ FAPI Detect): Sensor and Target Information

| S Rate | Pixel X | Pixel Y | Noise | Jitter | Tar | # | Length | Width | Speed | Ibk | ITg t | En 1 | En 2 | Ex 1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 120 | 30 | 30 | 0.02 | 0.25 | | 4 | 9.08 | 2.59 | 10.7529 | 0.2 | 0.9 | 60 | 161 | 395 | 496 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 3 | 24.89 | 3.68 | 15.7323 | 0.2 | 0.9 | 60 | 250 | 289 | 479 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 6 | 8.32 | 2.43 | 22.21 | 0.2 | 0.9 | 60 | 105 | 222 | 267 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 4 | 9.08 | 2.59 | 18.131 | 0.2 | 0.9 | 60 | 120 | 259 | 319 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 4 | 9.08 | 2.59 | 16.5743 | 0.2 | 0.9 | 60 | 126 | 277 | 343 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 5 | 19.27 | 2.44 | 19.3292 | 0.2 | 0.9 | 60 | 180 | 246 | 366 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 6 | 8.32 | 2.43 | 20.0954 | 0.2 | 0.9 | 60 | 110 | 239 | 289 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 1 | 4.57 | 2.16 | 17.9799 | 0.2 | 0.9 | 60 | 91 | 260 | 291 |
| 120 | 30 | 30 | 0.02 | 0.25 | | 4 | 9.08 | 2.59 | 14.5874 | 0.2 | 0.9 | 60 | 135 | 307 | 382 |

## Table B.10 Analysis Comparison (w/ FAPI Detect): PolyFit and FermiFit Results

| FAPI | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | N | γ | #T | FAR2 | #T | En1 | Ex2 | Poly | En1 | En2 | Ex1 | Ex2 | Fermi | En1 | En2 | Ex1 | Ex2 |
| | 3 | 1 | 5 | 30 | 1 | 85 | 470 | | 55 | 155 | 396 | 507 | | 55 | 172 | 327 | 507 |
| | 3 | 1 | 1 | 30 | 1 | 98 | 441 | | 68.9 | 243 | 172 | 489 | | 69 | 225 | 323 | 489 |
| | 3 | 1 | 2 | 30 | 1 | 101 | 248 | | 78.6 | 119 | 229 | 280 | | 79 | 149 | 223 | 280 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 78.6 | 119 | 229 | 280 | | 79 | 134 | 225 | 280 |
| | 3 | 1 | 12 | 30 | 1 | 107 | 299 | | 51.1 | 132 | 290 | 315 | | 51 | 130 | 297 | 315 |
| | 3 | 1 | 2 | 30 | 1 | 85 | 339 | | 55.1 | 195 | 252 | 364 | | 55 | 137 | 222 | 364 |
| | 3 | 1 | 10 | 30 | 2 | 91 | 186 | | 73.4 | 94.6 | 208 | 234 | | 73 | 119 | 210 | 234 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 73.4 | 94.6 | 208 | 234 | | 73 | 107 | 200 | 234 |
| | 3 | 1 | 0 | 30 | 0 | 0 | 0 | | 73.4 | 94.6 | 208 | 234 | | 73 | 107 | 200 | 234 |

## Table B.11 Sample Rate Analysis Data from Analysis Comparison w/ Duration Detect

| | PE1 | Pe2 | Pex1 | Pex2 | Fen1 | Fen2 | Fex1 | Fex2 | |
|---|---|---|---|---|---|---|---|---|---|
| 10 | 0.316 | 2.353 | 3.464 | 5.141 | 0.140 | 1.862 | 3.545 | 5.03 | 2.818 |
| 30 | 0.163 | 0.208 | 0.258 | 0.246 | 0.000 | 0.106 | 0.104 | 0 | 0.219 |
| 60 | 0.126 | 0.204 | 0.314 | 0.131 | 0.000 | 0.242 | 0.250 | 0 | 0.194 |
| 90 | 0.109 | 0.170 | 0.195 | 0.111 | 0.000 | 0.126 | 0.126 | 0 | 0.146 |
| 120 | 0.103 | 0.151 | 0.165 | 0.046 | 0.102 | 0.172 | 0.166 | 0.045833333 | 0.116 |
| | 0.163 | 0.617 | 0.879 | 1.135 | | 0.502 | 0.838 | | |
| | | 0.967 | 0.82 | | | 1.029 | 0.870 | | |

## Table B.12 Pixel Size Analysis Data from Analysis comparison w/ Threshold Detect

| | PE1 | Pe2 | Pex1 | Pex2 | Fen1 | Fen2 | Fex1 | Fex2 | |
|---|---|---|---|---|---|---|---|---|---|
| 5 | 0.762 | 0.905 | 0.701 | 3.049 | 0.761 | 0.996 | 0.658 | 3.049 | 1.354 |
| 10 | 0.902 | 1.027 | 0.784 | 3.417 | 0.901 | 1.144 | 0.860 | 3.419 | 1.532 |
| 30 | 1.217 | 0.968 | 0.977 | 3.319 | 1.216 | 0.992 | 1.042 | 3.319 | 1.620 |
| | 0.960 | 0.967 | 0.820 | 3.262 | 0.959 | 1.044 | 0.854 | 3.262 | |

# Table B.13 Noise Effects Results in Detection Testing

| | Info | S Rate | Pixel X | Pixel Y | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 60 | 30 | 30 | | | | | | | | | | |
| | | | | Threshold | | | | Duration | | | | FAPI | | |
| N | # F | Act Det | Th D | Th FA | FAR D | FAR FA | Du D | Du FA | FAR D | FAR FA | FPI D | FPI FA | FAR D | FAR FA |
| 1 | 600 | 339 | 232 | 116 | 339 | 239 | 132 | 51 | 319 | 183 | 339 | 241 | 339 | 241 |
| | 600 | 155 | 50 | 136 | 155 | 416 | 100 | 244 | 155 | 440 | 95 | 204 | 110 | 240 |
| | 600 | 467 | 319 | 40 | 467 | 106 | 243 | 20 | 444 | 96 | 421 | 29 | 444 | 7 |
| | 600 | 198 | 91 | 95 | 198 | 378 | 69 | 59 | 198 | 193 | 156 | 20 | 178 | 0 |
| | 600 | 238 | 130 | 158 | 238 | 337 | 141 | 179 | 238 | 349 | 238 | 342 | 238 | 342 |
| | 600 | 241 | 132 | 110 | 241 | 330 | 92 | 52 | 241 | 204 | 237 | 165 | 241 | 275 |
| | 600 | 220 | 92 | 104 | 220 | 357 | 171 | 247 | 220 | 378 | 220 | 360 | 220 | 360 |
| | 600 | 189 | 60 | 75 | 189 | 356 | 84 | 118 | 189 | 341 | 90 | 4 | 151 | 0 |
| | 600 | 192 | 96 | 128 | 192 | 385 | 123 | 179 | 192 | 403 | 192 | 388 | 192 | 388 |
| | 600 | 245 | 93 | 98 | 245 | 322 | 24 | 0 | 159 | 0 | 227 | 242 | 245 | 300 |
| | 600 | 246 | 156 | 133 | 246 | 324 | 99 | 53 | 241 | 96 | 245 | 230 | 246 | 293 |
| | 600 | 241 | 98 | 95 | 241 | 336 | 163 | 180 | 241 | 351 | 195 | 95 | 211 | 110 |
| | 600 | 179 | 57 | 61 | 177 | 334 | 98 | 159 | 179 | 414 | 140 | 82 | 148 | 70 |
| | 600 | 184 | 97 | 151 | 184 | 393 | 26 | 0 | 129 | 0 | 184 | 396 | 184 | 396 |
| | 600 | 249 | 110 | 75 | 249 | 324 | 115 | 90 | 249 | 323 | 249 | 331 | 249 | 331 |
| | 600 | 202 | 82 | 89 | 202 | 320 | 132 | 184 | 202 | 392 | 202 | 378 | 202 | 378 |
| | 600 | 215 | 97 | 89 | 215 | 350 | 141 | 161 | 215 | 374 | 203 | 103 | 210 | 100 |
| | 600 | 158 | 54 | 100 | 158 | 418 | 29 | 23 | 125 | 0 | 99 | 57 | 135 | 156 |
| | 600 | 210 | 88 | 83 | 210 | 366 | 88 | 87 | 210 | 384 | 210 | 370 | 210 | 370 |
| | 600 | 207 | 59 | 92 | 207 | 365 | 19 | 8 | 74 | 0 | 207 | 373 | 207 | 373 |
| | 600 | 206 | 113 | 124 | 206 | 371 | 19 | 0 | 62 | 0 | 206 | 374 | 206 | 374 |
| | 600 | 468 | 297 | 31 | 468 | 101 | 183 | 0 | 408 | 0 | 468 | 112 | 468 | 112 |
| | 600 | 122 | 42 | 148 | 122 | 458 | 96 | 313 | 122 | 478 | 122 | 458 | 122 | 458 |
| | 600 | 158 | 67 | 133 | 158 | 416 | 103 | 234 | 158 | 437 | 158 | 422 | 158 | 422 |
| | 600 | 168 | 45 | 74 | 168 | 402 | 109 | 218 | 168 | 426 | 168 | 412 | 168 | 412 |
| | 600 | 151 | 52 | 84 | 151 | 426 | 26 | 23 | 111 | 32 | 151 | 429 | 151 | 429 |
| | 600 | 317 | 131 | 84 | 317 | 263 | 184 | 120 | 317 | 278 | 231 | 49 | 291 | 114 |
| | 600 | 303 | 189 | 64 | 303 | 267 | 114 | 2 | 220 | 0 | 277 | 60 | 282 | 102 |
| | 600 | 185 | 53 | 100 | 185 | 392 | 54 | 108 | 185 | 377 | 63 | 97 | 101 | 83 |
| | 600 | 167 | 59 | 87 | 167 | 407 | 107 | 216 | 167 | 423 | 118 | 49 | 159 | 6 |
| | 600 | 246 | 90 | 68 | 246 | 325 | 135 | 144 | 246 | 348 | 246 | 334 | 246 | 334 |
| | 600 | 302 | 225 | 102 | 302 | 271 | 108 | 0 | 249 | 0 | 302 | 278 | 302 | 278 |
| | 600 | 378 | 242 | 60 | 378 | 200 | 132 | 0 | 275 | 0 | 378 | 202 | 378 | 202 |
| | 600 | 238 | 90 | 108 | 238 | 339 | 145 | 199 | 238 | 355 | 238 | 342 | 238 | 342 |
| | 600 | 208 | 78 | 63 | 208 | 246 | 54 | 21 | 165 | 39 | 170 | 64 | 203 | 4 |
| | 600 | 175 | 60 | 96 | 175 | 405 | 94 | 171 | 175 | 420 | 175 | 405 | 175 | 405 |
| | 600 | 185 | 97 | 128 | 185 | 394 | 46 | 33 | 160 | 42 | 185 | 395 | 185 | 395 |
| | 600 | 194 | 79 | 103 | 194 | 376 | 136 | 205 | 194 | 399 | 149 | 51 | 185 | 102 |
| | 600 | 256 | 85 | 57 | 256 | 282 | 112 | 99 | 256 | 311 | 148 | 13 | 162 | 0 |
| | 600 | 111 | 72 | 231 | 111 | 466 | 78 | 271 | 111 | 478 | 111 | 469 | 111 | 469 |
| | 600 | 198 | 46 | 38 | 198 | 279 | 147 | 236 | 198 | 401 | 112 | 17 | 126 | 0 |
| | 600 | 251 | 91 | 74 | 251 | 326 | 194 | 214 | 251 | 343 | 182 | 50 | 209 | 0 |
| | 600 | 228 | 97 | 96 | 228 | 348 | 105 | 116 | 228 | 366 | 228 | 352 | 228 | 352 |
| | 600 | 151 | 50 | 107 | 151 | 427 | 23 | 25 | 65 | 39 | 104 | 67 | 137 | 6 |
| | 600 | 217 | 111 | 160 | 217 | 362 | 106 | 153 | 217 | 371 | 217 | 363 | 217 | 363 |
| | 600 | 131 | 36 | 67 | 131 | 435 | 34 | 62 | 111 | 294 | 131 | 449 | 131 | 449 |
| | 600 | 284 | 213 | 124 | 284 | 296 | 153 | 56 | 263 | 241 | 284 | 296 | 284 | 296 |
| | 600 | 151 | 48 | 78 | 151 | 397 | 30 | 33 | 91 | 69 | 80 | 14 | 112 | 0 |
| | 600 | 131 | 33 | 103 | 131 | 441 | 40 | 147 | 131 | 461 | 131 | 449 | 131 | 449 |
| | 600 | 228 | 87 | 117 | 228 | 340 | 43 | 28 | 145 | 35 | 228 | 352 | 228 | 352 |
| 1.5 | 600 | 148 | 44 | 100 | 144 | 397 | 7 | 7 | 0 | 0 | 51 | 59 | 90 | 141 |
| | 600 | 239 | 44 | 25 | 207 | 186 | 33 | 0 | 42 | 0 | 159 | 37 | 199 | 0 |
| | 600 | 226 | 54 | 31 | 226 | 141 | 4 | 0 | 0 | 0 | 226 | 354 | 226 | 354 |
| | 600 | 296 | 187 | 78 | 296 | 270 | 94 | 0 | 233 | 0 | 296 | 284 | 296 | 284 |
| | 600 | 237 | 84 | 48 | 237 | 294 | 84 | 50 | 237 | 196 | 237 | 343 | 237 | 343 |
| | 600 | 117 | 26 | 39 | 117 | 333 | 45 | 129 | 117 | 451 | 117 | 463 | 117 | 463 |
| | 600 | 209 | 50 | 16 | 197 | 86 | 159 | 216 | 209 | 385 | 209 | 371 | 209 | 371 |
| | 600 | 268 | 58 | 23 | 263 | 181 | 140 | 111 | 268 | 323 | 268 | 312 | 268 | 312 |
| | 600 | 405 | 206 | 17 | 400 | 95 | 127 | 0 | 322 | 0 | 405 | 175 | 405 | 175 |
| | 600 | 319 | 93 | 29 | 319 | 237 | 50 | 0 | 140 | 0 | 235 | 57 | 284 | 110 |
| | 600 | 158 | 29 | 40 | 158 | 281 | 106 | 250 | 158 | 434 | 158 | 422 | 158 | 422 |
| | 600 | 150 | 46 | 95 | 150 | 394 | 101 | 264 | 150 | 445 | 150 | 430 | 150 | 430 |
| | 600 | 138 | 20 | 20 | 104 | 88 | 29 | 64 | 104 | 193 | 138 | 442 | 138 | 442 |
| | 600 | 210 | 42 | 23 | 198 | 42 | 124 | 194 | 210 | 383 | 210 | 370 | 210 | 370 |
| | 600 | 381 | 176 | 25 | 381 | 168 | 108 | 0 | 226 | 0 | 381 | 199 | 381 | 199 |
| | 600 | 280 | 112 | 52 | 280 | 293 | 89 | 23 | 248 | 18 | 280 | 300 | 280 | 300 |
| | 600 | 295 | 156 | 59 | 295 | 282 | 82 | 0 | 209 | 0 | 295 | 285 | 295 | 285 |
| | 600 | 239 | 75 | 49 | 239 | 329 | 118 | 112 | 239 | 348 | 239 | 341 | 239 | 341 |
| | 600 | 146 | 27 | 11 | 132 | 79 | 24 | 0 | 93 | 0 | 107 | 27 | 136 | 0 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 217 | 62 | 71 | 217 | 327 | 98 | 131 | 217 | 359 | 217 | 363 | 217 | 363 |
| | 600 | 186 | 71 | 96 | 186 | 394 | 131 | 235 | 186 | 406 | 186 | 394 | 186 | 394 |
| | 600 | 316 | 157 | 81 | 316 | 252 | 75 | 0 | 231 | 0 | 266 | 76 | 309 | 163 |
| | 600 | 250 | 46 | 19 | 243 | 126 | 38 | 9 | 171 | 0 | 250 | 330 | 250 | 330 |
| | 600 | 228 | 49 | 9 | 219 | 24 | 38 | 0 | 167 | 0 | 168 | 1 | 178 | 0 |
| | 600 | 181 | 59 | 87 | 181 | 394 | 128 | 232 | 181 | 412 | 181 | 399 | 181 | 399 |
| | 600 | 185 | 43 | 66 | 185 | 321 | 41 | 64 | 167 | 328 | 185 | 395 | 185 | 395 |
| | 600 | 179 | 41 | 47 | 162 | 284 | 87 | 154 | 179 | 395 | 110 | 32 | 97 | 0 |
| | 600 | 268 | 51 | 20 | 268 | 115 | 84 | 73 | 254 | 293 | 48 | 0 | 91 | 0 |
| | 600 | 250 | 107 | 99 | 250 | 325 | 157 | 179 | 250 | 344 | 250 | 330 | 250 | 330 |
| | 600 | 181 | 57 | 17 | 174 | 161 | 119 | 173 | 181 | 414 | 181 | 399 | 181 | 399 |
| | 600 | 226 | 107 | 59 | 226 | 302 | 76 | 29 | 206 | 35 | 226 | 354 | 226 | 354 |
| | 600 | 136 | 27 | 57 | 136 | 369 | 85 | 279 | 136 | 464 | 136 | 444 | 136 | 444 |
| | 600 | 252 | 91 | 46 | 252 | 261 | 140 | 122 | 252 | 340 | 252 | 328 | 252 | 328 |
| | 600 | 308 | 125 | 10 | 284 | 0 | 77 | 0 | 212 | 0 | 230 | 5 | 237 | 0 |
| | 600 | 283 | 150 | 40 | 265 | 260 | 91 | 0 | 211 | 0 | 283 | 297 | 283 | 297 |
| | 600 | 244 | 152 | 108 | 244 | 334 | 96 | 37 | 241 | 101 | 244 | 336 | 244 | 336 |
| | 600 | 212 | 49 | 49 | 212 | 286 | 7 | 0 | 0 | 0 | 212 | 368 | 212 | 368 |
| | 600 | 382 | 173 | 11 | 369 | 45 | 114 | 0 | 261 | 0 | 306 | 0 | 327 | 0 |
| | 600 | 122 | 32 | 92 | 122 | 455 | 58 | 194 | 122 | 468 | 15 | 1 | 0 | 0 |
| | 600 | 212 | 73 | 109 | 212 | 361 | 94 | 169 | 212 | 381 | 8 | 6 | 0 | 0 |
| | 600 | 249 | 98 | 66 | 249 | 286 | 153 | 126 | 249 | 343 | 249 | 331 | 249 | 331 |
| | 600 | 304 | 204 | 74 | 304 | 276 | 136 | 24 | 289 | 13 | 304 | 276 | 304 | 276 |
| | 600 | 253 | 37 | 2 | 175 | 0 | 94 | 80 | 253 | 295 | 38 | 2 | 0 | 0 |
| | 600 | 255 | 66 | 25 | 223 | 224 | 53 | 0 | 153 | 0 | 255 | 325 | 255 | 325 |
| | 600 | 144 | 32 | 48 | 144 | 367 | 57 | 118 | 144 | 448 | 73 | 6 | 0 | 0 |
| | 600 | 251 | 95 | 104 | 251 | 323 | 57 | 42 | 218 | 112 | 251 | 329 | 251 | 329 |
| | 600 | 180 | 96 | 114 | 180 | 400 | 107 | 162 | 180 | 411 | 180 | 400 | 180 | 400 |
| | 600 | 186 | 39 | 49 | 186 | 373 | 108 | 190 | 186 | 406 | 186 | 394 | 186 | 394 |
| 2 | 600 | 120 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 120 | 460 | 120 | 460 |
| | 600 | 229 | 34 | 2 | 140 | 0 | 59 | 57 | 213 | 225 | 229 | 351 | 229 | 351 |
| | 600 | 141 | 15 | 2 | 103 | 0 | 34 | 88 | 117 | 321 | 141 | 439 | 141 | 439 |
| | 600 | 177 | 31 | 0 | 99 | 0 | 84 | 96 | 177 | 329 | 108 | 1 | 118 | 0 |
| | 600 | 124 | 23 | 47 | 124 | 12 | 67 | 231 | 124 | 471 | 62 | 56 | 85 | 58 |
| | 600 | 200 | 33 | 0 | 131 | 0 | 32 | 0 | 113 | 0 | 200 | 380 | 200 | 380 |
| | 600 | 145 | 0 | 0 | 0 | 0 | 110 | 296 | 145 | 448 | 145 | 435 | 145 | 435 |
| | 600 | 196 | 37 | 19 | 184 | 15 | 109 | 159 | 196 | 398 | 196 | 384 | 196 | 384 |
| | 600 | 274 | 22 | 0 | 171 | 0 | 142 | 138 | 274 | 319 | 274 | 306 | 274 | 306 |
| | 600 | 292 | 46 | 0 | 195 | 0 | 78 | 12 | 219 | 0 | 292 | 288 | 292 | 288 |
| | 600 | 116 | 4 | 0 | 0 | 0 | 10 | 16 | 0 | 0 | 7 | 30 | 0 | 0 |
| | 600 | 244 | 33 | 0 | 162 | 0 | 28 | 0 | 70 | 0 | 244 | 336 | 244 | 336 |
| | 600 | 254 | 48 | 0 | 169 | 0 | 68 | 13 | 151 | 0 | 66 | 0 | 68 | 0 |
| | 600 | 226 | 30 | 32 | 198 | 21 | 58 | 68 | 212 | 186 | 0 | 0 | 0 | 0 |
| | 600 | 218 | 10 | 0 | 95 | 0 | 0 | 0 | 0 | 0 | 22 | 0 | 0 | 0 |
| | 600 | 203 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 203 | 377 | 203 | 377 |
| | 600 | 186 | 15 | 0 | 122 | 0 | 97 | 159 | 186 | 409 | 10 | 0 | 0 | 0 |
| | 600 | 275 | 64 | 0 | 190 | 0 | 54 | 0 | 143 | 0 | 199 | 0 | 207 | 0 |
| | 600 | 152 | 5 | 0 | 0 | 0 | 37 | 81 | 135 | 313 | 152 | 428 | 152 | 428 |
| | 600 | 230 | 7 | 0 | 0 | 0 | 138 | 146 | 230 | 362 | 230 | 350 | 230 | 350 |
| | 600 | 184 | 20 | 0 | 141 | 0 | 29 | 21 | 109 | 0 | 184 | 396 | 184 | 396 |
| | 600 | 250 | 20 | 0 | 150 | 0 | 68 | 56 | 208 | 281 | 250 | 330 | 250 | 330 |
| | 600 | 345 | 25 | 0 | 202 | 0 | 65 | 16 | 284 | 32 | 190 | 10 | 260 | 0 |
| | 600 | 177 | 24 | 0 | 112 | 0 | 93 | 142 | 177 | 417 | 177 | 403 | 177 | 403 |
| | 600 | 144 | 22 | 22 | 144 | 12 | 70 | 178 | 144 | 447 | 144 | 436 | 144 | 436 |
| | 600 | 187 | 9 | 0 | 75 | 0 | 33 | 22 | 124 | 44 | 187 | 393 | 187 | 393 |
| | 600 | 173 | 33 | 55 | 173 | 30 | 0 | 0 | 0 | 0 | 173 | 407 | 173 | 407 |
| | 600 | 203 | 50 | 34 | 203 | 30 | 37 | 13 | 149 | 0 | 203 | 377 | 203 | 377 |
| | 600 | 149 | 5 | 0 | 0 | 0 | 80 | 169 | 149 | 446 | 149 | 431 | 149 | 431 |
| | 600 | 212 | 21 | 13 | 88 | 10 | 155 | 252 | 212 | 381 | 212 | 368 | 212 | 368 |
| | 600 | 163 | 21 | 0 | 135 | 0 | 70 | 124 | 162 | 412 | 163 | 417 | 163 | 417 |
| | 600 | 223 | 37 | 0 | 176 | 0 | 127 | 100 | 223 | 370 | 223 | 357 | 223 | 357 |
| | 600 | 280 | 65 | 41 | 275 | 35 | 17 | 0 | 0 | 0 | 101 | 4 | 164 | 0 |
| | 600 | 331 | 192 | 48 | 331 | 48 | 76 | 0 | 218 | 0 | 287 | 4 | 327 | 1 |
| | 600 | 263 | 20 | 0 | 118 | 0 | 48 | 19 | 165 | 0 | 263 | 317 | 263 | 317 |
| | 600 | 332 | 125 | 4 | 286 | 0 | 74 | 0 | 172 | 0 | 332 | 248 | 332 | 248 |
| | 600 | 208 | 54 | 57 | 208 | 51 | 84 | 111 | 208 | 362 | 208 | 372 | 208 | 372 |
| | 600 | 214 | 38 | 0 | 162 | 0 | 10 | 0 | 32 | 0 | 214 | 366 | 214 | 366 |
| | 600 | 460 | 207 | 4 | 427 | 41 | 97 | 0 | 246 | 0 | 460 | 120 | 460 | 120 |
| | 600 | 124 | 24 | 73 | 124 | 50 | 9 | 15 | 32 | 0 | 124 | 456 | 124 | 456 |
| | 600 | 171 | 62 | 74 | 171 | 6- | 107 | 191 | 171 | 419 | 171 | 409 | 171 | 409 |
| | 600 | 325 | 59 | 19 | 317 | 8 | 19 | 0 | 0 | 0 | 269 | 26 | 307 | 0 |
| | 600 | 228 | 10 | 0 | 61 | 0 | 1 | 0 | 0 | 0 | 228 | 352 | 228 | 352 |
| | 600 | 138 | 13 | 0 | 97 | 0 | 11 | 0 | 56 | 0 | 138 | 442 | 138 | 442 |
| | 600 | 263 | 91 | 6 | 234 | 31 | 77 | 0 | 198 | 0 | 263 | 317 | 263 | 317 |
| | 600 | 284 | 77 | 0 | 214 | 0 | 73 | 0 | 168 | 0 | 284 | 296 | 284 | 296 |
| | 600 | 345 | 117 | 0 | 268 | 0 | 89 | 0 | 222 | 0 | 345 | 235 | 345 | 235 |
| | 600 | 240 | 52 | 32 | 240 | 28 | 44 | 29 | 187 | 93 | 240 | 340 | 240 | 340 |
| | 600 | 374 | 150 | 0 | 349 | 0 | 84 | 0 | 251 | 0 | 322 | 7 | 323 | 0 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 195 | 57 | 49 | 195 | 30 | 110 | 136 | 195 | 372 | 195 | 385 | 195 | 385 |
| 2.5 | 600 | 233 | 33 | 0 | 134 | 0 | 108 | 98 | 217 | 360 | 233 | 347 | 233 | 347 |
| | 600 | 146 | 0 | 0 | 0 | 0 | 32 | 79 | 114 | 269 | 146 | 434 | 146 | 434 |
| | 600 | 435 | 124 | 0 | 334 | 0 | 128 | 0 | 310 | 0 | 371 | 35 | 360 | 0 |
| | 600 | 237 | 26 | 0 | 118 | 0 | 104 | 97 | 230 | 318 | 114 | 0 | 120 | 0 |
| | 600 | 236 | 1 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 236 | 344 | 236 | 344 |
| | 600 | 239 | 6 | 0 | 0 | 0 | 113 | 128 | 239 | 340 | 239 | 341 | 239 | 341 |
| | 600 | 369 | 90 | 0 | 249 | 0 | 58 | 0 | 198 | 0 | 369 | 211 | 369 | 211 |
| | 600 | 258 | 7 | 0 | 35 | 0 | 64 | 40 | 233 | 172 | 62 | 0 | 0 | 0 |
| | 600 | 241 | 15 | 0 | 96 | 0 | 18 | 0 | 76 | 0 | 241 | 339 | 241 | 339 |
| | 600 | 183 | 4 | 0 | 0 | 0 | 45 | 51 | 183 | 207 | 47 | 9 | 70 | 0 |
| | 600 | 238 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 238 | 342 | 238 | 342 |
| | 600 | 162 | 8 | 0 | 87 | 0 | 0 | 0 | 0 | 0 | 162 | 418 | 162 | 418 |
| | 600 | 370 | 106 | 0 | 290 | 0 | 53 | 0 | 187 | 0 | 370 | 210 | 370 | 210 |
| | 600 | 196 | 6 | 0 | 52 | 0 | 117 | 184 | 196 | 396 | 7 | 0 | 0 | 0 |
| | 600 | 316 | 120 | 1 | 301 | 0 | 50 | 0 | 166 | 0 | 291 | 41 | 302 | 0 |
| | 600 | 373 | 97 | 0 | 267 | 0 | 26 | 0 | 94 | 0 | 293 | 0 | 295 | 0 |
| | 600 | 305 | 43 | 0 | 121 | 0 | 47 | 0 | 121 | 0 | 305 | 275 | 305 | 275 |
| | 600 | 170 | 0 | 0 | 0 | 0 | 44 | 82 | 148 | 358 | 170 | 410 | 170 | 410 |
| | 600 | 277 | 52 | 0 | 175 | 0 | 79 | 0 | 209 | 0 | 214 | 1 | 221 | 0 |
| | 600 | 138 | 9 | 0 | 76 | 0 | 62 | 157 | 138 | 435 | 138 | 442 | 138 | 442 |
| | 600 | 275 | 55 | 0 | 134 | 0 | 51 | 0 | 134 | 0 | 195 | 6 | 214 | 0 |
| | 600 | 286 | 46 | 3 | 157 | 41 | 169 | 147 | 286 | 308 | 286 | 294 | 286 | 294 |
| | 600 | 308 | 23 | 0 | 161 | 0 | 140 | 65 | 308 | 227 | 308 | 272 | 308 | 272 |
| | 600 | 392 | 68 | 0 | 252 | 0 | 55 | 0 | 172 | 0 | 291 | 0 | 282 | 0 |
| | 600 | 347 | 140 | 0 | 293 | 0 | 54 | 0 | 185 | 0 | 294 | 0 | 307 | 0 |
| | 600 | 380 | 104 | 0 | 308 | 0 | 58 | 0 | 217 | 0 | 380 | 200 | 380 | 200 |
| | 600 | 198 | 6 | 0 | 0 | 0 | 94 | 125 | 198 | 394 | 198 | 382 | 198 | 382 |
| | 600 | 191 | 0 | 0 | 0 | 0 | 26 | 34 | 141 | 121 | 191 | 389 | 191 | 389 |
| | 600 | 205 | 0 | 0 | 0 | 0 | 92 | 145 | 205 | 350 | 205 | 375 | 205 | 375 |
| | 600 | 142 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 303 | 81 | 0 | 194 | 0 | 98 | 0 | 235 | 0 | 303 | 277 | 303 | 277 |
| | 600 | 311 | 51 | 0 | 243 | 0 | 99 | 43 | 294 | 159 | 181 | 0 | 206 | 0 |
| | 600 | 273 | 92 | 0 | 214 | 0 | 58 | 0 | 160 | 0 | 273 | 307 | 273 | 307 |
| | 600 | 137 | 0 | 0 | 0 | 0 | 6 | 0 | 0 | 0 | 10 | 8 | 0 | 0 |
| | 600 | 228 | 0 | 0 | 0 | 0 | 141 | 170 | 228 | 366 | 228 | 352 | 228 | 352 |
| | 600 | 268 | 28 | 0 | 180 | 0 | 132 | 120 | 268 | 327 | 57 | 0 | 65 | 0 |
| | 600 | 338 | 23 | 0 | 165 | 0 | 36 | 2 | 136 | 0 | 338 | 242 | 338 | 242 |
| | 600 | 131 | 0 | 0 | 0 | 0 | 63 | 241 | 131 | 463 | 131 | 449 | 131 | 449 |
| | 600 | 226 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 226 | 354 | 226 | 354 |
| | 600 | 403 | 138 | 0 | 318 | 0 | 71 | 0 | 172 | 0 | 403 | 177 | 403 | 177 |
| | 600 | 295 | 24 | 0 | 99 | 0 | 23 | 0 | 70 | 0 | 295 | 285 | 295 | 285 |
| | 600 | 162 | 15 | 0 | 84 | 0 | 97 | 189 | 162 | 433 | 162 | 418 | 162 | 418 |
| | 600 | 217 | 0 | 0 | 0 | 0 | 33 | 9 | 132 | 0 | 217 | 363 | 217 | 363 |
| | 600 | 200 | 11 | 0 | 73 | 0 | 21 | 0 | 74 | 0 | 136 | 6 | 153 | 0 |
| | 600 | 284 | 16 | 0 | 79 | 0 | 60 | 1 | 208 | 0 | 86 | 0 | 111 | 0 |
| | 600 | 367 | 127 | 0 | 289 | 0 | 56 | 0 | 173 | 0 | 305 | 0 | 307 | 0 |
| | 600 | 138 | 0 | 0 | 0 | 0 | 64 | 183 | 138 | 455 | 138 | 442 | 138 | 442 |
| | 600 | 237 | 18 | 0 | 131 | 0 | 59 | 41 | 223 | 0 | 237 | 343 | 237 | 343 |
| | 600 | 285 | 84 | 0 | 204 | 0 | 81 | 0 | 204 | 0 | 285 | 295 | 285 | 295 |
| | 600 | 204 | 0 | 0 | 0 | 0 | 105 | 144 | 204 | 387 | 151 | 25 | 186 | 0 |
| 3 | 600 | 270 | 35 | 0 | 131 | 0 | 29 | 0 | 131 | 0 | 213 | 12 | 224 | 0 |
| | 600 | 173 | 1 | 0 | 0 | 0 | 71 | 131 | 173 | 421 | 173 | 407 | 173 | 407 |
| | 600 | 197 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 154 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 0 | 0 |
| | 600 | 307 | 52 | 0 | 192 | 0 | 64 | 0 | 207 | 0 | 177 | 0 | 196 | 0 |
| | 600 | 170 | 7 | 0 | 0 | 0 | 14 | 0 | 44 | 0 | 170 | 410 | 170 | 410 |
| | 600 | 218 | 0 | 0 | 0 | 0 | 20 | 0 | 82 | 0 | 218 | 362 | 218 | 362 |
| | 600 | 208 | 21 | 0 | 84 | 0 | 83 | 72 | 208 | 292 | 208 | 372 | 208 | 372 |
| | 600 | 328 | 0 | 0 | 0 | 0 | 96 | 50 | 295 | 249 | 328 | 252 | 328 | 252 |
| | 600 | 303 | 43 | 0 | 166 | 0 | 40 | 0 | 128 | 0 | 303 | 277 | 303 | 277 |
| | 600 | 224 | 38 | 0 | 156 | 0 | 67 | 21 | 224 | 25 | 224 | 356 | 224 | 356 |
| | 600 | 276 | 10 | 0 | 68 | 0 | 28 | 0 | 111 | 0 | 276 | 304 | 276 | 304 |
| | 600 | 168 | 0 | 0 | 0 | 0 | 46 | 81 | 168 | 354 | 168 | 412 | 168 | 412 |
| | 600 | 179 | 0 | 0 | 0 | 0 | 53 | 19 | 145 | 0 | 179 | 401 | 179 | 401 |
| | 600 | 216 | 8 | 0 | 57 | 0 | 91 | 120 | 216 | 366 | 216 | 364 | 216 | 364 |
| | 600 | 174 | 0 | 0 | 0 | 0 | 69 | 135 | 174 | 419 | 174 | 406 | 174 | 406 |
| | 600 | 447 | 111 | 0 | 309 | 0 | 166 | 0 | 316 | 0 | 275 | 0 | 301 | 0 |
| | 600 | 478 | 75 | 0 | 249 | 0 | 124 | 0 | 310 | 0 | 284 | 0 | 294 | 0 |
| | 600 | 250 | 1 | 0 | 0 | 0 | 136 | 147 | 250 | 344 | 250 | 330 | 250 | 330 |
| | 600 | 167 | 0 | 0 | 0 | 0 | 29 | 45 | 138 | 207 | 167 | 413 | 167 | 413 |
| | 600 | 316 | 46 | 0 | 211 | 0 | 50 | 0 | 211 | 0 | 316 | 264 | 316 | 264 |
| | 600 | 225 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 225 | 355 | 225 | 355 |
| | 600 | 194 | 0 | 0 | 0 | 0 | 63 | 94 | 194 | 346 | 0 | 0 | 0 | 0 |
| | 600 | 252 | 10 | 0 | 47 | 0 | 151 | 125 | 252 | 342 | 252 | 328 | 252 | 328 |
| | 600 | 188 | 0 | 0 | 0 | 0 | 90 | 124 | 184 | 396 | 0 | 0 | 0 | 0 |
| | 600 | 225 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 225 | 355 | 225 | 355 |
| | 600 | 210 | 0 | 0 | 0 | 0 | 19 | 0 | 96 | 0 | 210 | 370 | 210 | 370 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 122 | 0 | 0 | 0 | 0 | 61 | 223 | 122 | 470 | 122 | 458 | 122 | 458 |
| | 600 | 158 | 0 | 0 | 0 | 0 | 20 | 0 | 68 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 140 | 0 | 0 | 0 | 0 | 55 | 155 | 140 | 455 | 1 | 0 | 0 | 0 |
| | 600 | 150 | 7 | 0 | 71 | 0 | 26 | 20 | 140 | 0 | 150 | 430 | 150 | 430 |
| | 600 | 108 | 0 | 0 | 0 | 0 | 64 | 263 | 108 | 484 | 0 | 0 | 0 | 0 |
| | 600 | 292 | 6 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 292 | 288 | 292 | 288 |
| | 600 | 244 | 0 | 0 | 0 | 0 | 17 | 0 | 0 | 0 | 6 | 0 | 0 | 0 |
| | 600 | 183 | 26 | 0 | 148 | 0 | 36 | 28 | 161 | 59 | 25 | 0 | 0 | 0 |
| | 600 | 197 | 0 | 0 | 0 | 0 | 46 | 14 | 131 | 0 | 197 | 383 | 197 | 383 |
| | 600 | 219 | 12 | 0 | 105 | 0 | 8 | 0 | 0 | 0 | 219 | 361 | 219 | 361 |
| | 600 | 440 | 91 | 0 | 278 | 0 | 59 | 0 | 196 | 0 | 440 | 140 | 440 | 140 |
| | 600 | 365 | 77 | 0 | 207 | 0 | 84 | 0 | 166 | 0 | 255 | 0 | 267 | 0 |
| | 600 | 165 | 0 | 0 | 0 | 0 | 39 | 20 | 108 | 37 | 165 | 415 | 165 | 415 |
| | 600 | 207 | 5 | 0 | 30 | 0 | 9 | 0 | 56 | 0 | 9 | 0 | 0 | 0 |
| | 600 | 249 | 7 | 0 | 56 | 0 | 59 | 11 | 197 | 0 | 249 | 331 | 249 | 331 |
| | 600 | 245 | 0 | 0 | 0 | 0 | 106 | 104 | 229 | 349 | 245 | 335 | 245 | 335 |
| | 600 | 149 | 0 | 0 | 0 | 0 | 78 | 190 | 149 | 446 | 149 | 431 | 149 | 431 |
| | 600 | 275 | 24 | 0 | 124 | 0 | 46 | 0 | 161 | 0 | 118 | 0 | 131 | 0 |
| | 600 | 153 | 7 | 0 | 81 | 0 | 0 | 0 | 0 | 0 | 153 | 427 | 153 | 427 |
| | 600 | 363 | 49 | 0 | 161 | 0 | 71 | 0 | 193 | 0 | 265 | 0 | 307 | 0 |
| | 600 | 117 | 0 | 0 | 0 | 0 | 9 | 18 | 0 | 0 | 117 | 463 | 117 | 463 |
| | 600 | 280 | 0 | 0 | 0 | 0 | 141 | 128 | 280 | 310 | 280 | 300 | 280 | 300 |
| | 600 | 455 | 125 | 0 | 329 | 0 | 68 | 0 | 240 | 0 | 455 | 125 | 455 | 125 |
| 3 5 | 600 | 394 | 15 | 0 | 87 | 0 | 28 | 0 | 61 | 0 | 394 | 186 | 394 | 186 |
| | 600 | 147 | 0 | 0 | 0 | 0 | 19 | 0 | 0 | 0 | 147 | 433 | 147 | 433 |
| | 600 | 128 | 0 | 0 | 0 | 0 | 54 | 154 | 128 | 458 | 128 | 452 | 128 | 452 |
| | 600 | 162 | 0 | 0 | 0 | 0 | 79 | 159 | 162 | 432 | 23 | 0 | 0 | 0 |
| | 600 | 276 | 0 | 0 | 0 | 0 | 11 | 0 | 0 | 0 | 276 | 304 | 276 | 304 |
| | 600 | 317 | 0 | 0 | 0 | 0 | 11 | 0 | 43 | 0 | 5 | 0 | 0 | 0 |
| | 600 | 277 | 0 | 0 | 0 | 0 | 22 | 0 | 95 | 0 | 101 | 0 | 108 | 0 |
| | 600 | 240 | 0 | 0 | 0 | 0 | 37 | 11 | 172 | 32 | 161 | 28 | 227 | 0 |
| | 600 | 178 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 178 | 402 | 178 | 402 |
| | 600 | 250 | 0 | 0 | 0 | 0 | 117 | 121 | 250 | 345 | 39 | 0 | 91 | 0 |
| | 600 | 109 | 0 | 0 | 0 | 0 | 38 | 159 | 109 | 485 | 109 | 471 | 109 | 471 |
| | 600 | 303 | 0 | 0 | 0 | 0 | 72 | 37 | 262 | 126 | 303 | 277 | 303 | 277 |
| | 600 | 289 | 0 | 0 | 0 | 0 | 106 | 77 | 287 | 270 | 38 | 0 | 0 | 0 |
| | 600 | 241 | 0 | 0 | 0 | 0 | 33 | 0 | 121 | 0 | 112 | 2 | 98 | 0 |
| | 600 | 166 | 0 | 0 | 0 | 0 | 74 | 151 | 166 | 413 | 166 | 414 | 166 | 414 |
| | 600 | 202 | 0 | 0 | 0 | 0 | 99 | 134 | 202 | 391 | 174 | 78 | 202 | 91 |
| | 600 | 308 | 31 | 0 | 131 | 0 | 64 | 0 | 201 | 0 | 308 | 272 | 308 | 272 |
| | 600 | 277 | 57 | 0 | 171 | 0 | 10 | 0 | 38 | 0 | 106 | 0 | 119 | 0 |
| | 600 | 245 | 3 | 0 | 0 | 0 | 11 | 0 | 30 | 0 | 49 | 0 | 75 | 0 |
| | 600 | 241 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 241 | 339 | 241 | 339 |
| | 600 | 178 | 0 | 0 | 0 | 0 | 46 | 48 | 162 | 247 | 178 | 402 | 178 | 402 |
| | 600 | 187 | 0 | 0 | 0 | 0 | 51 | 40 | 187 | 111 | 5 | 0 | 0 | 0 |
| | 600 | 338 | 47 | 0 | 148 | 0 | 49 | 0 | 148 | 0 | 161 | 0 | 164 | 0 |
| | 600 | 312 | 11 | 0 | 45 | 0 | 26 | 0 | 90 | 0 | 312 | 268 | 312 | 268 |
| | 600 | 259 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 259 | 321 | 259 | 321 |
| | 600 | 128 | 0 | 0 | 0 | 0 | 69 | 186 | 128 | 464 | 128 | 452 | 128 | 452 |
| | 600 | 212 | 0 | 0 | 0 | 0 | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 267 | 65 | 0 | 172 | 0 | 14 | 0 | 69 | 0 | 267 | 313 | 267 | 313 |
| | 600 | 302 | 0 | 0 | 0 | 0 | 40 | 0 | 205 | 0 | 302 | 278 | 302 | 278 |
| | 600 | 227 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 58 | 0 | 110 | 0 |
| | 600 | 159 | 0 | 0 | 0 | 0 | 12 | 0 | 0 | 0 | 159 | 421 | 159 | 421 |
| | 600 | 136 | 0 | 0 | 0 | 0 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 144 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 144 | 436 | 144 | 436 |
| | 600 | 191 | 0 | 0 | 0 | 0 | 17 | 0 | 56 | 0 | 167 | 233 | 191 | 283 |
| | 600 | 167 | 0 | 0 | 0 | 0 | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| | 600 | 287 | 0 | 0 | 0 | 0 | 57 | 0 | 167 | 0 | 287 | 293 | 287 | 293 |
| | 600 | 158 | 0 | 0 | 0 | 0 | 53 | 105 | 158 | 381 | 158 | 422 | 158 | 422 |
| | 600 | 166 | 0 | 0 | 0 | 0 | 81 | 174 | 166 | 426 | 0 | 0 | 0 | 0 |
| | 600 | 175 | 0 | 0 | 0 | 0 | 23 | 0 | 105 | 0 | 175 | 405 | 175 | 405 |
| | 600 | 171 | 0 | 0 | 0 | 0 | 22 | 0 | 60 | 0 | 171 | 409 | 171 | 409 |
| | 600 | 209 | 0 | 0 | 0 | 0 | 20 | 1 | 91 | 0 | 209 | 371 | 209 | 371 |
| | 600 | 237 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 237 | 343 | 237 | 343 |
| | 600 | 249 | 0 | 0 | 0 | 0 | 21 | 0 | 108 | 0 | 249 | 331 | 249 | 331 |
| | 600 | 195 | 0 | 0 | 0 | 0 | 24 | 9 | 141 | 0 | 195 | 385 | 195 | 385 |
| | 600 | 272 | 0 | 0 | 0 | 0 | 98 | 76 | 272 | 281 | 5 | 0 | 0 | 0 |
| | 600 | 391 | 29 | 0 | 117 | 0 | 30 | 0 | 117 | 0 | 391 | 189 | 391 | 189 |
| | 600 | 242 | 0 | 0 | 0 | 0 | 66 | 28 | 240 | 47 | 46 | 0 | 62 | 0 |
| | 600 | 166 | 0 | 0 | 0 | 0 | 60 | 109 | 166 | 372 | 31 | 0 | 0 | 0 |
| | 600 | 284 | 0 | 0 | 0 | 0 | 95 | 39 | 257 | 140 | 284 | 296 | 284 | 296 |
| | 600 | 333 | 0 | 0 | 0 | 0 | 9 | 0 | 0 | 0 | 333 | 247 | 333 | 247 |
| 4 | 600 | 217 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 217 | 363 | 217 | 363 |
| | 600 | 199 | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 97 | 0 | 114 | 0 |
| | 600 | 185 | 0 | 0 | 0 | 0 | 10 | 0 | 30 | 0 | 152 | 43 | 175 | 0 |
| | 600 | 255 | 0 | 0 | 0 | 0 | 14 | 0 | 64 | 0 | 92 | 0 | 96 | 0 |
| | 600 | 352 | 6 | 0 | 30 | 0 | 17 | 0 | 46 | 0 | 276 | 1 | 305 | 0 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 217 | 0 | 0 | 0 | 0 | 47 | 19 | 188 | 1 | 217 | 363 | 217 | 363 |
| | 600 | 166 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 166 | 414 | 166 | 414 |
| | 600 | 182 | 0 | 0 | 0 | 0 | 47 | 104 | 131 | 370 | 7 | 2 | 0 | 0 |
| | 600 | 127 | 0 | 0 | 0 | 0 | 62 | 188 | 127 | 466 | 127 | 453 | 127 | 453 |
| | 600 | 191 | 0 | 0 | 0 | 0 | 64 | 78 | 191 | 292 | 191 | 389 | 191 | 389 |
| | 600 | 152 | 0 | 0 | 0 | 0 | 68 | 127 | 152 | 434 | 0 | 0 | 0 | 0 |
| | 600 | 146 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 146 | 434 | 146 | 434 |
| | 600 | 221 | 13 | 0 | 95 | 0 | 72 | 15 | 189 | 7 | 221 | 359 | 221 | 359 |
| | 600 | 455 | 21 | 0 | 119 | 0 | 59 | 0 | 191 | 0 | 236 | 0 | 236 | 0 |
| | 600 | 110 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 110 | 470 | 110 | 470 |
| | 600 | 108 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 0 |
| | 600 | 178 | 0 | 0 | 0 | 0 | 38 | 20 | 154 | 16 | 178 | 402 | 178 | 402 |
| | 600 | 213 | 0 | 0 | 0 | 0 | 21 | 0 | 32 | 0 | 213 | 367 | 213 | 367 |
| | 600 | 146 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 190 | 2 | 0 | 0 | 0 | 44 | 14 | 161 | 4 | 190 | 390 | 190 | 390 |
| | 600 | 466 | 24 | 0 | 135 | 0 | 16 | 0 | 69 | 0 | 466 | 114 | 466 | 114 |
| | 600 | 257 | 16 | 0 | 60 | 0 | 11 | 0 | 40 | 0 | 257 | 323 | 257 | 323 |
| | 600 | 303 | 43 | 0 | 129 | 0 | 17 | 0 | 82 | 0 | 303 | 277 | 303 | 277 |
| | 600 | 341 | 7 | 0 | 40 | 0 | 36 | 0 | 126 | 0 | 161 | 0 | 175 | 0 |
| | 600 | 253 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 253 | 327 | 253 | 327 |
| | 600 | 239 | 0 | 0 | 0 | 0 | 72 | 72 | 223 | 293 | 239 | 341 | 239 | 341 |
| | 600 | 152 | 0 | 0 | 0 | 0 | 25 | 26 | 118 | 4 | 49 | 17 | 104 | 0 |
| | 600 | 232 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | 0 | 0 | 0 |
| | 600 | 261 | 0 | 0 | 0 | 0 | 44 | 0 | 113 | 0 | 63 | 0 | 110 | 0 |
| | 600 | 220 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 177 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 177 | 403 | 177 | 403 |
| | 600 | 211 | 0 | 0 | 0 | 0 | 14 | 0 | 76 | 0 | 30 | 0 | 78 | 0 |
| | 600 | 146 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 243 | 0 | 0 | 0 | 0 | 40 | 20 | 182 | 35 | 0 | 0 | 0 | 0 |
| | 600 | 215 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| | 600 | 127 | 0 | 0 | 0 | 0 | 42 | 115 | 127 | 406 | 127 | 453 | 127 | 453 |
| | 600 | 224 | 0 | 0 | 0 | 0 | 10 | 0 | 0 | 0 | 224 | 356 | 224 | 356 |
| | 600 | 299 | 7 | 0 | 0 | 0 | 15 | 0 | 58 | 0 | 299 | 281 | 299 | 281 |
| | 600 | 173 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 173 | 407 | 173 | 407 |
| | 600 | 178 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 178 | 402 | 178 | 402 |
| | 600 | 306 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 306 | 274 | 306 | 274 |
| | 600 | 308 | 8 | 0 | 38 | 0 | 19 | 0 | 59 | 0 | 308 | 272 | 308 | 272 |
| | 600 | 205 | 0 | 0 | 0 | 0 | 47 | 56 | 176 | 143 | 205 | 375 | 205 | 375 |
| | 600 | 218 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 218 | 362 | 218 | 362 |
| | 600 | 336 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 336 | 244 | 336 | 244 |
| | 600 | 109 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 149 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 149 | 431 | 149 | 431 |
| | 600 | 298 | 0 | 0 | 0 | 0 | 35 | 0 | 158 | 0 | 6 | 0 | 0 | 0 |
| | 600 | 176 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 221 | 0 | 0 | 0 | 0 | 101 | 138 | 206 | 370 | 221 | 359 | 221 | 359 |
| 2 | 600 | 172 | 17 | 0 | 62 | 0 | 12 | 0 | 40 | 0 | 172 | 408 | 172 | 408 |
| | 600 | 181 | 26 | 1 | 126 | 0 | 87 | 162 | 181 | 405 | 20 | 0 | 0 | 0 |
| | 600 | 261 | 118 | 0 | 233 | 0 | 56 | 0 | 139 | 0 | 261 | 319 | 261 | 319 |
| | 600 | 181 | 2 | 0 | 0 | 0 | 46 | 61 | 164 | 218 | 8 | 0 | 0 | 0 |
| | 600 | 222 | 96 | 95 | 222 | 344 | 45 | 7 | 123 | 0 | 110 | 1 | 116 | 0 |
| | 600 | 250 | 17 | 0 | 155 | 0 | 83 | 56 | 250 | 138 | 250 | 330 | 250 | 330 |
| | 600 | 300 | 86 | 31 | 300 | 187 | 28 | 0 | 97 | 0 | 300 | 280 | 300 | 280 |
| | 600 | 148 | 29 | 23 | 131 | 113 | 16 | 0 | 33 | 0 | 148 | 432 | 148 | 432 |
| | 600 | 196 | 63 | 20 | 186 | 83 | 76 | 43 | 194 | 161 | 118 | 0 | 133 | 0 |
| | 600 | 254 | 135 | 9 | 235 | 0 | 60 | 0 | 163 | 0 | 195 | 0 | 200 | 0 |
| | 600 | 382 | 195 | 0 | 367 | 0 | 86 | 0 | 228 | 0 | 382 | 198 | 382 | 198 |
| | 600 | 197 | 69 | 51 | 197 | 326 | 114 | 149 | 197 | 395 | 197 | 383 | 197 | 383 |
| | 600 | 185 | 17 | 0 | 107 | 0 | 95 | 170 | 185 | 407 | 76 | 11 | 113 | 0 |
| | 600 | 423 | 212 | 5 | 384 | 0 | 98 | 0 | 260 | 0 | 423 | 157 | 423 | 157 |
| | 600 | 275 | 129 | 0 | 246 | 0 | 150 | 27 | 246 | 53 | 275 | 305 | 275 | 305 |
| | 600 | 169 | 12 | 8 | 93 | 0 | 109 | 239 | 169 | 426 | 169 | 411 | 169 | 411 |
| | 600 | 198 | 50 | 0 | 157 | 0 | 87 | 58 | 198 | 267 | 198 | 382 | 198 | 382 |
| | 600 | 254 | 101 | 0 | 195 | 0 | 135 | 25 | 229 | 52 | 254 | 326 | 254 | 326 |
| | 600 | 141 | 35 | 19 | 123 | 76 | 96 | 236 | 141 | 452 | 141 | 439 | 141 | 439 |
| | 600 | 206 | 51 | 31 | 188 | 222 | 18 | 0 | 0 | 0 | 105 | 2 | 143 | 0 |
| | 600 | 122 | 5 | 0 | 0 | 0 | 35 | 120 | 114 | 433 | 2 | 0 | 0 | 0 |
| | 600 | 152 | 51 | 65 | 152 | 407 | 112 | 215 | 152 | 442 | 152 | 428 | 152 | 428 |
| | 600 | 121 | 0 | 0 | 0 | 0 | 39 | 103 | 121 | 456 | 2 | 0 | 0 | 0 |
| | 600 | 166 | 28 | 0 | 125 | 0 | 79 | 112 | 166 | 413 | 166 | 414 | 166 | 414 |
| | 600 | 330 | 150 | 0 | 276 | 0 | 136 | 0 | 212 | 0 | 290 | 0 | 296 | 0 |
| | 600 | 148 | 21 | 0 | 96 | 0 | 50 | 63 | 133 | 230 | 55 | 0 | 57 | 0 |
| | 600 | 195 | 35 | 13 | 195 | 13 | 61 | 54 | 163 | 132 | 61 | 0 | 57 | 0 |
| | 600 | 230 | 1 | 0 | 0 | 0 | 69 | 70 | 218 | 294 | 230 | 350 | 230 | 350 |
| | 600 | 328 | 40 | 0 | 198 | 0 | 41 | 0 | 178 | 0 | 171 | 0 | 191 | 0 |
| | 600 | 191 | 23 | 0 | 153 | 0 | 104 | 158 | 191 | 401 | 191 | 389 | 191 | 389 |
| | 600 | 196 | 66 | 19 | 163 | 50 | 24 | 0 | 95 | 0 | 196 | 384 | 196 | 384 |
| | 600 | 285 | 134 | 0 | 263 | 0 | 57 | 0 | 116 | 0 | 251 | 28 | 264 | 23 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 142 | 13 | 0 | 52 | 0 | 56 | 147 | 128 | 431 | 70 | 8 | 96 | 0 |
| | 600 | 173 | 37 | 30 | 147 | 247 | 114 | 224 | 173 | 420 | 25 | 0 | 0 | 0 |
| | 600 | 210 | 37 | 13 | 183 | 0 | 115 | 151 | 210 | 383 | 210 | 370 | 210 | 370 |
| | 600 | 489 | 263 | 3 | 458 | 28 | 181 | 0 | 351 | 0 | 453 | 0 | 468 | 0 |
| | 600 | 264 | 50 | 7 | 216 | 0 | 175 | 169 | 264 | 331 | 65 | 1 | 99 | 0 |
| | 600 | 217 | 25 | 0 | 151 | 0 | 71 | 70 | 217 | 278 | 21 | 0 | 0 | 0 |
| | 600 | 277 | 40 | 0 | 198 | 0 | 188 | 143 | 277 | 311 | 277 | 303 | 277 | 303 |
| | 600 | 175 | 29 | 16 | 157 | 132 | 117 | 211 | 175 | 419 | 175 | 405 | 175 | 405 |
| | 600 | 190 | 22 | 0 | 114 | 0 | 132 | 165 | 190 | 404 | 190 | 390 | 190 | 390 |
| | 600 | 261 | 29 | 0 | 156 | 0 | 53 | 2 | 228 | 0 | 261 | 319 | 261 | 319 |
| | 600 | 388 | 155 | 0 | 315 | 0 | 89 | 0 | 196 | 0 | 318 | 0 | 341 | 0 |
| | 600 | 188 | 59 | 21 | 188 | 147 | 52 | 0 | 128 | 0 | 188 | 392 | 188 | 392 |
| | 600 | 457 | 153 | 0 | 332 | 0 | 167 | 0 | 322 | 0 | 457 | 123 | 457 | 123 |
| | 600 | 163 | 19 | 0 | 92 | 0 | 97 | 178 | 163 | 430 | 163 | 417 | 163 | 417 |
| | 600 | 268 | 59 | 8 | 242 | 28 | 85 | 45 | 237 | 171 | 268 | 312 | 268 | 312 |
| | 600 | 305 | 53 | 0 | 211 | 0 | 46 | 0 | 152 | 0 | 253 | 10 | 281 | 13 |
| | 600 | 173 | 27 | 42 | 173 | 339 | 59 | 96 | 173 | 401 | 32 | 0 | 51 | 0 |
| 0.03 | 600 | 181 | 55 | 3 | 159 | 0 | 21 | 0 | 46 | 0 | 181 | 399 | 181 | 399 |
| | 600 | 174 | 56 | 45 | 147 | 282 | 51 | 27 | 147 | 107 | 126 | 0 | 132 | 0 |
| | 600 | 283 | 102 | 0 | 257 | 0 | 79 | 0 | 216 | 0 | 283 | 297 | 283 | 297 |
| | 600 | 255 | 147 | 0 | 200 | 0 | 99 | 0 | 180 | 0 | 255 | 325 | 255 | 325 |
| | 600 | 249 | 57 | 0 | 227 | 0 | 119 | 71 | 249 | 268 | 249 | 331 | 249 | 331 |
| | 600 | 182 | 54 | 0 | 131 | 0 | 88 | 19 | 167 | 0 | 182 | 398 | 182 | 398 |
| | 600 | 153 | 24 | 53 | 153 | 336 | 24 | 51 | 121 | 217 | 153 | 427 | 153 | 427 |
| | 600 | 185 | 55 | 31 | 185 | 273 | 85 | 70 | 185 | 267 | 185 | 395 | 185 | 395 |
| | 600 | 243 | 72 | 26 | 243 | 156 | 136 | 119 | 243 | 350 | 67 | 0 | 0 | 0 |
| | 600 | 154 | 20 | 11 | 119 | 61 | 92 | 190 | 154 | 439 | 154 | 426 | 154 | 426 |
| | 600 | 272 | 0 | 0 | 0 | 0 | 113 | 95 | 272 | 307 | 272 | 308 | 272 | 308 |
| | 600 | 144 | 14 | 0 | 93 | 0 | 83 | 185 | 144 | 450 | 144 | 436 | 144 | 436 |
| | 600 | 121 | 10 | 0 | 79 | 0 | 15 | 7 | 41 | 0 | 121 | 459 | 121 | 459 |
| | 600 | 203 | 56 | 0 | 148 | 0 | 52 | 0 | 140 | 0 | 199 | 31 | 203 | 40 |
| | 600 | 337 | 59 | 0 | 239 | 0 | 28 | 0 | 103 | 0 | 192 | 0 | 206 | 0 |
| | 600 | 196 | 28 | 0 | 112 | 0 | 84 | 129 | 196 | 397 | 28 | 0 | 0 | 0 |
| | 600 | 193 | 57 | 0 | 170 | 0 | 44 | 0 | 84 | 0 | 107 | 0 | 125 | 0 |
| | 600 | 265 | 181 | 42 | 265 | 251 | 97 | 0 | 163 | 0 | 216 | 0 | 215 | 0 |
| | 600 | 243 | 85 | 26 | 243 | 172 | 82 | 25 | 229 | 26 | 243 | 337 | 243 | 337 |
| | 600 | 217 | 27 | 0 | 145 | 0 | 44 | 0 | 131 | 0 | 217 | 363 | 217 | 363 |
| | 600 | 278 | 97 | 33 | 272 | 249 | 56 | 0 | 185 | 0 | 210 | 2 | 218 | 0 |
| | 600 | 337 | 48 | 0 | 244 | 0 | 114 | 23 | 322 | 98 | 184 | 0 | 263 | 0 |
| | 600 | 183 | 55 | 13 | 175 | 117 | 59 | 31 | 148 | 10 | 141 | 4 | 148 | 4 |
| | 600 | 199 | 69 | 25 | 198 | 96 | 17 | 0 | 58 | 0 | 199 | 381 | 199 | 381 |
| | 600 | 175 | 29 | 6 | 172 | 8 | 139 | 273 | 175 | 418 | 18 | 0 | 0 | 0 |
| | 600 | 290 | 73 | 29 | 290 | 209 | 51 | 0 | 159 | 0 | 290 | 290 | 290 | 290 |
| | 600 | 132 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 210 | 85 | 57 | 210 | 358 | 18 | 0 | 96 | 0 | 135 | 0 | 145 | 0 |
| | 600 | 293 | 220 | 61 | 293 | 285 | 105 | 0 | 195 | 0 | 293 | 287 | 293 | 287 |
| | 600 | 466 | 265 | 0 | 410 | 0 | 193 | 0 | 305 | 0 | 466 | 114 | 466 | 114 |
| | 600 | 250 | 64 | 0 | 228 | 0 | 44 | 0 | 200 | 0 | 189 | 0 | 216 | 0 |
| | 600 | 201 | 75 | 6 | 171 | 0 | 51 | 0 | 117 | 0 | 201 | 379 | 201 | 379 |
| | 600 | 218 | 40 | 0 | 158 | 0 | 43 | 0 | 145 | 0 | 218 | 362 | 218 | 362 |
| | 600 | 336 | 75 | 16 | 302 | 163 | 168 | 92 | 336 | 259 | 149 | 0 | 199 | 0 |
| | 600 | 364 | 205 | 0 | 327 | 0 | 138 | 0 | 212 | 0 | 364 | 216 | 364 | 216 |
| | 600 | 325 | 172 | 0 | 263 | 0 | 126 | 0 | 229 | 0 | 325 | 255 | 325 | 255 |
| | 600 | 224 | 77 | 33 | 208 | 184 | 24 | 0 | 109 | 0 | 102 | 0 | 142 | 0 |
| | 600 | 212 | 26 | 0 | 141 | 0 | 42 | 0 | 147 | 0 | 93 | 0 | 131 | 0 |
| | 600 | 148 | 41 | 15 | 123 | 51 | 52 | 28 | 145 | 59 | 55 | 0 | 89 | 0 |
| | 600 | 228 | 101 | 89 | 228 | 349 | 8 | 0 | 0 | 0 | 228 | 352 | 228 | 352 |
| | 600 | 288 | 65 | 8 | 268 | 30 | 164 | 108 | 288 | 300 | 288 | 292 | 288 | 292 |
| | 600 | 226 | 29 | 0 | 159 | 0 | 100 | 66 | 226 | 302 | 182 | 2 | 185 | 2 |
| | 600 | 347 | 199 | 0 | 297 | 0 | 127 | 0 | 214 | 0 | 303 | 0 | 309 | 0 |
| | 600 | 199 | 77 | 49 | 183 | 254 | 66 | 33 | 181 | 123 | 199 | 381 | 199 | 381 |
| | 600 | 154 | 28 | 0 | 128 | 0 | 43 | 31 | 144 | 59 | 102 | 0 | 109 | 0 |
| | 600 | 280 | 124 | 37 | 280 | 221 | 105 | 22 | 276 | 3 | 280 | 300 | 280 | 300 |
| | 600 | 263 | 36 | 0 | 178 | 0 | 70 | 27 | 209 | 31 | 263 | 317 | 263 | 317 |
| | 600 | 108 | 39 | 104 | 108 | 467 | 37 | 97 | 102 | 443 | 108 | 472 | 108 | 472 |
| | 600 | 210 | 42 | 0 | 157 | 0 | 51 | 0 | 170 | 0 | 2 | 0 | 0 | 0 |
| | 600 | 210 | 46 | 3 | 192 | 20 | 41 | 0 | 157 | 0 | 2 | 0 | 0 | 0 |
| 0.02 | 600 | 249 | 42 | 0 | 170 | 0 | 27 | 0 | 100 | 0 | 184 | 0 | 199 | 0 |
| | 600 | 180 | 39 | 0 | 125 | 0 | 45 | 0 | 125 | 0 | 110 | 0 | 112 | 0 |
| | 600 | 253 | 59 | 0 | 196 | 0 | 108 | 60 | 250 | 263 | 198 | 0 | 205 | 0 |
| | 600 | 193 | 55 | 0 | 161 | 0 | 44 | 0 | 156 | 0 | 193 | 387 | 193 | 387 |
| | 600 | 447 | 334 | 0 | 419 | 0 | 262 | 0 | 348 | 0 | 350 | 0 | 350 | 0 |
| | 600 | 251 | 85 | 8 | 243 | 36 | 55 | 0 | 172 | 0 | 251 | 329 | 251 | 329 |
| | 600 | 284 | 119 | 0 | 227 | 0 | 124 | 0 | 239 | 0 | 204 | 0 | 205 | 0 |
| | 600 | 277 | 211 | 0 | 257 | 0 | 210 | 0 | 257 | 0 | 237 | 0 | 237 | 0 |
| | 600 | 173 | 28 | 0 | 137 | 0 | 69 | 88 | 173 | 377 | 173 | 407 | 173 | 407 |
| | 600 | 172 | 68 | 30 | 172 | 201 | 80 | 63 | 172 | 235 | 108 | 0 | 123 | 0 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 135 | 37 | 36 | 125 | 313 | 35 | 20 | 107 | 0 | 91 | 4 | 97 | 0 |
| | 600 | 157 | 0 | 0 | 0 | 0 | 86 | 190 | 157 | 437 | 0 | 0 | 0 | 0 |
| | 600 | 220 | 77 | 0 | 156 | 0 | 59 | 0 | 125 | 0 | 160 | 0 | 170 | 0 |
| | 600 | 244 | 92 | 0 | 173 | 0 | 74 | 0 | 164 | 0 | 244 | 336 | 244 | 336 |
| | 600 | 186 | 85 | 58 | 186 | 340 | 31 | 0 | 129 | 0 | 111 | 0 | 114 | 0 |
| | 600 | 243 | 145 | 3 | 225 | 0 | 148 | 18 | 234 | 32 | 243 | 337 | 243 | 337 |
| | 600 | 303 | 218 | 0 | 266 | 0 | 229 | 21 | 274 | 60 | 303 | 277 | 303 | 277 |
| | 600 | 342 | 119 | 0 | 211 | 0 | 119 | 0 | 211 | 0 | 201 | 0 | 204 | 0 |
| | 600 | 288 | 131 | 0 | 233 | 0 | 99 | 0 | 194 | 0 | 288 | 292 | 288 | 292 |
| | 600 | 143 | 47 | 14 | 139 | 57 | 47 | 18 | 132 | 31 | 143 | 437 | 143 | 437 |
| | 600 | 122 | 21 | 0 | 100 | 0 | 62 | 148 | 122 | 463 | 0 | 0 | 0 | 0 |
| | 600 | 333 | 154 | 0 | 285 | 0 | 84 | 0 | 204 | 0 | 206 | 0 | 213 | 0 |
| | 600 | 238 | 58 | 0 | 184 | 0 | 44 | 0 | 169 | 0 | 28 | 0 | 0 | 0 |
| | 600 | 346 | 104 | 0 | 313 | 0 | 75 | 0 | 274 | 0 | 346 | 234 | 346 | 234 |
| | 600 | 157 | 60 | 119 | 157 | 419 | 111 | 229 | 157 | 434 | 4 | 0 | 0 | 0 |
| | 600 | 297 | 201 | 0 | 248 | 0 | 179 | 0 | 227 | 0 | 222 | 0 | 222 | 0 |
| | 600 | 213 | 53 | 0 | 175 | 0 | 70 | 15 | 185 | 0 | 213 | 367 | 213 | 367 |
| | 600 | 141 | 33 | 0 | 114 | 0 | 70 | 76 | 139 | 355 | 141 | 439 | 141 | 439 |
| | 600 | 201 | 43 | 11 | 194 | 61 | 27 | 0 | 150 | 0 | 201 | 379 | 201 | 379 |
| | 600 | 153 | 23 | 0 | 110 | 0 | 45 | 12 | 134 | 0 | 48 | 0 | 97 | 0 |
| | 600 | 228 | 27 | 0 | 168 | 0 | 20 | 0 | 41 | 0 | 168 | 0 | 178 | 0 |
| | 600 | 286 | 202 | 0 | 261 | 0 | 168 | 0 | 214 | 0 | 286 | 294 | 286 | 294 |
| | 600 | 179 | 42 | 0 | 148 | 0 | 71 | 31 | 171 | 121 | 111 | 0 | 132 | 0 |
| | 600 | 211 | 45 | 0 | 147 | 0 | 36 | 0 | 104 | 0 | 55 | 0 | 59 | 0 |
| | 600 | 138 | 0 | 0 | 0 | 0 | 35 | 47 | 138 | 146 | 138 | 442 | 138 | 442 |
| | 600 | 225 | 80 | 1 | 197 | 0 | 44 | 0 | 137 | 0 | 146 | 0 | 146 | 0 |
| | 600 | 327 | 210 | 0 | 287 | 0 | 170 | 0 | 235 | 0 | 327 | 253 | 327 | 253 |
| | 600 | 326 | 254 | 29 | 326 | 218 | 229 | 0 | 292 | 0 | 291 | 0 | 291 | 0 |
| | 600 | 207 | 66 | 0 | 123 | 0 | 86 | 0 | 133 | 0 | 207 | 373 | 207 | 373 |
| | 600 | 302 | 115 | 9 | 299 | 35 | 111 | 4 | 288 | 0 | 302 | 278 | 302 | 278 |
| | 600 | 204 | 134 | 45 | 204 | 270 | 99 | 0 | 144 | 0 | 150 | 0 | 152 | 0 |
| | 600 | 177 | 107 | 25 | 177 | 128 | 134 | 116 | 177 | 396 | 177 | 403 | 177 | 403 |
| | 600 | 175 | 75 | 44 | 158 | 344 | 91 | 84 | 175 | 335 | 175 | 405 | 175 | 405 |
| | 600 | 113 | 9 | 0 | 35 | 0 | 31 | 41 | 108 | 145 | 113 | 467 | 113 | 467 |
| | 600 | 215 | 53 | 0 | 163 | 0 | 46 | 0 | 153 | 0 | 128 | 0 | 150 | 0 |
| | 600 | 228 | 77 | 0 | 161 | 0 | 86 | 0 | 165 | 0 | 228 | 352 | 228 | 352 |
| | 600 | 186 | 54 | 0 | 162 | 0 | 85 | 45 | 167 | 169 | 77 | 0 | 95 | 0 |
| | 600 | 471 | 336 | 0 | 422 | 0 | 268 | 0 | 364 | 0 | 471 | 109 | 471 | 109 |
| | 600 | 303 | 123 | 0 | 244 | 0 | 108 | 0 | 196 | 0 | 303 | 277 | 303 | 277 |
| | 600 | 117 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 6 | 0 | 0 | 0 |
| 0.01 | 600 | 185 | 130 | 0 | 151 | 0 | 155 | 47 | 185 | 164 | 144 | 0 | 144 | 0 |
| | 600 | 183 | 64 | 0 | 124 | 0 | 88 | 0 | 149 | 0 | 111 | 0 | 111 | 0 |
| | 600 | 175 | 18 | 0 | 121 | 0 | 61 | 5 | 151 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 186 | 86 | 0 | 151 | 0 | 156 | 100 | 186 | 379 | 121 | 0 | 121 | 0 |
| | 600 | 116 | 10 | 0 | 64 | 0 | 68 | 105 | 116 | 467 | 0 | 0 | 0 | 0 |
| | 600 | 189 | 64 | 0 | 137 | 0 | 102 | 0 | 166 | 0 | 189 | 391 | 189 | 391 |
| | 600 | 205 | 80 | 0 | 162 | 0 | 113 | 0 | 173 | 0 | 133 | 0 | 133 | 0 |
| | 600 | 177 | 49 | 0 | 147 | 0 | 84 | 64 | 165 | 238 | 177 | 403 | 177 | 403 |
| | 600 | 154 | 0 | 0 | 0 | 0 | 31 | 0 | 124 | 0 | 154 | 426 | 154 | 426 |
| | 600 | 360 | 284 | 0 | 315 | 0 | 268 | 0 | 304 | 0 | 303 | 0 | 303 | 0 |
| | 600 | 168 | 94 | 0 | 127 | 0 | 55 | 0 | 101 | 0 | 168 | 412 | 168 | 412 |
| | 600 | 179 | 47 | 0 | 126 | 0 | 102 | 0 | 150 | 0 | 179 | 401 | 179 | 401 |
| | 600 | 212 | 63 | 0 | 148 | 0 | 78 | 0 | 155 | 0 | 139 | 0 | 139 | 0 |
| | 600 | 218 | 146 | 0 | 177 | 0 | 154 | 0 | 181 | 0 | 218 | 362 | 218 | 362 |
| | 600 | 299 | 188 | 0 | 235 | 0 | 174 | 0 | 216 | 0 | 219 | 0 | 224 | 0 |
| | 600 | 149 | 66 | 0 | 108 | 0 | 86 | 0 | 130 | 0 | 149 | 431 | 149 | 431 |
| | 600 | 181 | 59 | 0 | 135 | 0 | 58 | 0 | 135 | 0 | 108 | 0 | 114 | 0 |
| | 600 | 152 | 74 | 0 | 134 | 0 | 47 | 0 | 110 | 0 | 152 | 428 | 152 | 428 |
| | 600 | 214 | 87 | 0 | 144 | 0 | 117 | 0 | 168 | 0 | 214 | 366 | 214 | 366 |
| | 600 | 348 | 189 | 0 | 300 | 0 | 96 | 0 | 243 | 0 | 348 | 232 | 348 | 232 |
| | 600 | 174 | 45 | 0 | 125 | 0 | 60 | 0 | 127 | 0 | 65 | 0 | 68 | 0 |
| | 600 | 265 | 215 | 0 | 235 | 0 | 200 | 0 | 226 | 0 | 232 | 0 | 232 | 0 |
| | 600 | 180 | 133 | 0 | 159 | 0 | 133 | 0 | 159 | 0 | 140 | 0 | 140 | 0 |
| | 600 | 306 | 254 | 0 | 281 | 0 | 227 | 0 | 257 | 0 | 306 | 274 | 306 | 274 |
| | 600 | 185 | 48 | 0 | 118 | 0 | 63 | 0 | 121 | 0 | 185 | 395 | 185 | 395 |
| | 600 | 266 | 219 | 0 | 243 | 0 | 203 | 0 | 226 | 0 | 266 | 314 | 266 | 314 |
| | 600 | 122 | 27 | 0 | 100 | 0 | 51 | 59 | 122 | 244 | 122 | 458 | 122 | 458 |
| | 600 | 202 | 48 | 0 | 135 | 0 | 64 | 0 | 141 | 0 | 130 | 0 | 130 | 0 |
| | 600 | 142 | 58 | 21 | 140 | 92 | 44 | 0 | 128 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 152 | 73 | 0 | 123 | 0 | 58 | 0 | 115 | 0 | 152 | 428 | 152 | 428 |
| | 600 | 148 | 72 | 0 | 109 | 0 | 122 | 114 | 142 | 367 | 95 | 0 | 95 | 0 |
| | 600 | 196 | 15 | 0 | 102 | 0 | 73 | 14 | 184 | 6 | 196 | 384 | 196 | 384 |
| | 600 | 186 | 34 | 0 | 117 | 0 | 65 | 0 | 141 | 0 | 186 | 394 | 186 | 394 |
| | 600 | 258 | 93 | 0 | 198 | 0 | 145 | 0 | 224 | 0 | 258 | 322 | 258 | 322 |
| | 600 | 278 | 175 | 0 | 218 | 0 | 161 | 0 | 196 | 0 | 278 | 302 | 278 | 302 |
| | 600 | 203 | 116 | 0 | 168 | 0 | 99 | 0 | 168 | 0 | 142 | 0 | 142 | 0 |
| | 600 | 486 | 398 | 0 | 452 | 0 | 356 | 0 | 411 | 0 | 415 | 0 | 416 | 0 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 369 | 284 | 0 | 319 | 0 | 277 | 0 | 313 | 0 | 300 | 0 | 300 | 0 |
| | 600 | 211 | 121 | 0 | 156 | 0 | 163 | 0 | 197 | 0 | 149 | 0 | 149 | 0 |
| | 600 | 203 | 131 | 0 | 165 | 0 | 117 | 0 | 139 | 0 | 203 | 377 | 203 | 377 |
| | 600 | 238 | 118 | 0 | 185 | 0 | 128 | 0 | 198 | 0 | 166 | 0 | 166 | 0 |
| | 600 | 270 | 216 | 0 | 230 | 0 | 208 | 0 | 229 | 0 | 270 | 310 | 270 | 310 |
| | 600 | 160 | 50 | 0 | 105 | 0 | 89 | 0 | 132 | 0 | 160 | 420 | 160 | 420 |
| | 600 | 122 | 7 | 0 | 65 | 0 | 50 | 0 | 112 | 0 | 122 | 458 | 122 | 458 |
| | 600 | 183 | 120 | 0 | 147 | 0 | 128 | 0 | 160 | 0 | 183 | 397 | 183 | 397 |
| | 600 | 367 | 303 | 0 | 325 | 0 | 269 | 0 | 301 | 0 | 307 | 0 | 307 | 0 |
| | 600 | 228 | 53 | 0 | 159 | 0 | 80 | 0 | 177 | 0 | 228 | 352 | 228 | 352 |
| | 600 | 201 | 131 | 0 | 170 | 0 | 163 | 60 | 191 | 283 | 144 | 0 | 146 | 0 |
| | 600 | 261 | 130 | 0 | 216 | 0 | 135 | 0 | 216 | 0 | 261 | 319 | 261 | 319 |
| | 600 | 339 | 222 | 0 | 269 | 0 | 206 | 0 | 253 | 0 | 227 | 0 | 227 | 0 |
| 0 | 600 | 216 | 152 | 0 | 152 | 0 | 152 | 0 | 152 | 0 | 156 | 0 | 156 | 0 |
| | 600 | 191 | 125 | 0 | 125 | 0 | 159 | 0 | 159 | 0 | 127 | 0 | 127 | 0 |
| | 600 | 297 | 195 | 0 | 195 | 0 | 247 | 0 | 247 | 0 | 198 | 0 | 198 | 0 |
| | 600 | 138 | 86 | 0 | 86 | 0 | 106 | 0 | 106 | 0 | 83 | 0 | 83 | 0 |
| | 600 | 180 | 128 | 0 | 128 | 0 | 128 | 0 | 128 | 0 | 129 | 0 | 129 | 0 |
| | 600 | 138 | 86 | 0 | 86 | 0 | 106 | 0 | 106 | 0 | 83 | 0 | 83 | 0 |
| | 600 | 214 | 138 | 0 | 138 | 0 | 172 | 0 | 172 | 0 | 140 | 0 | 140 | 0 |
| | 600 | 170 | 106 | 0 | 106 | 0 | 130 | 0 | 130 | 0 | 105 | 0 | 105 | 0 |
| | 600 | 194 | 124 | 0 | 124 | 0 | 156 | 0 | 156 | 0 | 126 | 0 | 126 | 0 |
| | 600 | 158 | 102 | 0 | 102 | 0 | 128 | 0 | 128 | 0 | 102 | 0 | 102 | 0 |
| | 600 | 319 | 263 | 0 | 263 | 0 | 255 | 0 | 255 | 0 | 266 | 0 | 266 | 0 |
| | 600 | 217 | 143 | 0 | 143 | 0 | 181 | 0 | 181 | 0 | 144 | 0 | 144 | 0 |
| | 600 | 208 | 130 | 0 | 130 | 0 | 160 | 0 | 160 | 0 | 131 | 0 | 131 | 0 |
| | 600 | 414 | 342 | 0 | 342 | 0 | 330 | 0 | 330 | 0 | 345 | 0 | 345 | 0 |
| | 600 | 178 | 126 | 0 | 126 | 0 | 126 | 0 | 126 | 0 | 129 | 0 | 129 | 0 |
| | 600 | 240 | 150 | 0 | 150 | 0 | 184 | 0 | 184 | 0 | 151 | 0 | 151 | 0 |
| | 600 | 230 | 150 | 0 | 150 | 0 | 192 | 0 | 192 | 0 | 153 | 0 | 153 | 0 |
| | 600 | 359 | 297 | 0 | 297 | 0 | 287 | 0 | 287 | 0 | 300 | 0 | 300 | 0 |
| | 600 | 219 | 143 | 0 | 143 | 0 | 181 | 0 | 181 | 0 | 145 | 0 | 145 | 0 |
| | 600 | 218 | 140 | 0 | 140 | 0 | 176 | 0 | 176 | 0 | 143 | 0 | 143 | 0 |
| | 600 | 148 | 92 | 0 | 92 | 0 | 112 | 0 | 112 | 0 | 91 | 0 | 91 | 0 |
| | 600 | 453 | 375 | 0 | 375 | 0 | 361 | 0 | 361 | 0 | 378 | 0 | 378 | 0 |
| | 600 | 234 | 166 | 0 | 166 | 0 | 166 | 0 | 166 | 0 | 169 | 0 | 169 | 0 |
| | 600 | 274 | 194 | 0 | 194 | 0 | 194 | 0 | 194 | 0 | 197 | 0 | 197 | 0 |
| | 600 | 312 | 258 | 0 | 258 | 0 | 248 | 0 | 248 | 0 | 260 | 0 | 260 | 0 |
| | 600 | 159 | 99 | 0 | 99 | 0 | 121 | 0 | 121 | 0 | 98 | 0 | 98 | 0 |
| | 600 | 186 | 116 | 0 | 116 | 0 | 142 | 0 | 142 | 0 | 117 | 0 | 117 | 0 |
| | 600 | 230 | 150 | 0 | 150 | 0 | 192 | 0 | 192 | 0 | 153 | 0 | 153 | 0 |
| | 600 | 209 | 131 | 0 | 131 | 0 | 161 | 0 | 161 | 0 | 132 | 0 | 132 | 0 |
| | 600 | 287 | 203 | 0 | 203 | 0 | 203 | 0 | 203 | 0 | 207 | 0 | 207 | 0 |
| | 600 | 300 | 212 | 0 | 212 | 0 | 212 | 0 | 212 | 0 | 216 | 0 | 216 | 0 |
| | 600 | 289 | 205 | 0 | 205 | 0 | 205 | 0 | 205 | 0 | 208 | 0 | 208 | 0 |
| | 600 | 205 | 131 | 0 | 131 | 0 | 165 | 0 | 165 | 0 | 134 | 0 | 134 | 0 |
| | 600 | 195 | 121 | 0 | 121 | 0 | 149 | 0 | 149 | 0 | 122 | 0 | 122 | 0 |
| | 600 | 148 | 96 | 0 | 96 | 0 | 120 | 0 | 120 | 0 | 96 | 0 | 96 | 0 |
| | 600 | 256 | 212 | 0 | 212 | 0 | 204 | 0 | 204 | 0 | 213 | 0 | 213 | 0 |
| | 600 | 175 | 109 | 0 | 109 | 0 | 133 | 0 | 133 | 0 | 108 | 0 | 108 | 0 |
| | 600 | 181 | 119 | 0 | 119 | 0 | 151 | 0 | 151 | 0 | 119 | 0 | 119 | 0 |
| | 600 | 146 | 94 | 0 | 94 | 0 | 118 | 0 | 118 | 0 | 94 | 0 | 94 | 0 |
| | 600 | 241 | 171 | 0 | 171 | 0 | 171 | 0 | 171 | 0 | 174 | 0 | 174 | 0 |
| | 600 | 175 | 109 | 0 | 109 | 0 | 133 | 0 | 133 | 0 | 108 | 0 | 108 | 0 |
| | 600 | 344 | 244 | 0 | 244 | 0 | 244 | 0 | 244 | 0 | 248 | 0 | 248 | 0 |
| | 600 | 238 | 156 | 0 | 156 | 0 | 198 | 0 | 198 | 0 | 158 | 0 | 158 | 0 |
| | 600 | 157 | 97 | 0 | 97 | 0 | 119 | 0 | 119 | 0 | 96 | 0 | 96 | 0 |
| | 600 | 237 | 155 | 0 | 155 | 0 | 197 | 0 | 197 | 0 | 157 | 0 | 157 | 0 |
| | 600 | 157 | 97 | 0 | 97 | 0 | 119 | 0 | 119 | 0 | 96 | 0 | 96 | 0 |
| | 600 | 265 | 173 | 0 | 173 | 0 | 221 | 0 | 221 | 0 | 177 | 0 | 177 | 0 |
| | 600 | 246 | 160 | 0 | 160 | 0 | 204 | 0 | 204 | 0 | 163 | 0 | 163 | 0 |
| | 600 | 343 | 283 | 0 | 283 | 0 | 273 | 0 | 273 | 0 | 286 | 0 | 286 | 0 |
| | 600 | 152 | 94 | 0 | 94 | 0 | 116 | 0 | 116 | 0 | 93 | 0 | 93 | 0 |

# Table B.14 Noise Effects Analysis

| | Duration | | | | Threshold | | | | FAPI | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tot D | | Tot FA | FA D | FA FA | Tot D | Tot FA | FA D | FA FA | Tot D | Tot FA | FA D | FA FA |
| | 46.66% | 26.10% | 99.98% | 91.00% | 45.11% | 29.29% | 89.39% | 65.46% | 89.42% | 62.56% | 94.32% | 63.65% |
| | 35.20% | 13.89% | 96.97% | 67.10% | 38.30% | 24.86% | 80.99% | 57.84% | 87.48% | 69.89% | 88.63% | 70.46% |
| | 19.30% | 3.36% | 69.36% | 2.45% | 27.38% | 16.77% | 64.07% | 41.76% | 85.80% | 72.51% | 87.74% | 72.09% |
| | 14.78% | 0.02% | 46.00% | 0.24% | 24.88% | 15.02% | 63.15% | 39.92% | 83.68% | 61.00% | 84.42% | 60.24% |
| | 7.58% | 0.00% | 27.64% | 0.00% | 22.93% | 12.84% | 60.22% | 34.99% | 77.58% | 63.91% | 78.31% | 63.85% |
| | 2.28% | 0.00% | 7.58% | 0.00% | 16.47% | 9.82% | 47.16% | 29.29% | 69.35% | 56.59% | 70.79% | 56.76% |
| | 1.39% | 0.00% | 5.84% | 0.00% | 10.09% | 5.24% | 31.39% | 14.99% | 68.70% | 55.25% | 70.64% | 54.92% |
| Vary Noise, Keep N @ 2 | | | | | | | | | | | | |
| | 19.30% | 3.36% | 66.94% | 23.50% | 27.02% | 15.44% | 63.50% | 40.71% | 86.24% | 73.29% | 87.73% | 72.65% |
| | 26.96% | 2.87% | 76.18% | 14.85% | 36.70% | 21.59% | 77.13% | 55.29% | 78.65% | 51.07% | 80.44% | 50.93% |
| | 31.58% | 4.46% | 83.40% | 24.96% | 31.78% | 9.89% | 72.91% | 26.82% | 78.96% | 50.76% | 80.97% | 50.80% |
| | 41.17% | 2.34% | 81.32% | 13.13% | 40.19% | 7.19% | 76.98% | 21.69% | 78.52% | 42.72% | 79.88% | 42.70% |
| | 52.11% | 0.11% | 77.98% | 0.48% | 56.89% | 2.98% | 82.31% | 11.28% | 83.23% | 51.22% | 83.38% | 51.22% |
| | 69.48% | 0.00% | 69.48% | 0.00% | 77.87% | 0.00% | 77.87% | 0.00% | 70.14% | 0.00% | 70.14% | 0.00% |

# Table B.15 Jitter Effect Results in Detection Testing

| | S Info | S Rate | Pixel X | Pixel Y | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 60 | 30 | 30 | | | | | | | | | | |
| | | | Threshold (values - # frames) | | | | | Duration (values - # frames) | | | | FAPI (values - # frames) | | |
| | | | | | FAR | FAR | | Du | FAR | FAR | FPI | FPI | FAR | FAR |
| | # F | Act D | Th D | Th FA | D | FA | Du D | FA | D | FA | D | FA | D | FA |
| 0.25 | 600 | 269 | 221 | 71 | 269 | 305 | 239 | 103 | 269 | 315 | 269 | 311 | 269 | 311 |
| | 600 | 182 | 70 | 69 | 182 | 367 | 172 | 184 | 182 | 412 | 85 | 0 | 143 | 0 |
| | 600 | 223 | 191 | 86 | 223 | 348 | 212 | 173 | 223 | 369 | 195 | 1 | 195 | 12 |
| | 600 | 193 | 165 | 85 | 193 | 386 | 169 | 99 | 193 | 379 | 173 | 16 | 173 | 30 |
| | 600 | 245 | 181 | 69 | 219 | 306 | 205 | 108 | 245 | 308 | 245 | 335 | 245 | 335 |
| | 600 | 341 | 312 | 57 | 341 | 233 | 311 | 54 | 340 | 229 | 341 | 239 | 341 | 239 |
| | 600 | 225 | 115 | 46 | 225 | 238 | 202 | 134 | 225 | 366 | 225 | 355 | 225 | 355 |
| | 600 | 281 | 227 | 68 | 281 | 287 | 245 | 87 | 281 | 272 | 243 | 8 | 252 | 3 |
| | 600 | 168 | 122 | 65 | 168 | 375 | 139 | 133 | 168 | 388 | 168 | 412 | 168 | 412 |
| | 600 | 358 | 325 | 51 | 358 | 216 | 335 | 64 | 358 | 190 | 358 | 222 | 358 | 222 |
| | 600 | 191 | 161 | 121 | 191 | 378 | 178 | 175 | 191 | 401 | 191 | 389 | 191 | 389 |
| | 600 | 242 | 145 | 34 | 242 | 217 | 227 | 149 | 242 | 353 | 242 | 338 | 242 | 338 |
| | 600 | 224 | 169 | 42 | 224 | 291 | 216 | 173 | 224 | 370 | 224 | 356 | 224 | 356 |
| | 600 | 254 | 220 | 56 | 254 | 307 | 217 | 39 | 241 | 152 | 254 | 326 | 254 | 326 |
| | 600 | 202 | 169 | 104 | 202 | 372 | 193 | 169 | 202 | 385 | 173 | 0 | 177 | 0 |
| | 600 | 254 | 200 | 63 | 254 | 325 | 222 | 106 | 248 | 326 | 254 | 326 | 254 | 326 |
| | 600 | 203 | 163 | 73 | 203 | 371 | 184 | 145 | 203 | 377 | 158 | 0 | 161 | 0 |
| | 600 | 374 | 350 | 66 | 374 | 205 | 324 | 34 | 365 | 133 | 357 | 70 | 357 | 89 |
| | 600 | 274 | 251 | 61 | 274 | 296 | 250 | 58 | 274 | 229 | 256 | 11 | 256 | 15 |
| | 600 | 189 | 143 | 54 | 189 | 369 | 162 | 113 | 189 | 361 | 155 | 0 | 161 | 0 |
| | 600 | 121 | 74 | 146 | 121 | 457 | 105 | 170 | 121 | 463 | 53 | 0 | 0 | 0 |
| | 600 | 235 | 155 | 50 | 224 | 292 | 202 | 125 | 235 | 359 | 235 | 345 | 235 | 345 |
| | 600 | 323 | 277 | 67 | 323 | 252 | 282 | 77 | 323 | 251 | 323 | 257 | 323 | 257 |
| | 600 | 162 | 129 | 84 | 162 | 375 | 139 | 108 | 162 | 357 | 123 | 0 | 123 | 0 |
| | 600 | 206 | 160 | 42 | 206 | 228 | 180 | 108 | 206 | 350 | 178 | 0 | 178 | 0 |
| | 600 | 285 | 227 | 36 | 284 | 211 | 263 | 108 | 285 | 305 | 285 | 295 | 285 | 295 |
| | 600 | 195 | 144 | 89 | 195 | 356 | 156 | 118 | 180 | 356 | 155 | 0 | 155 | 0 |
| | 600 | 184 | 160 | 101 | 184 | 391 | 171 | 184 | 184 | 410 | 184 | 396 | 184 | 396 |
| | 600 | 180 | 150 | 118 | 180 | 392 | 147 | 105 | 180 | 381 | 180 | 400 | 180 | 400 |
| | 600 | 314 | 286 | 65 | 314 | 263 | 281 | 61 | 314 | 262 | 290 | 8 | 290 | 0 |
| | 600 | 190 | 167 | 82 | 190 | 386 | 169 | 102 | 190 | 386 | 165 | 1 | 166 | 1 |
| | 600 | 208 | 181 | 110 | 208 | 372 | 178 | 103 | 208 | 386 | 208 | 372 | 208 | 372 |
| | 600 | 209 | 167 | 82 | 209 | 360 | 198 | 186 | 209 | 384 | 173 | 0 | 173 | 0 |
| | 600 | 218 | 178 | 55 | 218 | 300 | 191 | 95 | 218 | 346 | 218 | 362 | 218 | 362 |
| | 600 | 119 | 40 | 54 | 117 | 327 | 62 | 138 | 119 | 457 | 52 | 0 | 0 | 0 |
| | 600 | 330 | 251 | 52 | 316 | 226 | 99 | 13 | 252 | 0 | 330 | 250 | 330 | 250 |
| | 600 | 123 | 43 | 88 | 123 | 426 | 115 | 188 | 123 | 471 | 123 | 457 | 123 | 457 |
| | 600 | 336 | 307 | 84 | 336 | 238 | 242 | 7 | 309 | 3 | 309 | 29 | 309 | 41 |
| | 600 | 256 | 224 | 51 | 256 | 195 | 229 | 63 | 242 | 295 | 256 | 324 | 256 | 324 |
| | 600 | 260 | 176 | 37 | 260 | 259 | 33 | 0 | 151 | 0 | 203 | 0 | 205 | 0 |
| | 600 | 183 | 152 | 109 | 183 | 385 | 164 | 147 | 183 | 404 | 183 | 397 | 183 | 397 |
| | 600 | 151 | 45 | 72 | 151 | 397 | 138 | 171 | 151 | 444 | 76 | 0 | 56 | 0 |
| | 600 | 427 | 372 | 39 | 427 | 148 | 393 | 47 | 427 | 163 | 386 | 2 | 386 | 0 |
| | 600 | 199 | 156 | 76 | 199 | 338 | 169 | 109 | 199 | 261 | 166 | 2 | 166 | 0 |
| | 600 | 155 | 136 | 132 | 155 | 420 | 138 | 152 | 155 | 433 | 155 | 425 | 155 | 425 |
| | 600 | 290 | 261 | 80 | 290 | 286 | 279 | 114 | 290 | 300 | 290 | 290 | 290 | 290 |
| | 600 | 176 | 68 | 59 | 176 | 351 | 101 | 98 | 168 | 393 | 113 | 0 | 124 | 0 |
| | 600 | 313 | 249 | 47 | 313 | 233 | 281 | 91 | 313 | 277 | 313 | 267 | 313 | 267 |
| | 600 | 325 | 264 | 59 | 325 | 252 | 92 | 5 | 277 | 0 | 325 | 255 | 325 | 255 |
| | 600 | 163 | 120 | 71 | 163 | 348 | 154 | 185 | 163 | 431 | 133 | 0 | 133 | 0 |
| | 600 | 420 | 206 | 0 | 267 | 0 | 330 | 15 | 390 | 3 | 420 | 160 | 420 | 160 |
| | 600 | 138 | 40 | 7 | 130 | 69 | 114 | 118 | 138 | 425 | 36 | 0 | 0 | 0 |
| | 600 | 281 | 232 | 21 | 257 | 105 | 230 | 18 | 257 | 105 | 281 | 299 | 281 | 299 |
| | 600 | 187 | 116 | 4 | 167 | 40 | 160 | 55 | 184 | 188 | 135 | 0 | 135 | 0 |
| | 600 | 191 | 26 | 0 | 119 | 0 | 163 | 87 | 191 | 353 | 191 | 389 | 191 | 389 |
| | 600 | 159 | 72 | 11 | 124 | 113 | 133 | 88 | 159 | 366 | 159 | 421 | 159 | 421 |
| | 600 | 128 | 22 | 16 | 128 | 83 | 53 | 115 | 128 | 377 | 128 | 452 | 128 | 452 |
| | 600 | 204 | 160 | 53 | 196 | 301 | 179 | 111 | 204 | 357 | 138 | 0 | 138 | 0 |
| | 600 | 145 | 17 | 0 | 119 | 0 | 66 | 108 | 145 | 368 | 0 | 0 | 0 | 0 |
| | 600 | 230 | 116 | 22 | 201 | 128 | 178 | 88 | 230 | 300 | 230 | 350 | 230 | 350 |
| | 600 | 281 | 124 | 9 | 257 | 33 | 217 | 58 | 281 | 209 | 281 | 299 | 281 | 299 |
| | 600 | 264 | 206 | 16 | 248 | 97 | 244 | 90 | 264 | 306 | 264 | 316 | 264 | 316 |
| | 600 | 193 | 19 | 0 | 153 | 0 | 140 | 122 | 193 | 397 | 193 | 387 | 193 | 387 |
| | 600 | 200 | 65 | 7 | 181 | 4 | 179 | 126 | 200 | 352 | 117 | 0 | 119 | 0 |
| | 600 | 252 | 205 | 21 | 242 | 102 | 222 | 78 | 242 | 227 | 252 | 328 | 252 | 328 |
| | 600 | 121 | 32 | 42 | 121 | 338 | 52 | 100 | 121 | 381 | 121 | 459 | 121 | 459 |
| | 600 | 159 | 68 | 14 | 149 | 46 | 125 | 83 | 159 | 246 | 95 | 0 | 95 | 0 |
| | 600 | 359 | 285 | 17 | 342 | 67 | 289 | 22 | 342 | 72 | 359 | 221 | 359 | 221 |
| | 600 | 170 | 21 | 0 | 97 | 0 | 145 | 132 | 170 | 371 | 24 | 0 | 0 | 0 |
| | 600 | 299 | 199 | 37 | 299 | 220 | 242 | 69 | 299 | 262 | 299 | 281 | 299 | 281 |
| | 600 | 279 | 66 | 7 | 245 | 0 | 253 | 103 | 279 | 297 | 279 | 301 | 279 | 301 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 600 | 194 | 49 | 5 | 147 | 46 | 163 | 107 | 194 | 392 | 102 | 0 | 102 | 0 |
| 600 | 341 | 162 | 26 | 330 | 175 | 22 | 0 | 66 | 0 | 341 | 239 | 341 | 239 |
| 600 | 170 | 110 | 51 | 170 | 322 | 132 | 79 | 170 | 330 | 170 | 410 | 170 | 410 |
| 600 | 238 | 127 | 20 | 238 | 168 | 220 | 116 | 238 | 357 | 157 | 0 | 157 | 0 |
| 600 | 187 | 126 | 38 | 180 | 249 | 150 | 78 | 187 | 261 | 113 | 0 | 115 | 0 |
| 600 | 246 | 157 | 19 | 228 | 172 | 221 | 105 | 246 | 341 | 246 | 334 | 246 | 334 |
| 600 | 156 | 44 | 2 | 143 | 14 | 128 | 98 | 156 | 424 | 156 | 424 | 156 | 424 |
| 600 | 260 | 106 | 28 | 260 | 229 | 212 | 84 | 260 | 229 | 260 | 320 | 260 | 320 |
| 600 | 141 | 33 | 0 | 111 | 0 | 123 | 119 | 141 | 415 | 141 | 439 | 141 | 439 |
| 600 | 254 | 213 | 26 | 243 | 106 | 208 | 21 | 238 | 58 | 254 | 326 | 254 | 326 |
| 600 | 151 | 108 | 43 | 151 | 325 | 122 | 129 | 151 | 420 | 151 | 429 | 151 | 429 |
| 600 | 169 | 30 | 32 | 169 | 139 | 115 | 136 | 169 | 393 | 0 | 0 | 0 | 0 |
| 600 | 216 | 158 | 19 | 206 | 95 | 173 | 55 | 212 | 162 | 216 | 364 | 216 | 364 |
| 600 | 198 | 145 | 68 | 198 | 355 | 173 | 120 | 198 | 362 | 133 | 0 | 135 | 0 |
| 600 | 396 | 307 | 17 | 359 | 92 | 351 | 54 | 396 | 196 | 308 | 0 | 316 | 0 |
| 600 | 234 | 148 | 10 | 218 | 110 | 197 | 111 | 234 | 357 | 234 | 346 | 234 | 346 |
| 600 | 150 | 49 | 6 | 129 | 0 | 130 | 144 | 150 | 442 | 150 | 430 | 150 | 430 |
| 600 | 329 | 231 | 25 | 293 | 181 | 301 | 92 | 329 | 256 | 329 | 251 | 329 | 251 |
| 600 | 370 | 263 | 11 | 328 | 78 | 282 | 26 | 330 | 118 | 280 | 0 | 280 | 0 |
| 600 | 217 | 115 | 44 | 197 | 253 | 193 | 116 | 217 | 373 | 217 | 363 | 217 | 363 |
| 600 | 156 | 40 | 5 | 132 | 23 | 125 | 89 | 153 | 305 | 156 | 424 | 156 | 424 |
| 600 | 460 | 384 | 14 | 440 | 60 | 0 | 0 | 0 | 0 | 376 | 0 | 377 | 0 |
| 600 | 239 | 168 | 53 | 239 | 296 | 209 | 123 | 239 | 355 | 163 | 0 | 166 | 0 |
| 600 | 196 | 55 | 6 | 161 | 52 | 150 | 77 | 196 | 249 | 196 | 384 | 196 | 384 |
| 600 | 150 | 67 | 36 | 150 | 178 | 124 | 123 | 150 | 414 | 66 | 0 | 0 | 0 |
| 600 | 170 | 21 | 2 | 128 | 3 | 107 | 122 | 170 | 360 | 170 | 410 | 170 | 410 |
| 600 | 202 | 76 | 7 | 183 | 93 | 177 | 122 | 202 | 370 | 202 | 378 | 202 | 378 |
| 600 | 186 | 89 | 15 | 178 | 69 | 154 | 108 | 185 | 380 | 186 | 394 | 186 | 394 |
| 600 | 367 | 280 | 20 | 322 | 152 | 303 | 32 | 342 | 94 | 367 | 213 | 367 | 213 |
| 600 | 325 | 238 | 10 | 294 | 51 | 246 | 10 | 294 | 51 | 325 | 255 | 325 | 255 |
| 600 | 307 | 232 | 5 | 284 | 0 | 241 | 7 | 284 | 0 | 243 | 0 | 243 | 0 |
| 600 | 172 | 3 | 0 | 0 | 0 | 140 | 83 | 172 | 398 | 172 | 408 | 172 | 408 |
| 600 | 197 | 16 | 0 | 106 | 0 | 143 | 53 | 184 | 210 | 197 | 383 | 197 | 383 |
| 600 | 148 | 14 | 0 | 90 | 0 | 108 | 86 | 141 | 402 | 0 | 0 | 0 | 0 |
| 600 | 451 | 280 | 0 | 359 | 0 | 351 | 13 | 410 | 41 | 320 | 0 | 321 | 0 |
| 600 | 308 | 174 | 0 | 266 | 0 | 223 | 26 | 282 | 37 | 308 | 272 | 308 | 272 |
| 600 | 108 | 6 | 0 | 35 | 0 | 63 | 156 | 108 | 412 | 108 | 472 | 108 | 472 |
| 600 | 153 | 3 | 0 | 0 | 0 | 75 | 109 | 153 | 398 | 0 | 0 | 0 | 0 |
| 600 | 299 | 154 | 0 | 235 | 0 | 209 | 37 | 246 | 130 | 299 | 281 | 299 | 281 |
| 600 | 253 | 181 | 7 | 224 | 37 | 183 | 7 | 224 | 15 | 253 | 327 | 253 | 327 |
| 600 | 147 | 33 | 0 | 108 | 0 | 110 | 71 | 147 | 363 | 147 | 433 | 147 | 433 |
| 600 | 186 | 98 | 0 | 134 | 0 | 147 | 31 | 174 | 3 | 110 | 0 | 112 | 0 |
| 600 | 241 | 126 | 1 | 176 | 0 | 186 | 50 | 225 | 171 | 131 | 0 | 131 | 0 |
| 600 | 196 | 96 | 1 | 128 | 0 | 162 | 51 | 196 | 88 | 196 | 384 | 196 | 384 |
| 600 | 303 | 171 | 3 | 229 | 0 | 231 | 39 | 285 | 156 | 178 | 0 | 178 | 0 |
| 600 | 348 | 150 | 0 | 224 | 0 | 257 | 38 | 348 | 193 | 348 | 232 | 348 | 232 |
| 600 | 387 | 261 | 4 | 341 | 24 | 320 | 36 | 369 | 147 | 387 | 193 | 387 | 193 |
| 600 | 250 | 170 | 3 | 219 | 0 | 193 | 13 | 231 | 57 | 182 | 0 | 182 | 0 |
| 600 | 261 | 12 | 0 | 76 | 0 | 204 | 47 | 256 | 187 | 261 | 319 | 261 | 319 |
| 600 | 222 | 42 | 0 | 168 | 0 | 190 | 100 | 222 | 367 | 222 | 358 | 222 | 358 |
| 600 | 181 | 26 | 2 | 129 | 0 | 140 | 114 | 181 | 397 | 181 | 399 | 181 | 399 |
| 600 | 191 | 23 | 0 | 132 | 0 | 159 | 103 | 191 | 399 | 191 | 389 | 191 | 389 |
| 600 | 469 | 276 | 0 | 347 | 0 | 358 | 10 | 406 | 57 | 469 | 111 | 469 | 111 |
| 600 | 197 | 19 | 0 | 110 | 0 | 170 | 109 | 197 | 375 | 0 | 0 | 0 | 0 |
| 600 | 187 | 16 | 0 | 55 | 0 | 132 | 48 | 178 | 211 | 187 | 393 | 187 | 393 |
| 600 | 249 | 11 | 0 | 85 | 0 | 177 | 36 | 240 | 96 | 0 | 0 | 0 | 0 |
| 600 | 131 | 0 | 0 | 0 | 0 | 63 | 128 | 131 | 370 | 0 | 0 | 0 | 0 |
| 600 | 203 | 1 | 0 | 0 | 0 | 159 | 62 | 203 | 293 | 0 | 0 | 0 | 0 |
| 600 | 187 | 0 | 0 | 0 | 0 | 139 | 57 | 181 | 232 | 187 | 393 | 187 | 393 |
| 600 | 307 | 221 | 7 | 277 | 55 | 221 | 7 | 270 | 0 | 207 | 0 | 207 | 0 |
| 600 | 285 | 19 | 0 | 130 | 0 | 40 | 0 | 200 | 0 | 285 | 295 | 285 | 295 |
| 600 | 352 | 243 | 3 | 295 | 0 | 294 | 38 | 332 | 142 | 238 | 0 | 238 | 0 |
| 600 | 449 | 277 | 0 | 381 | 0 | 336 | 9 | 410 | 50 | 449 | 131 | 449 | 131 |
| 600 | 260 | 147 | 1 | 199 | 0 | 187 | 30 | 223 | 84 | 260 | 320 | 260 | 320 |
| 600 | 341 | 137 | 0 | 193 | 0 | 247 | 40 | 321 | 153 | 341 | 239 | 341 | 239 |
| 600 | 140 | 9 | 0 | 0 | 0 | 101 | 84 | 140 | 328 | 140 | 440 | 140 | 440 |
| 600 | 181 | 13 | 0 | 92 | 0 | 135 | 72 | 181 | 325 | 181 | 399 | 181 | 399 |
| 600 | 207 | 19 | 0 | 100 | 0 | 150 | 62 | 205 | 214 | 207 | 373 | 207 | 373 |
| 600 | 288 | 192 | 5 | 245 | 33 | 223 | 19 | 263 | 43 | 288 | 292 | 288 | 292 |
| 600 | 216 | 48 | 0 | 171 | 0 | 155 | 51 | 206 | 206 | 216 | 364 | 216 | 364 |
| 600 | 330 | 231 | 4 | 298 | 38 | 277 | 23 | 326 | 80 | 239 | 0 | 239 | 0 |
| 600 | 298 | 208 | 1 | 253 | 0 | 223 | 9 | 264 | 35 | 219 | 0 | 224 | 0 |
| 600 | 159 | 38 | 0 | 119 | 0 | 114 | 60 | 146 | 267 | 159 | 421 | 159 | 421 |
| 600 | 257 | 170 | 4 | 226 | 28 | 184 | 12 | 235 | 46 | 257 | 323 | 257 | 323 |
| 600 | 179 | 87 | 0 | 134 | 0 | 135 | 46 | 171 | 124 | 179 | 401 | 179 | 401 |
| 600 | 328 | 8 | 0 | 77 | 0 | 36 | 0 | 194 | 0 | 328 | 252 | 328 | 252 |
| 600 | 143 | 23 | 0 | 89 | 0 | 120 | 119 | 143 | 393 | 143 | 437 | 143 | 437 |
| 600 | 184 | 10 | 0 | 72 | 0 | 139 | 91 | 184 | 315 | 184 | 396 | 184 | 396 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 600 | 177 | 83 | 0 | 143 | 0 | 134 | 46 | 176 | 133 | 86 | 0 | 89 | 0 |
| 600 | 198 | 8 | 0 | 85 | 0 | 93 | 29 | 175 | 67 | 0 | 0 | 0 | 0 |
| 600 | 219 | 0 | 0 | 0 | 0 | 143 | 38 | 204 | 66 | 219 | 361 | 219 | 361 |
| 600 | 333 | 105 | 0 | 170 | 0 | 214 | 8 | 260 | 0 | 126 | 0 | 126 | 0 |
| 600 | 316 | 199 | 2 | 249 | 0 | 211 | 3 | 252 | 0 | 184 | 0 | 190 | 0 |
| 600 | 183 | 0 | 0 | 0 | 0 | 55 | 58 | 171 | 199 | 183 | 397 | 183 | 397 |
| 600 | 171 | 13 | 0 | 69 | 0 | 74 | 33 | 165 | 110 | 171 | 409 | 171 | 409 |
| 600 | 146 | 26 | 0 | 96 | 0 | 112 | 81 | 142 | 246 | 146 | 434 | 146 | 434 |
| 600 | 157 | 0 | 0 | 0 | 0 | 75 | 48 | 156 | 106 | 157 | 423 | 157 | 423 |
| 600 | 222 | 4 | 0 | 0 | 0 | 176 | 67 | 222 | 248 | 222 | 358 | 222 | 358 |
| 600 | 193 | 6 | 0 | 35 | 0 | 145 | 82 | 193 | 303 | 0 | 0 | 0 | 0 |
| 600 | 272 | 160 | 0 | 198 | 0 | 179 | 5 | 178 | 0 | 272 | 308 | 272 | 308 |
| 600 | 148 | 0 | 0 | 0 | 0 | 64 | 23 | 136 | 36 | 148 | 432 | 148 | 432 |
| 600 | 152 | 0 | 0 | 0 | 0 | 52 | 69 | 152 | 324 | 0 | 0 | 0 | 0 |
| 600 | 284 | 126 | 0 | 192 | 0 | 181 | 16 | 255 | 98 | 284 | 296 | 284 | 296 |
| 600 | 181 | 49 | 0 | 76 | 0 | 136 | 57 | 169 | 244 | 181 | 399 | 181 | 399 |
| 600 | 232 | 0 | 0 | 0 | 0 | 126 | 46 | 215 | 99 | 0 | 0 | 0 | 0 |
| 600 | 187 | 0 | 0 | 0 | 0 | 61 | 79 | 187 | 310 | 0 | 0 | 0 | 0 |
| 600 | 261 | 155 | 0 | 193 | 0 | 174 | 2 | 196 | 0 | 261 | 319 | 261 | 319 |
| 600 | 173 | 0 | 0 | 0 | 0 | 86 | 38 | 173 | 79 | 173 | 407 | 173 | 407 |
| 600 | 147 | 0 | 0 | 0 | 0 | 103 | 27 | 133 | 32 | 0 | 0 | 0 | 0 |
| 600 | 217 | 0 | 0 | 0 | 0 | 162 | 86 | 194 | 308 | 217 | 363 | 217 | 363 |
| 600 | 179 | 0 | 0 | 0 | 0 | 88 | 40 | 171 | 94 | 0 | 0 | 0 | 0 |
| 600 | 187 | 0 | 0 | 0 | 0 | 120 | 32 | 187 | 49 | 0 | 0 | 0 | 0 |
| 600 | 109 | 0 | 0 | 0 | 0 | 36 | 95 | 109 | 436 | 109 | 471 | 109 | 471 |
| 600 | 215 | 0 | 0 | 0 | 0 | 105 | 37 | 159 | 124 | 0 | 0 | 0 | 0 |
| 600 | 285 | 180 | 0 | 219 | 0 | 190 | 0 | 226 | 0 | 285 | 295 | 285 | 295 |
| 600 | 261 | 0 | 0 | 0 | 0 | 148 | 51 | 243 | 159 | 261 | 319 | 261 | 319 |
| 600 | 341 | 51 | 0 | 142 | 0 | 235 | 14 | 302 | 42 | 341 | 239 | 341 | 239 |
| 600 | 117 | 0 | 0 | 0 | 0 | 39 | 87 | 117 | 370 | 0 | 0 | 0 | 0 |
| 600 | 139 | 0 | 0 | 0 | 0 | 65 | 28 | 133 | 129 | 0 | 0 | 0 | 0 |
| 600 | 331 | 104 | 0 | 172 | 0 | 236 | 20 | 319 | 48 | 331 | 249 | 331 | 249 |
| 600 | 229 | 0 | 0 | 0 | 0 | 173 | 77 | 229 | 298 | 229 | 351 | 229 | 351 |
| 600 | 369 | 245 | 7 | 299 | 48 | 292 | 22 | 338 | 51 | 369 | 211 | 369 | 211 |
| 600 | 213 | 0 | 0 | 0 | 0 | 140 | 27 | 200 | 157 | 0 | 0 | 0 | 0 |
| 600 | 219 | 0 | 0 | 0 | 0 | 109 | 36 | 206 | 86 | 219 | 361 | 219 | 361 |
| 600 | 164 | 0 | 0 | 0 | 0 | 88 | 51 | 164 | 153 | 0 | 0 | 0 | 0 |
| 600 | 200 | 0 | 0 | 0 | 0 | 116 | 25 | 178 | 62 | 0 | 0 | 0 | 0 |
| 600 | 274 | 63 | 0 | 114 | 0 | 186 | 19 | 257 | 67 | 274 | 306 | 274 | 306 |
| 600 | 115 | 0 | 0 | 0 | 0 | 45 | 89 | 101 | 382 | 115 | 465 | 115 | 465 |
| 600 | 177 | 59 | 0 | 93 | 0 | 120 | 10 | 159 | 49 | 75 | 0 | 75 | 0 |
| 600 | 174 | 0 | 0 | 0 | 0 | 108 | 16 | 170 | 8 | 0 | 0 | 0 | 0 |
| 600 | 113 | 0 | 0 | 0 | 0 | 29 | 42 | 104 | 0 | 113 | 467 | 113 | 467 |
| 600 | 185 | 83 | 0 | 117 | 0 | 142 | 34 | 179 | 93 | 185 | 395 | 185 | 395 |
| 600 | 445 | 199 | 0 | 252 | 0 | 293 | 7 | 363 | 0 | 260 | 0 | 260 | 0 |
| 600 | 206 | 40 | 0 | 72 | 0 | 140 | 21 | 182 | 64 | 206 | 374 | 206 | 374 |
| 600 | 187 | 0 | 0 | 0 | 0 | 145 | 75 | 187 | 277 | 0 | 0 | 0 | 0 |
| 600 | 259 | 96 | 0 | 147 | 0 | 163 | 10 | 221 | 73 | 259 | 321 | 259 | 321 |
| 600 | 162 | 8 | 0 | 53 | 0 | 78 | 43 | 143 | 170 | 162 | 418 | 162 | 418 |
| 600 | 112 | 0 | 0 | 0 | 0 | 25 | 54 | 110 | 216 | 112 | 468 | 112 | 468 |
| 600 | 170 | 0 | 0 | 0 | 0 | 58 | 65 | 170 | 207 | 0 | 0 | 0 | 0 |
| 600 | 228 | 0 | 0 | 0 | 0 | 76 | 11 | 196 | 0 | 228 | 352 | 228 | 352 |
| 600 | 183 | 69 | 0 | 108 | 0 | 105 | 4 | 152 | 0 | 8 | 0 | 0 | 0 |
| 600 | 163 | 0 | 0 | 0 | 0 | 79 | 5 | 152 | 0 | 0 | 0 | 0 | 0 |
| 600 | 167 | 0 | 0 | 0 | 0 | 53 | 49 | 167 | 153 | 167 | 413 | 167 | 413 |
| 600 | 213 | 20 | 0 | 79 | 0 | 129 | 3 | 183 | 0 | 55 | 0 | 55 | 0 |
| 600 | 250 | 0 | 0 | 0 | 0 | 179 | 29 | 230 | 74 | 0 | 0 | 0 | 0 |
| 600 | 193 | 0 | 0 | 0 | 0 | 135 | 60 | 193 | 192 | 0 | 0 | 0 | 0 |
| 600 | 297 | 162 | 0 | 204 | 0 | 182 | 0 | 214 | 0 | 297 | 283 | 297 | 283 |
| 600 | 312 | 168 | 0 | 208 | 0 | 186 | 0 | 214 | 0 | 312 | 268 | 312 | 268 |
| 600 | 185 | 0 | 0 | 0 | 0 | 141 | 39 | 181 | 22 | 0 | 0 | 0 | 0 |
| 600 | 317 | 13 | 0 | 83 | 0 | 184 | 13 | 271 | 49 | 46 | 0 | 65 | 0 |
| 600 | 150 | 0 | 0 | 0 | 0 | 85 | 40 | 150 | 103 | 0 | 0 | 0 | 0 |
| 600 | 281 | 168 | 1 | 195 | 0 | 192 | 5 | 237 | 37 | 281 | 299 | 281 | 299 |
| 600 | 239 | 0 | 0 | 0 | 0 | 173 | 60 | 220 | 263 | 0 | 0 | 0 | 0 |
| 600 | 301 | 147 | 0 | 186 | 0 | 194 | 1 | 237 | 0 | 301 | 279 | 301 | 279 |
| 600 | 450 | 233 | 0 | 268 | 0 | 309 | 9 | 390 | 46 | 247 | 0 | 248 | 0 |
| 600 | 204 | 0 | 0 | 0 | 0 | 68 | 12 | 159 | 0 | 204 | 376 | 204 | 376 |
| 600 | 121 | 0 | 0 | 0 | 0 | 27 | 59 | 112 | 266 | 121 | 459 | 121 | 459 |
| 600 | 324 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 600 | 112 | 0 | 0 | 0 | 0 | 25 | 54 | 78 | 157 | 112 | 468 | 112 | 468 |
| 600 | 114 | 0 | 0 | 0 | 0 | 31 | 59 | 92 | 246 | 114 | 466 | 114 | 466 |
| 600 | 144 | 0 | 0 | 0 | 0 | 23 | 6 | 102 | 0 | 0 | 0 | 0 | 0 |
| 600 | 148 | 0 | 0 | 0 | 0 | 99 | 40 | 130 | 60 | 0 | 0 | 0 | 0 |
| 600 | 208 | 0 | 0 | 0 | 0 | 70 | 10 | 187 | 31 | 208 | 372 | 208 | 372 |
| 600 | 334 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 334 | 246 | 334 | 246 |
| 600 | 146 | 0 | 0 | 0 | 0 | 56 | 9 | 107 | 33 | 146 | 434 | 146 | 434 |
| 600 | 317 | 188 | 0 | 243 | 0 | 189 | 0 | 243 | 0 | 169 | 0 | 169 | 0 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 600 | 342 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 600 | 185 | 0 | 0 | 0 | 0 | 101 | 42 | 181 | 76 | 0 | 0 | 0 | 0 |
| 600 | 308 | 59 | 0 | 120 | 0 | 181 | 4 | 276 | 1 | 0 | 0 | 0 | 0 |
| 600 | 256 | 0 | 0 | 0 | 0 | 172 | 37 | 229 | 173 | 0 | 0 | 0 | 0 |
| 600 | 206 | 0 | 0 | 0 | 0 | 49 | 47 | 192 | 112 | 206 | 374 | 206 | 374 |
| 600 | 301 | 0 | 0 | 0 | 0 | 102 | 12 | 273 | 0 | 0 | 0 | 0 | 0 |
| 600 | 221 | 0 | 0 | 0 | 0 | 86 | 19 | 189 | 0 | 221 | 359 | 221 | 359 |
| 600 | 279 | 141 | 0 | 188 | 0 | 172 | 0 | 219 | 0 | 279 | 301 | 279 | 301 |
| 600 | 114 | 0 | 0 | 0 | 0 | 21 | 13 | 95 | 1 | 0 | 0 | 0 | 0 |
| 600 | 267 | 0 | 0 | 0 | 0 | 183 | 40 | 235 | 107 | 267 | 313 | 267 | 313 |
| 600 | 159 | 0 | 0 | 0 | 0 | 52 | 57 | 159 | 242 | 0 | 0 | 0 | 0 |
| 600 | 194 | 0 | 0 | 0 | 0 | 69 | 9 | 186 | 4 | 194 | 386 | 194 | 386 |
| 600 | 403 | 182 | 0 | 210 | 0 | 231 | 0 | 282 | 0 | 403 | 177 | 403 | 177 |
| 600 | 298 | 0 | 0 | 0 | 0 | 185 | 4 | 255 | 0 | 298 | 282 | 298 | 282 |
| 600 | 176 | 10 | 0 | 59 | 0 | 113 | 4 | 145 | 0 | 0 | 0 | 0 | 0 |
| 600 | 206 | 24 | 0 | 70 | 0 | 119 | 2 | 178 | 0 | 0 | 0 | 0 | 0 |
| 600 | 232 | 0 | 0 | 0 | 0 | 143 | 4 | 197 | 0 | 0 | 0 | 0 | 0 |
| 600 | 325 | 159 | 0 | 201 | 0 | 209 | 1 | 247 | 0 | 169 | 0 | 169 | 0 |
| 600 | 425 | 183 | 0 | 230 | 0 | 303 | 4 | 357 | 0 | 425 | 155 | 425 | 155 |
| 600 | 128 | 0 | 0 | 0 | 0 | 34 | 57 | 113 | 200 | 0 | 0 | 0 | 0 |
| 600 | 318 | 135 | 0 | 159 | 0 | 195 | 0 | 213 | 0 | 170 | 0 | 170 | 0 |
| 600 | 195 | 0 | 0 | 0 | 0 | 85 | 6 | 157 | 0 | 195 | 385 | 195 | 385 |
| 600 | 346 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 600 | 181 | 0 | 0 | 0 | 0 | 45 | 3 | 133 | 0 | 181 | 399 | 181 | 399 |
| 600 | 195 | 0 | 0 | 0 | 0 | 94 | 0 | 134 | 0 | 195 | 385 | 195 | 385 |
| 600 | 226 | 0 | 0 | 0 | 0 | 118 | 0 | 169 | 0 | 226 | 354 | 226 | 354 |
| 600 | 203 | 0 | 0 | 0 | 0 | 48 | 21 | 182 | 55 | 203 | 377 | 203 | 377 |
| 600 | 147 | 0 | 0 | 0 | 0 | 32 | 5 | 103 | 0 | 147 | 433 | 147 | 433 |
| 600 | 281 | 110 | 0 | 120 | 0 | 152 | 0 | 179 | 0 | 126 | 0 | 126 | 0 |
| 600 | 369 | 169 | 0 | 199 | 0 | 243 | 0 | 302 | 0 | 369 | 211 | 369 | 211 |
| 600 | 328 | 101 | 0 | 142 | 0 | 187 | 0 | 231 | 0 | 141 | 0 | 141 | 0 |
| 600 | 268 | 0 | 0 | 0 | 0 | 59 | 1 | 211 | 0 | 0 | 0 | 0 | 0 |
| 600 | 280 | 0 | 0 | 0 | 0 | 128 | 0 | 194 | 0 | 280 | 300 | 280 | 300 |
| 600 | 223 | 0 | 0 | 0 | 0 | 116 | 0 | 165 | 0 | 223 | 357 | 223 | 357 |
| 600 | 162 | 0 | 0 | 0 | 0 | 33 | 4 | 124 | 0 | 162 | 418 | 162 | 418 |
| 600 | 410 | 142 | 0 | 156 | 0 | 259 | 1 | 308 | 0 | 172 | 0 | 172 | 0 |
| 600 | 396 | 121 | 0 | 146 | 0 | 216 | 0 | 255 | 0 | 396 | 184 | 396 | 184 |
| 600 | 166 | 0 | 0 | 0 | 0 | 65 | 33 | 152 | 95 | 166 | 414 | 166 | 414 |
| 600 | 249 | 0 | 0 | 0 | 0 | 136 | 15 | 196 | 0 | 249 | 331 | 249 | 331 |
| 600 | 264 | 0 | 0 | 0 | 0 | 55 | 2 | 185 | 0 | 0 | 0 | 0 | 0 |
| 600 | 249 | 0 | 0 | 0 | 0 | 132 | 1 | 177 | 0 | 249 | 331 | 249 | 331 |
| 600 | 159 | 0 | 0 | 0 | 0 | 41 | 32 | 139 | 83 | 159 | 421 | 159 | 421 |
| 600 | 193 | 0 | 0 | 0 | 0 | 41 | 0 | 151 | 0 | 193 | 387 | 193 | 387 |
| 600 | 175 | 0 | 0 | 0 | 0 | 72 | 27 | 158 | 60 | 175 | 405 | 175 | 405 |
| 600 | 378 | 157 | 0 | 203 | 0 | 205 | 0 | 245 | 0 | 153 | 0 | 153 | 0 |
| 600 | 213 | 0 | 0 | 0 | 0 | 50 | 6 | 157 | 38 | 213 | 367 | 213 | 367 |
| 600 | 258 | 0 | 0 | 0 | 0 | 59 | 1 | 191 | 0 | 0 | 0 | 0 | 0 |
| 600 | 166 | 0 | 0 | 0 | 0 | 90 | 42 | 166 | 48 | 0 | 0 | 0 | 0 |
| 600 | 227 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 227 | 353 | 227 | 353 |
| 600 | 261 | 75 | 0 | 88 | 0 | 147 | 0 | 171 | 0 | 261 | 319 | 261 | 319 |
| 600 | 122 | 0 | 0 | 0 | 0 | 27 | 20 | 100 | 0 | 0 | 0 | 0 | 0 |
| 600 | 146 | 0 | 0 | 0 | 0 | 29 | 33 | 135 | 134 | 146 | 434 | 146 | 434 |
| 600 | 250 | 79 | 0 | 94 | 0 | 143 | 0 | 184 | 0 | 250 | 330 | 250 | 330 |
| 600 | 157 | 0 | 0 | 0 | 0 | 58 | 37 | 147 | 135 | 0 | 0 | 0 | 0 |
| 600 | 257 | 0 | 0 | 0 | 0 | 55 | 0 | 185 | 0 | 0 | 0 | 0 | 0 |
| 600 | 290 | 120 | 0 | 155 | 0 | 154 | 0 | 202 | 0 | 131 | 0 | 131 | 0 |
| 600 | 143 | 0 | 0 | 0 | 0 | 29 | 17 | 113 | 30 | 143 | 437 | 143 | 437 |
| 600 | 300 | 12 | 0 | 0 | 0 | 148 | 0 | 196 | 0 | 0 | 0 | 0 | 0 |
| 600 | 120 | 0 | 0 | 0 | 0 | 24 | 32 | 76 | 45 | 0 | 0 | 0 | 0 |
| 600 | 387 | 121 | 0 | 156 | 0 | 206 | 0 | 235 | 0 | 387 | 193 | 387 | 193 |
| 600 | 148 | 0 | 0 | 0 | 0 | 43 | 6 | 109 | 0 | 0 | 0 | 0 | 0 |
| 600 | 217 | 0 | 0 | 0 | 0 | 54 | 0 | 164 | 0 | 217 | 363 | 217 | 363 |
| 600 | 144 | 0 | 0 | 0 | 0 | 30 | 32 | 96 | 77 | 0 | 0 | 0 | 0 |
| 600 | 225 | 0 | 0 | 0 | 0 | 87 | 17 | 211 | 46 | 0 | 0 | 0 | 0 |
| 600 | 162 | 0 | 0 | 0 | 0 | 67 | 20 | 142 | 0 | 162 | 418 | 162 | 418 |
| 600 | 189 | 0 | 0 | 0 | 0 | 44 | 1 | 162 | 0 | 0 | 0 | 0 | 0 |
| 600 | 336 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 600 | 281 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 600 | 123 | 0 | 0 | 0 | 0 | 27 | 33 | 89 | 68 | 123 | 457 | 123 | 457 |
| 600 | 193 | 0 | 0 | 0 | 0 | 43 | 0 | 162 | 0 | 193 | 387 | 193 | 387 |
| 600 | 171 | 0 | 0 | 0 | 0 | 86 | 17 | 139 | 84 | 0 | 0 | 0 | 0 |
| 600 | 334 | 0 | 0 | 0 | 0 | 165 | 0 | 216 | 0 | 334 | 246 | 334 | 246 |
| 600 | 120 | 0 | 0 | 0 | 0 | 16 | 18 | 83 | 41 | 120 | 460 | 120 | 460 |
| 600 | 143 | 0 | 0 | 0 | 0 | 26 | 0 | 88 | 0 | 143 | 437 | 143 | 437 |
| 600 | 301 | 0 | 0 | 0 | 0 | 111 | 0 | 174 | 0 | 0 | 0 | 0 | 0 |
| 600 | 267 | 0 | 0 | 0 | 0 | 41 | 0 | 177 | 0 | 0 | 0 | 0 | 0 |
| 600 | 139 | 0 | 0 | 0 | 0 | 15 | 0 | 31 | 0 | 0 | 0 | 0 | 0 |
| 600 | 385 | 155 | 0 | 190 | 0 | 219 | 0 | 278 | 0 | 128 | 0 | 128 | 0 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 182 | 0 | 0 | 0 | 0 | 32 | 6 | 130 | 2 | 182 | 398 | 182 | 398 |
| | 600 | 211 | 0 | 0 | 0 | 0 | 29 | 10 | 44 | 9 | 0 | 0 | 0 | 0 |
| | 600 | 346 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 346 | 234 | 346 | 234 |
| | 600 | 476 | 78 | 0 | 121 | 0 | 208 | 0 | 260 | 0 | 476 | 104 | 476 | 104 |
| | 600 | 255 | 76 | 0 | 99 | 0 | 111 | 0 | 120 | 0 | 255 | 325 | 255 | 325 |
| | 600 | 200 | 0 | 0 | 0 | 0 | 39 | 0 | 122 | 0 | 200 | 380 | 200 | 380 |
| | 600 | 179 | 0 | 0 | 0 | 0 | 72 | 0 | 114 | 0 | 179 | 401 | 179 | 401 |
| | 600 | 198 | 0 | 0 | 0 | 0 | 32 | 0 | 135 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 128 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 128 | 452 | 128 | 452 |
| | 600 | 262 | 101 | 0 | 113 | 0 | 120 | 0 | 144 | 0 | 93 | 0 | 93 | 0 |
| | 600 | 268 | 84 | 0 | 109 | 0 | 121 | 0 | 160 | 0 | 268 | 312 | 268 | 312 |
| | 600 | 208 | 0 | 0 | 0 | 0 | 35 | 0 | 133 | 0 | 208 | 372 | 208 | 372 |
| | 600 | 340 | 120 | 0 | 147 | 0 | 149 | 0 | 188 | 0 | 340 | 240 | 340 | 240 |
| | 600 | 176 | 0 | 0 | 0 | 0 | 20 | 1 | 36 | 0 | 176 | 404 | 176 | 404 |
| | 600 | 151 | 0 | 0 | 0 | 0 | 19 | 0 | 55 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 469 | 70 | 0 | 86 | 0 | 0 | 0 | 0 | 0 | 469 | 111 | 469 | 111 |
| | 600 | 153 | 0 | 0 | 0 | 0 | 21 | 0 | 77 | 0 | 153 | 427 | 153 | 427 |
| | 600 | 229 | 0 | 0 | 0 | 0 | 100 | 0 | 147 | 0 | 229 | 351 | 229 | 351 |
| | 600 | 415 | 42 | 0 | 69 | 0 | 187 | 0 | 235 | 0 | 106 | 0 | 106 | 0 |
| | 600 | 181 | 0 | 0 | 0 | 0 | 57 | 2 | 137 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 283 | 9 | 0 | 0 | 0 | 137 | 0 | 148 | 0 | 91 | 0 | 91 | 0 |
| | 600 | 185 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 185 | 395 | 185 | 395 |
| | 600 | 229 | 0 | 0 | 0 | 0 | 33 | 0 | 141 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 205 | 0 | 0 | 0 | 0 | 24 | 2 | 113 | 0 | 205 | 375 | 205 | 375 |
| | 600 | 204 | 0 | 0 | 0 | 0 | 28 | 5 | 153 | 0 | 204 | 376 | 204 | 376 |
| | 600 | 132 | 0 | 0 | 0 | 0 | 15 | 5 | 52 | 0 | 132 | 448 | 132 | 448 |
| | 600 | 274 | 61 | 0 | 80 | 0 | 122 | 0 | 150 | 0 | 110 | 0 | 110 | 0 |
| | 600 | 252 | 0 | 0 | 0 | 0 | 41 | 0 | 137 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 185 | 0 | 0 | 0 | 0 | 24 | 7 | 63 | 0 | 185 | 395 | 185 | 395 |
| | 600 | 132 | 0 | 0 | 0 | 0 | 17 | 8 | 53 | 0 | 132 | 448 | 132 | 448 |
| | 600 | 302 | 0 | 0 | 0 | 0 | 117 | 0 | 179 | 0 | 302 | 278 | 302 | 278 |
| | 600 | 247 | 0 | 0 | 0 | 0 | 90 | 0 | 136 | 0 | 247 | 333 | 247 | 333 |
| | 600 | 350 | 65 | 0 | 92 | 0 | 163 | 0 | 191 | 0 | 350 | 230 | 350 | 230 |
| | 600 | 184 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 184 | 396 | 184 | 396 |
| | 600 | 178 | 0 | 0 | 0 | 0 | 48 | 12 | 138 | 0 | 178 | 402 | 178 | 402 |
| | 600 | 154 | 0 | 0 | 0 | 0 | 30 | 0 | 108 | 0 | 154 | 426 | 154 | 426 |
| | 600 | 185 | 0 | 0 | 0 | 0 | 70 | 0 | 121 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 155 | 0 | 0 | 0 | 0 | 41 | 5 | 135 | 8 | 155 | 425 | 155 | 425 |
| | 600 | 429 | 104 | 0 | 127 | 0 | 227 | 0 | 271 | 0 | 190 | 0 | 190 | 0 |
| | 600 | 482 | 85 | 0 | 132 | 0 | 50 | 0 | 90 | 0 | 111 | 0 | 111 | 0 |
| | 600 | 185 | 0 | 0 | 0 | 0 | 77 | 0 | 124 | 0 | 185 | 395 | 185 | 395 |
| | 600 | 239 | 0 | 0 | 0 | 0 | 32 | 0 | 131 | 0 | 239 | 341 | 239 | 341 |
| | 600 | 206 | 0 | 0 | 0 | 0 | 91 | 0 | 134 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 234 | 0 | 0 | 0 | 0 | 34 | 0 | 148 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 288 | 33 | 0 | 66 | 0 | 128 | 0 | 167 | 0 | 288 | 292 | 288 | 292 |
| 0 | 600 | 183 | 111 | 0 | 111 | 0 | 141 | 0 | 141 | 0 | 111 | 0 | 111 | 0 |
| | 600 | 185 | 0 | 0 | 0 | 0 | 149 | 0 | 149 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 146 | 0 | 0 | 0 | 0 | 108 | 0 | 108 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 264 | 0 | 0 | 0 | 0 | 212 | 0 | 212 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 214 | 130 | 0 | 130 | 0 | 164 | 0 | 164 | 0 | 130 | 0 | 130 | 0 |
| | 600 | 176 | 92 | 0 | 92 | 0 | 130 | 0 | 130 | 0 | 80 | 0 | 80 | 0 |
| | 600 | 179 | 109 | 0 | 109 | 0 | 137 | 0 | 137 | 0 | 108 | 0 | 108 | 0 |
| | 600 | 247 | 133 | 0 | 133 | 0 | 189 | 0 | 189 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 128 | 0 | 0 | 0 | 0 | 94 | 0 | 94 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 321 | 195 | 0 | 195 | 0 | 247 | 0 | 247 | 0 | 195 | 0 | 195 | 0 |
| | 600 | 244 | 148 | 0 | 148 | 0 | 188 | 0 | 188 | 0 | 148 | 0 | 148 | 0 |
| | 600 | 168 | 0 | 0 | 0 | 0 | 136 | 0 | 136 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 174 | 0 | 0 | 0 | 0 | 128 | 0 | 128 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 219 | 115 | 0 | 115 | 0 | 163 | 0 | 163 | 0 | 104 | 0 | 104 | 0 |
| | 600 | 262 | 0 | 0 | 0 | 0 | 210 | 0 | 210 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 288 | 220 | 0 | 220 | 0 | 208 | 0 | 208 | 0 | 220 | 0 | 220 | 0 |
| | 600 | 201 | 123 | 0 | 123 | 0 | 155 | 0 | 155 | 0 | 122 | 0 | 122 | 0 |
| | 600 | 339 | 207 | 0 | 207 | 0 | 261 | 0 | 261 | 0 | 205 | 0 | 205 | 0 |
| | 600 | 155 | 0 | 0 | 0 | 0 | 115 | 0 | 115 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 467 | 359 | 0 | 359 | 0 | 339 | 0 | 339 | 0 | 357 | 0 | 357 | 0 |
| | 600 | 198 | 106 | 0 | 106 | 0 | 152 | 0 | 152 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 238 | 124 | 0 | 124 | 0 | 176 | 0 | 176 | 0 | 114 | 0 | 114 | 0 |
| | 600 | 241 | 147 | 0 | 147 | 0 | 185 | 0 | 185 | 0 | 145 | 0 | 145 | 0 |
| | 600 | 220 | 118 | 0 | 118 | 0 | 168 | 0 | 168 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 189 | 0 | 0 | 0 | 0 | 151 | 0 | 151 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 192 | 116 | 0 | 116 | 0 | 148 | 0 | 148 | 0 | 116 | 0 | 116 | 0 |
| | 600 | 245 | 0 | 0 | 0 | 0 | 197 | 0 | 197 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 246 | 150 | 0 | 150 | 0 | 190 | 0 | 190 | 0 | 148 | 0 | 148 | 0 |
| | 600 | 241 | 125 | 0 | 125 | 0 | 177 | 0 | 177 | 0 | 115 | 0 | 115 | 0 |
| | 600 | 179 | 97 | 0 | 97 | 0 | 137 | 0 | 137 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 184 | 112 | 0 | 112 | 0 | 142 | 0 | 142 | 0 | 111 | 0 | 111 | 0 |
| | 600 | 249 | 129 | 0 | 129 | 0 | 183 | 0 | 183 | 0 | 120 | 0 | 120 | 0 |
| | 600 | 202 | 122 | 0 | 122 | 0 | 154 | 0 | 154 | 0 | 122 | 0 | 122 | 0 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 215 | 131 | 0 | 131 | 0 | 165 | 0 | 165 | 0 | 129 | 0 | 129 | 0 |
| | 600 | 158 | 82 | 0 | 82 | 0 | 116 | 0 | 116 | 0 | 69 | 0 | 69 | 0 |
| | 600 | 210 | 128 | 0 | 128 | 0 | 162 | 0 | 162 | 0 | 127 | 0 | 127 | 0 |
| | 600 | 207 | 0 | 0 | 0 | 0 | 153 | 0 | 153 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 206 | 126 | 0 | 126 | 0 | 158 | 0 | 158 | 0 | 125 | 0 | 125 | 0 |
| | 600 | 468 | 358 | 0 | 358 | 0 | 338 | 0 | 338 | 0 | 358 | 0 | 358 | 0 |
| | 600 | 122 | 0 | 0 | 0 | 0 | 90 | 0 | 90 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 158 | 0 | 0 | 0 | 0 | 126 | 0 | 126 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 168 | 88 | 0 | 88 | 0 | 124 | 0 | 124 | 0 | 76 | 0 | 76 | 0 |
| | 600 | 151 | 79 | 0 | 79 | 0 | 111 | 0 | 111 | 0 | 65 | 0 | 65 | 0 |
| | 600 | 317 | 171 | 0 | 171 | 0 | 245 | 0 | 245 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 303 | 231 | 0 | 231 | 0 | 219 | 0 | 219 | 0 | 231 | 0 | 231 | 0 |
| | 600 | 185 | 0 | 0 | 0 | 0 | 137 | 0 | 137 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 167 | 0 | 0 | 0 | 0 | 135 | 0 | 135 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 246 | 128 | 0 | 128 | 0 | 182 | 0 | 182 | 0 | 118 | 0 | 118 | 0 |
| | 600 | 302 | 232 | 0 | 232 | 0 | 218 | 0 | 218 | 0 | 231 | 0 | 231 | 0 |
| | 600 | 378 | 290 | 0 | 290 | 0 | 274 | 0 | 274 | 0 | 289 | 0 | 289 | 0 |
| 0.1 | 600 | 238 | 15 | 0 | 32 | 0 | 164 | 0 | 185 | 0 | 238 | 342 | 238 | 342 |
| | 600 | 208 | 121 | 0 | 139 | 0 | 162 | 0 | 178 | 0 | 118 | 0 | 118 | 0 |
| | 600 | 175 | 11 | 0 | 66 | 0 | 123 | 0 | 139 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 185 | 116 | 0 | 117 | 0 | 142 | 0 | 153 | 0 | 106 | 0 | 106 | 0 |
| | 600 | 194 | 28 | 0 | 104 | 0 | 158 | 1 | 174 | 0 | 194 | 386 | 194 | 386 |
| | 600 | 256 | 116 | 0 | 163 | 0 | 206 | 0 | 226 | 0 | 3 | 0 | 0 | 0 |
| | 600 | 111 | 0 | 0 | 0 | 0 | 26 | 0 | 94 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 198 | 2 | 0 | 0 | 0 | 136 | 0 | 148 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 251 | 51 | 0 | 148 | 0 | 169 | 0 | 187 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 228 | 6 | 0 | 0 | 0 | 163 | 0 | 176 | 0 | 228 | 352 | 228 | 352 |
| | 600 | 151 | 46 | 0 | 91 | 0 | 124 | 0 | 136 | 0 | 37 | 0 | 0 | 0 |
| | 600 | 217 | 28 | 0 | 108 | 0 | 151 | 0 | 173 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 131 | 0 | 0 | 0 | 0 | 41 | 0 | 105 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 284 | 214 | 0 | 230 | 0 | 208 | 0 | 216 | 0 | 217 | 0 | 217 | 0 |
| | 600 | 151 | 0 | 0 | 0 | 0 | 108 | 0 | 120 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 131 | 0 | 0 | 0 | 0 | 44 | 0 | 103 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 228 | 38 | 0 | 127 | 0 | 188 | 2 | 207 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 446 | 308 | 0 | 321 | 0 | 327 | 0 | 341 | 0 | 446 | 134 | 446 | 134 |
| | 600 | 207 | 42 | 0 | 110 | 0 | 171 | 2 | 176 | 0 | 207 | 373 | 207 | 373 |
| | 600 | 150 | 1 | 0 | 0 | 0 | 107 | 0 | 118 | 0 | 150 | 430 | 150 | 430 |
| | 600 | 310 | 236 | 0 | 249 | 0 | 224 | 0 | 238 | 0 | 310 | 270 | 310 | 270 |
| | 600 | 169 | 0 | 0 | 0 | 0 | 46 | 0 | 142 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 262 | 58 | 0 | 146 | 0 | 212 | 0 | 238 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 207 | 42 | 0 | 108 | 0 | 172 | 0 | 189 | 0 | 1 | 0 | 0 | 0 |
| | 600 | 209 | 26 | 0 | 104 | 0 | 144 | 0 | 155 | 0 | 1 | 0 | 0 | 0 |
| | 600 | 211 | 127 | 0 | 132 | 0 | 157 | 0 | 163 | 0 | 211 | 369 | 211 | 369 |
| | 600 | 148 | 0 | 0 | 0 | 0 | 37 | 0 | 114 | 0 | 148 | 432 | 148 | 432 |
| | 600 | 239 | 48 | 0 | 128 | 0 | 195 | 0 | 210 | 0 | 3 | 0 | 0 | 0 |
| | 600 | 226 | 45 | 0 | 136 | 0 | 183 | 2 | 198 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 296 | 224 | 0 | 235 | 0 | 211 | 0 | 217 | 0 | 296 | 284 | 296 | 284 |
| | 600 | 237 | 140 | 0 | 157 | 0 | 176 | 0 | 189 | 0 | 139 | 0 | 139 | 0 |
| | 600 | 117 | 0 | 0 | 0 | 0 | 26 | 0 | 95 | 0 | 117 | 463 | 117 | 463 |
| | 600 | 209 | 52 | 0 | 120 | 0 | 159 | 1 | 185 | 0 | 209 | 371 | 209 | 371 |
| | 600 | 268 | 20 | 0 | 0 | 0 | 193 | 0 | 217 | 0 | 268 | 312 | 268 | 312 |
| | 600 | 405 | 276 | 0 | 300 | 0 | 302 | 0 | 317 | 0 | 405 | 175 | 405 | 175 |
| | 600 | 319 | 177 | 0 | 198 | 0 | 238 | 0 | 261 | 0 | 175 | 0 | 175 | 0 |
| | 600 | 158 | 0 | 0 | 0 | 0 | 36 | 0 | 112 | 0 | 158 | 422 | 158 | 422 |
| | 600 | 150 | 4 | 0 | 0 | 0 | 100 | 0 | 108 | 0 | 150 | 430 | 150 | 430 |
| | 600 | 138 | 4 | 0 | 0 | 0 | 104 | 0 | 111 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 210 | 1 | 0 | 0 | 0 | 141 | 0 | 160 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 381 | 287 | 0 | 297 | 0 | 276 | 0 | 286 | 0 | 381 | 199 | 381 | 199 |
| | 600 | 280 | 153 | 0 | 168 | 0 | 219 | 0 | 243 | 0 | 280 | 300 | 280 | 300 |
| | 600 | 295 | 225 | 0 | 245 | 0 | 212 | 0 | 213 | 0 | 295 | 285 | 295 | 285 |
| | 600 | 239 | 139 | 0 | 157 | 0 | 194 | 1 | 215 | 0 | 239 | 341 | 239 | 341 |
| | 600 | 146 | 31 | 0 | 80 | 0 | 114 | 1 | 120 | 0 | 2 | 0 | 0 | 0 |
| | 600 | 205 | 0 | 0 | 0 | 0 | 147 | 0 | 160 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 272 | 173 | 0 | 184 | 0 | 201 | 0 | 223 | 0 | 156 | 0 | 156 | 0 |
| | 600 | 217 | 37 | 0 | 121 | 0 | 151 | 0 | 158 | 0 | 217 | 363 | 217 | 363 |
| | 600 | 186 | 0 | 0 | 0 | 0 | 136 | 0 | 153 | 0 | 186 | 394 | 186 | 394 |
| | 600 | 316 | 200 | 0 | 222 | 0 | 240 | 0 | 264 | 0 | 316 | 264 | 316 | 264 |
| 0.2 | 600 | 250 | 16 | 0 | 72 | 0 | 194 | 69 | 250 | 295 | 0 | 0 | 0 | 0 |
| | 600 | 228 | 115 | 0 | 154 | 0 | 167 | 10 | 203 | 72 | 228 | 352 | 228 | 352 |
| | 600 | 181 | 0 | 0 | 0 | 0 | 141 | 33 | 171 | 72 | 0 | 0 | 0 | 0 |
| | 600 | 185 | 39 | 0 | 120 | 0 | 139 | 58 | 164 | 107 | 0 | 0 | 0 | 0 |
| | 600 | 179 | 2 | 0 | 0 | 0 | 144 | 64 | 172 | 135 | 179 | 401 | 179 | 401 |
| | 600 | 268 | 10 | 0 | 0 | 0 | 199 | 34 | 241 | 76 | 0 | 0 | 0 | 0 |
| | 600 | 250 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 181 | 85 | 0 | 121 | 0 | 129 | 13 | 149 | 61 | 181 | 399 | 181 | 399 |
| | 600 | 226 | 101 | 0 | 141 | 0 | 179 | 24 | 210 | 70 | 226 | 354 | 226 | 354 |
| | 600 | 136 | 8 | 0 | 0 | 0 | 112 | 64 | 136 | 271 | 136 | 444 | 136 | 444 |
| | 600 | 252 | 125 | 0 | 155 | 0 | 180 | 12 | 219 | 42 | 133 | 0 | 137 | 0 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 308 | 209 | 0 | 226 | 0 | 229 | 4 | 240 | 0 | 225 | 0 | 225 | 0 |
| | 600 | 283 | 209 | 4 | 229 | 0 | 211 | 4 | 242 | 0 | 207 | 0 | 207 | 0 |
| | 600 | 244 | 141 | 0 | 163 | 0 | 184 | 25 | 230 | 87 | 131 | 0 | 131 | 0 |
| | 600 | 212 | 41 | 0 | 138 | 0 | 163 | 46 | 207 | 132 | 6 | 0 | 0 | 0 |
| | 600 | 382 | 252 | 0 | 282 | 0 | 292 | 6 | 341 | 30 | 275 | 0 | 275 | 0 |
| | 600 | 122 | 0 | 0 | 0 | 0 | 74 | 138 | 122 | 458 | 122 | 458 | 122 | 458 |
| | 600 | 212 | 0 | 0 | 0 | 0 | 70 | 52 | 209 | 181 | 0 | 0 | 0 | 0 |
| | 600 | 249 | 145 | 0 | 176 | 0 | 173 | 9 | 224 | 3 | 142 | 0 | 143 | 0 |
| | 600 | 304 | 213 | 0 | 247 | 0 | 223 | 2 | 247 | 0 | 304 | 276 | 304 | 276 |
| | 600 | 253 | 42 | 0 | 120 | 0 | 188 | 29 | 223 | 65 | 253 | 327 | 253 | 327 |
| | 600 | 255 | 119 | 0 | 160 | 0 | 193 | 8 | 237 | 11 | 255 | 325 | 255 | 325 |
| | 600 | 144 | 0 | 0 | 0 | 0 | 119 | 73 | 144 | 341 | 144 | 436 | 144 | 436 |
| | 600 | 251 | 20 | 0 | 76 | 0 | 194 | 35 | 240 | 110 | 251 | 329 | 251 | 329 |
| | 600 | 180 | 80 | 0 | 99 | 0 | 136 | 12 | 156 | 0 | 99 | 0 | 99 | 0 |
| | 600 | 186 | 0 | 0 | 0 | 0 | 150 | 48 | 186 | 235 | 186 | 394 | 186 | 394 |
| | 600 | 120 | 6 | 0 | 0 | 0 | 62 | 97 | 120 | 405 | 0 | 0 | 0 | 0 |
| | 600 | 229 | 28 | 0 | 84 | 0 | 165 | 30 | 216 | 86 | 229 | 351 | 229 | 351 |
| | 600 | 141 | 5 | 0 | 0 | 0 | 114 | 68 | 141 | 267 | 141 | 439 | 141 | 439 |
| | 600 | 177 | 84 | 0 | 116 | 0 | 131 | 12 | 160 | 72 | 177 | 403 | 177 | 403 |
| | 600 | 124 | 0 | 0 | 0 | 0 | 36 | 59 | 120 | 240 | 124 | 456 | 124 | 456 |
| | 600 | 200 | 121 | 0 | 150 | 0 | 148 | 19 | 192 | 82 | 119 | 0 | 121 | 0 |
| | 600 | 145 | 0 | 0 | 0 | 0 | 34 | 66 | 138 | 202 | 0 | 0 | 0 | 0 |
| | 600 | 196 | 23 | 0 | 109 | 0 | 155 | 31 | 192 | 50 | 196 | 384 | 196 | 384 |
| | 600 | 274 | 21 | 0 | 67 | 0 | 202 | 31 | 274 | 105 | 274 | 306 | 274 | 306 |
| | 600 | 292 | 123 | 0 | 167 | 0 | 224 | 25 | 274 | 96 | 165 | 0 | 165 | 0 |
| | 600 | 116 | 0 | 0 | 0 | 0 | 52 | 81 | 113 | 281 | 116 | 464 | 116 | 464 |
| | 600 | 244 | 153 | 0 | 187 | 0 | 175 | 3 | 226 | 0 | 244 | 336 | 244 | 336 |
| | 600 | 254 | 131 | 0 | 175 | 0 | 202 | 22 | 238 | 82 | 132 | 0 | 132 | 0 |
| | 600 | 226 | 34 | 0 | 133 | 0 | 167 | 44 | 215 | 226 | 226 | 354 | 226 | 354 |
| | 600 | 218 | 5 | 0 | 0 | 0 | 170 | 48 | 218 | 113 | 0 | 0 | 0 | 0 |
| | 600 | 203 | 0 | 0 | 0 | 0 | 30 | 1 | 108 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 186 | 10 | 0 | 0 | 0 | 151 | 54 | 186 | 274 | 0 | 0 | 0 | 0 |
| | 600 | 275 | 213 | 14 | 219 | 73 | 227 | 22 | 258 | 95 | 206 | 0 | 206 | 0 |
| | 600 | 152 | 4 | 0 | 0 | 0 | 120 | 63 | 148 | 209 | 0 | 0 | 0 | 0 |
| | 600 | 230 | 5 | 0 | 0 | 0 | 183 | 49 | 230 | 158 | 0 | 0 | 0 | 0 |
| | 600 | 184 | 21 | 0 | 79 | 0 | 138 | 27 | 169 | 37 | 0 | 0 | 0 | 0 |
| | 600 | 250 | 17 | 0 | 37 | 0 | 184 | 31 | 238 | 68 | 250 | 330 | 250 | 330 |
| | 600 | 345 | 0 | 0 | 0 | 0 | 271 | 47 | 336 | 168 | 345 | 235 | 345 | 235 |
| | 600 | 177 | 80 | 0 | 104 | 0 | 125 | 8 | 147 | 0 | 93 | 0 | 93 | 0 |
| 0 3 | 600 | 144 | 17 | 0 | 52 | 0 | 119 | 140 | 136 | 421 | 144 | 436 | 144 | 436 |
| | 600 | 187 | 37 | 9 | 129 | 0 | 154 | 117 | 187 | 383 | 187 | 393 | 187 | 393 |
| | 600 | 173 | 0 | 0 | 0 | 0 | 64 | 90 | 173 | 351 | 173 | 407 | 173 | 407 |
| | 600 | 203 | 92 | 24 | 197 | 12 | 164 | 88 | 203 | 291 | 203 | 377 | 203 | 377 |
| | 600 | 149 | 5 | 0 | 0 | 0 | 103 | 78 | 149 | 351 | 0 | 0 | 0 | 0 |
| | 600 | 212 | 0 | 0 | 0 | 0 | 55 | 52 | 196 | 180 | 212 | 368 | 212 | 368 |
| | 600 | 163 | 17 | 0 | 70 | 0 | 114 | 59 | 157 | 153 | 163 | 417 | 163 | 417 |
| | 600 | 223 | 107 | 2 | 185 | 0 | 172 | 66 | 203 | 248 | 107 | 0 | 107 | 0 |
| | 600 | 280 | 11 | 0 | 0 | 0 | 226 | 81 | 280 | 280 | 280 | 300 | 280 | 300 |
| | 600 | 331 | 219 | 7 | 277 | 31 | 283 | 51 | 321 | 171 | 331 | 249 | 331 | 249 |
| | 600 | 263 | 2 | 0 | 0 | 0 | 31 | 1 | 114 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 332 | 233 | 16 | 292 | 62 | 254 | 24 | 316 | 95 | 332 | 248 | 332 | 248 |
| | 600 | 208 | 39 | 3 | 156 | 0 | 171 | 102 | 208 | 368 | 208 | 372 | 208 | 372 |
| | 600 | 214 | 122 | 7 | 197 | 1 | 155 | 44 | 205 | 106 | 214 | 366 | 214 | 366 |
| | 600 | 460 | 261 | 0 | 319 | 0 | 348 | 18 | 420 | 76 | 460 | 120 | 460 | 120 |
| | 600 | 124 | 0 | 0 | 0 | 0 | 50 | 141 | 124 | 448 | 0 | 0 | 0 | 0 |
| | 600 | 171 | 42 | 3 | 122 | 0 | 150 | 133 | 171 | 391 | 171 | 409 | 171 | 409 |
| | 600 | 325 | 5 | 0 | 0 | 0 | 36 | 0 | 154 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 228 | 49 | 6 | 159 | 0 | 174 | 95 | 228 | 334 | 0 | 0 | 0 | 0 |
| | 600 | 138 | 0 | 0 | 0 | 0 | 86 | 90 | 136 | 362 | 0 | 0 | 0 | 0 |
| | 600 | 263 | 211 | 41 | 263 | 103 | 226 | 60 | 263 | 273 | 263 | 317 | 263 | 317 |
| | 600 | 284 | 212 | 20 | 250 | 109 | 230 | 54 | 273 | 196 | 223 | 0 | 225 | 0 |
| | 600 | 345 | 272 | 22 | 339 | 109 | 297 | 40 | 339 | 134 | 252 | 0 | 252 | 0 |
| | 600 | 240 | 8 | 0 | 40 | 0 | 143 | 81 | 230 | 277 | 0 | 0 | 0 | 0 |
| | 600 | 374 | 275 | 15 | 349 | 7 | 316 | 47 | 365 | 138 | 254 | 0 | 254 | 0 |
| | 600 | 195 | 29 | 0 | 127 | 0 | 147 | 97 | 195 | 379 | 195 | 385 | 195 | 385 |
| | 600 | 233 | 122 | 7 | 201 | 1 | 194 | 101 | 233 | 313 | 233 | 347 | 233 | 347 |
| | 600 | 146 | 0 | 0 | 0 | 0 | 61 | 111 | 146 | 409 | 0 | 0 | 0 | 0 |
| | 600 | 435 | 284 | 7 | 354 | 0 | 340 | 16 | 415 | 88 | 303 | 0 | 309 | 0 |
| | 600 | 237 | 54 | 0 | 102 | 0 | 190 | 80 | 237 | 306 | 237 | 343 | 237 | 343 |
| | 600 | 236 | 67 | 0 | 123 | 0 | 174 | 52 | 236 | 234 | 128 | 0 | 128 | 0 |
| | 600 | 239 | 14 | 0 | 49 | 0 | 168 | 64 | 239 | 260 | 239 | 341 | 239 | 341 |
| | 600 | 369 | 266 | 14 | 330 | 43 | 310 | 41 | 350 | 114 | 369 | 211 | 369 | 211 |
| | 600 | 258 | 41 | 2 | 131 | 0 | 206 | 92 | 258 | 316 | 0 | 0 | 0 | 0 |
| | 600 | 241 | 116 | 0 | 185 | 0 | 175 | 40 | 241 | 175 | 241 | 339 | 241 | 339 |
| | 600 | 183 | 16 | 0 | 43 | 0 | 149 | 117 | 183 | 412 | 183 | 397 | 183 | 397 |
| | 600 | 238 | 60 | 16 | 170 | 0 | 185 | 100 | 224 | 336 | 0 | 0 | 0 | 0 |
| | 600 | 162 | 29 | 1 | 117 | 0 | 129 | 102 | 155 | 405 | 0 | 0 | 0 | 0 |
| | 600 | 370 | 259 | 14 | 324 | 56 | 272 | 20 | 332 | 91 | 267 | 0 | 270 | 0 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 600 | 196 | 18 | 6 | 57 | 0 | 165 | 112 | 196 | 379 | 196 | 384 | 196 | 384 |
| | 600 | 316 | 205 | 6 | 244 | 0 | 231 | 17 | 294 | 54 | 224 | 0 | 224 | 0 |
| | 600 | 373 | 201 | 0 | 260 | 0 | 289 | 24 | 338 | 53 | 373 | 207 | 373 | 207 |
| | 600 | 305 | 204 | 6 | 279 | 18 | 226 | 13 | 279 | 30 | 305 | 275 | 305 | 275 |
| | 600 | 170 | 0 | 0 | 0 | 0 | 68 | 105 | 170 | 405 | 170 | 410 | 170 | 410 |
| | 600 | 277 | 208 | 21 | 259 | 59 | 216 | 30 | 260 | 98 | 203 | 0 | 203 | 0 |
| | 600 | 138 | 9 | 0 | 0 | 0 | 109 | 125 | 138 | 428 | 138 | 442 | 138 | 442 |
| | 600 | 275 | 193 | 10 | 240 | 43 | 209 | 20 | 247 | 84 | 275 | 305 | 275 | 305 |
| | 600 | 286 | 83 | 21 | 263 | 59 | 37 | 0 | 153 | 0 | 0 | 0 | 0 | 0 |
| | 600 | 308 | 137 | 2 | 223 | 0 | 240 | 50 | 293 | 231 | 308 | 272 | 308 | 272 |
| | 600 | 392 | 225 | 2 | 293 | 0 | 307 | 39 | 372 | 141 | 392 | 188 | 392 | 188 |
| 0.4 | 600 | 347 | 209 | 16 | 294 | 62 | 289 | 62 | 347 | 218 | 257 | 0 | 273 | 0 |
| | 600 | 380 | 213 | 16 | 297 | 38 | 168 | 4 | 189 | 0 | 272 | 0 | 272 | 0 |
| | 600 | 198 | 23 | 10 | 97 | 0 | 139 | 106 | 184 | 387 | 198 | 382 | 198 | 382 |
| | 600 | 191 | 8 | 7 | 0 | 0 | 66 | 112 | 180 | 393 | 0 | 0 | 0 | 0 |
| | 600 | 205 | 23 | 8 | 92 | 0 | 169 | 141 | 205 | 356 | 205 | 375 | 205 | 375 |
| | 600 | 142 | 0 | 0 | 0 | 0 | 50 | 113 | 142 | 349 | 0 | 0 | 0 | 0 |
| | 600 | 303 | 161 | 8 | 235 | 19 | 220 | 39 | 286 | 164 | 303 | 277 | 303 | 277 |
| | 600 | 311 | 103 | 12 | 226 | 0 | 258 | 102 | 311 | 279 | 107 | 0 | 107 | 0 |
| | 600 | 273 | 169 | 12 | 229 | 22 | 213 | 39 | 273 | 181 | 192 | 0 | 194 | 0 |
| | 600 | 137 | 21 | 24 | 28 | 106 | 44 | 98 | 137 | 334 | 137 | 443 | 137 | 443 |
| | 600 | 228 | 30 | 8 | 106 | 0 | 182 | 100 | 228 | 350 | 0 | 0 | 0 | 0 |
| | 600 | 268 | 36 | 10 | 165 | 2 | 61 | 25 | 166 | 88 | 268 | 312 | 268 | 312 |
| | 600 | 338 | 2 | 0 | 0 | 0 | 66 | 20 | 263 | 34 | 338 | 242 | 338 | 242 |
| | 600 | 131 | 6 | 7 | 0 | 0 | 61 | 147 | 131 | 460 | 131 | 449 | 131 | 449 |
| | 600 | 226 | 16 | 1 | 56 | 0 | 187 | 127 | 226 | 338 | 226 | 354 | 226 | 354 |
| | 600 | 403 | 245 | 12 | 352 | 75 | 338 | 42 | 393 | 176 | 262 | 0 | 268 | 0 |
| | 600 | 295 | 142 | 22 | 225 | 99 | 234 | 90 | 295 | 273 | 295 | 285 | 295 | 285 |
| | 600 | 162 | 0 | 0 | 0 | 0 | 127 | 134 | 162 | 429 | 162 | 418 | 162 | 418 |
| | 600 | 217 | 2 | 0 | 0 | 0 | 152 | 109 | 217 | 360 | 217 | 363 | 217 | 363 |
| | 600 | 200 | 81 | 11 | 165 | 21 | 155 | 104 | 200 | 390 | 200 | 380 | 200 | 380 |
| | 600 | 284 | 55 | 2 | 181 | 0 | 235 | 87 | 284 | 296 | 136 | 0 | 143 | 0 |
| | 600 | 367 | 214 | 10 | 323 | 12 | 292 | 46 | 367 | 180 | 367 | 213 | 367 | 213 |
| | 600 | 138 | 10 | 17 | 6 | 53 | 71 | 169 | 138 | 456 | 138 | 442 | 138 | 442 |
| | 600 | 237 | 6 | 0 | 0 | 0 | 185 | 108 | 237 | 358 | 0 | 0 | 0 | 0 |
| | 600 | 285 | 213 | 30 | 284 | 97 | 242 | 81 | 285 | 249 | 285 | 295 | 285 | 295 |
| | 600 | 204 | 1 | 0 | 0 | 0 | 137 | 106 | 204 | 350 | 204 | 376 | 204 | 376 |
| | 600 | 270 | 144 | 9 | 181 | 0 | 203 | 44 | 264 | 209 | 178 | 0 | 185 | 0 |
| | 600 | 173 | 0 | 0 | 0 | 0 | 140 | 150 | 173 | 421 | 0 | 0 | 0 | 0 |
| | 600 | 197 | 20 | 6 | 65 | 0 | 161 | 129 | 197 | 395 | 197 | 383 | 197 | 383 |
| | 600 | 154 | 0 | 0 | 0 | 0 | 120 | 129 | 154 | 410 | 154 | 426 | 154 | 426 |
| | 600 | 307 | 194 | 18 | 277 | 44 | 262 | 76 | 307 | 247 | 307 | 273 | 307 | 273 |
| | 600 | 170 | 4 | 0 | 0 | 0 | 120 | 108 | 170 | 381 | 0 | 0 | 0 | 0 |
| | 600 | 218 | 23 | 8 | 86 | 0 | 39 | 29 | 148 | 15 | 0 | 0 | 0 | 0 |
| | 600 | 208 | 89 | 18 | 162 | 53 | 171 | 103 | 208 | 341 | 87 | 0 | 87 | 0 |
| | 600 | 328 | 0 | 0 | 0 | 0 | 74 | 34 | 246 | 150 | 328 | 252 | 328 | 252 |
| | 600 | 303 | 244 | 66 | 303 | 255 | 256 | 75 | 303 | 260 | 303 | 277 | 303 | 277 |
| | 600 | 224 | 88 | 18 | 182 | 50 | 174 | 100 | 224 | 365 | 224 | 356 | 224 | 356 |
| | 600 | 276 | 215 | 53 | 276 | 199 | 235 | 78 | 276 | 239 | 276 | 304 | 276 | 304 |
| | 600 | 168 | 22 | 25 | 64 | 0 | 147 | 162 | 168 | 422 | 0 | 0 | 0 | 0 |
| | 600 | 179 | 63 | 7 | 92 | 14 | 133 | 83 | 179 | 386 | 179 | 401 | 179 | 401 |
| | 600 | 216 | 1 | 0 | 0 | 0 | 156 | 105 | 216 | 378 | 0 | 0 | 0 | 0 |
| | 600 | 174 | 21 | 4 | 105 | 0 | 143 | 120 | 174 | 418 | 174 | 406 | 174 | 406 |
| | 600 | 447 | 247 | 10 | 364 | 48 | 1 | 0 | 0 | 0 | 447 | 133 | 447 | 133 |
| | 600 | 478 | 212 | 1 | 303 | 0 | 1 | 0 | 0 | 0 | 319 | 0 | 319 | 0 |
| | 600 | 250 | 13 | 0 | 0 | 0 | 203 | 99 | 250 | 337 | 250 | 330 | 250 | 330 |
| | 600 | 167 | 9 | 2 | 0 | 0 | 85 | 119 | 167 | 407 | 167 | 413 | 167 | 413 |
| | 600 | 316 | 119 | 13 | 218 | 0 | 250 | 78 | 316 | 260 | 316 | 264 | 316 | 264 |
| | 600 | 225 | 45 | 24 | 178 | 80 | 187 | 125 | 225 | 316 | 225 | 355 | 225 | 355 |
| | 600 | 194 | 0 | 0 | 0 | 0 | 51 | 78 | 155 | 370 | 194 | 386 | 194 | 386 |
| | 600 | 252 | 50 | 24 | 184 | 78 | 193 | 115 | 252 | 262 | 252 | 328 | 252 | 328 |
| 0.5 | 600 | 188 | 20 | 19 | 0 | 0 | 68 | 99 | 188 | 361 | 188 | 392 | 188 | 392 |
| | 600 | 225 | 9 | 7 | 0 | 0 | 38 | 42 | 121 | 77 | 0 | 0 | 0 | 0 |
| | 600 | 210 | 49 | 52 | 180 | 222 | 155 | 132 | 210 | 383 | 32 | 0 | 0 | 0 |
| | 600 | 122 | 0 | 0 | 0 | 0 | 49 | 147 | 122 | 454 | 0 | 0 | 0 | 0 |
| | 600 | 158 | 16 | 13 | 57 | 1 | 78 | 141 | 158 | 429 | 158 | 422 | 158 | 422 |
| | 600 | 140 | 0 | 0 | 0 | 0 | 54 | 135 | 140 | 452 | 140 | 440 | 140 | 440 |
| | 600 | 150 | 0 | 0 | 0 | 0 | 72 | 126 | 132 | 398 | 150 | 430 | 150 | 430 |
| | 600 | 108 | 10 | 7 | 45 | 0 | 42 | 143 | 108 | 411 | 0 | 0 | 0 | 0 |
| | 600 | 292 | 41 | 10 | 132 | 45 | 217 | 97 | 288 | 288 | 55 | 0 | 76 | 0 |
| | 600 | 244 | 77 | 67 | 244 | 233 | 181 | 120 | 244 | 335 | 13 | 0 | 0 | 0 |
| | 600 | 183 | 35 | 20 | 167 | 34 | 154 | 141 | 183 | 409 | 183 | 397 | 183 | 397 |
| | 600 | 197 | 39 | 9 | 112 | 0 | 140 | 87 | 197 | 301 | 88 | 0 | 93 | 0 |
| | 600 | 219 | 100 | 45 | 219 | 160 | 185 | 131 | 219 | 376 | 92 | 0 | 101 | 0 |
| | 600 | 440 | 182 | 8 | 310 | 6 | 35 | 0 | 82 | 0 | 440 | 140 | 440 | 140 |
| | 600 | 365 | 117 | 5 | 236 | 0 | 286 | 68 | 365 | 226 | 240 | 0 | 251 | 0 |
| | 600 | 165 | 2 | 1 | 0 | 0 | 129 | 150 | 165 | 407 | 0 | 0 | 0 | 0 |
| | 600 | 207 | 58 | 11 | 179 | 0 | 155 | 124 | 207 | 366 | 207 | 373 | 207 | 373 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 600 | 249 | 105 | 44 | 186 | 142 | 193 | 100 | 249 | 296 | 249 | 331 | 249 | 331 |
| 600 | 245 | 2 | 0 | 0 | 0 | 205 | 139 | 245 | 350 | 0 | 0 | 0 | 0 |
| 600 | 149 | 16 | 18 | 68 | 43 | 77 | 143 | 149 | 437 | 149 | 431 | 149 | 431 |
| 600 | 275 | 29 | 14 | 149 | 37 | 218 | 114 | 275 | 308 | 275 | 305 | 275 | 305 |
| 600 | 153 | 16 | 18 | 88 | 44 | 68 | 116 | 153 | 421 | 0 | 0 | 0 | 0 |
| 600 | 363 | 116 | 6 | 217 | 2 | 271 | 59 | 363 | 195 | 363 | 217 | 363 | 217 |
| 600 | 117 | 28 | 40 | 115 | 152 | 56 | 121 | 117 | 433 | 0 | 0 | 0 | 0 |
| 600 | 280 | 16 | 3 | 0 | 0 | 182 | 103 | 280 | 306 | 280 | 300 | 280 | 300 |
| 600 | 455 | 64 | 1 | 126 | 0 | 26 | 0 | 90 | 0 | 455 | 125 | 455 | 125 |
| 600 | 394 | 78 | 5 | 135 | 0 | 315 | 52 | 393 | 139 | 212 | 0 | 214 | 0 |
| 600 | 147 | 41 | 73 | 120 | 307 | 109 | 168 | 147 | 448 | 147 | 433 | 147 | 433 |
| 600 | 128 | 7 | 12 | 0 | 0 | 48 | 146 | 128 | 461 | 128 | 452 | 128 | 452 |
| 600 | 162 | 3 | 6 | 0 | 0 | 85 | 134 | 162 | 429 | 162 | 418 | 162 | 418 |
| 600 | 276 | 41 | 18 | 166 | 44 | 213 | 109 | 276 | 317 | 0 | 0 | 0 | 0 |
| 600 | 317 | 32 | 5 | 99 | 0 | 245 | 106 | 317 | 274 | 317 | 263 | 317 | 263 |
| 600 | 277 | 68 | 19 | 207 | 73 | 232 | 113 | 277 | 314 | 277 | 303 | 277 | 303 |
| 600 | 240 | 21 | 8 | 68 | 0 | 154 | 123 | 240 | 351 | 240 | 340 | 240 | 340 |
| 600 | 178 | 42 | 38 | 133 | 122 | 138 | 150 | 178 | 398 | 0 | 0 | 0 | 0 |
| 600 | 250 | 3 | 4 | 0 | 0 | 138 | 113 | 250 | 340 | 0 | 0 | 0 | 0 |
| 600 | 109 | 0 | 0 | 0 | 0 | 41 | 156 | 109 | 482 | 109 | 471 | 109 | 471 |
| 600 | 303 | 3 | 0 | 0 | 0 | 71 | 52 | 269 | 137 | 303 | 277 | 303 | 277 |
| 600 | 289 | 52 | 34 | 212 | 165 | 237 | 103 | 289 | 305 | 289 | 291 | 289 | 291 |
| 600 | 241 | 56 | 35 | 200 | 92 | 176 | 105 | 241 | 340 | 241 | 339 | 241 | 339 |
| 600 | 166 | 32 | 61 | 120 | 283 | 105 | 150 | 166 | 406 | 3 | 0 | 0 | 0 |
| 600 | 202 | 20 | 20 | 4 | 34 | 130 | 133 | 202 | 390 | 202 | 378 | 202 | 378 |
| 600 | 308 | 185 | 32 | 297 | 120 | 233 | 56 | 308 | 190 | 190 | 0 | 201 | 0 |
| 600 | 277 | 159 | 56 | 277 | 214 | 222 | 112 | 277 | 302 | 277 | 303 | 277 | 303 |
| 600 | 245 | 19 | 9 | 71 | 0 | 200 | 119 | 245 | 321 | 245 | 335 | 245 | 335 |
| 600 | 241 | 0 | 0 | 0 | 0 | 137 | 101 | 241 | 318 | 0 | 0 | 0 | 0 |
| 600 | 178 | 16 | 7 | 42 | 0 | 130 | 157 | 178 | 417 | 178 | 402 | 178 | 402 |
| 600 | 187 | 20 | 19 | 103 | 30 | 146 | 149 | 187 | 404 | 0 | 0 | 0 | 0 |
| 600 | 338 | 191 | 23 | 297 | 41 | 289 | 89 | 338 | 247 | 197 | 0 | 214 | 0 |
| 600 | 312 | 166 | 20 | 271 | 73 | 243 | 72 | 291 | 266 | 206 | 0 | 213 | 0 |

# Table B.16 Jitter Effects Analysis

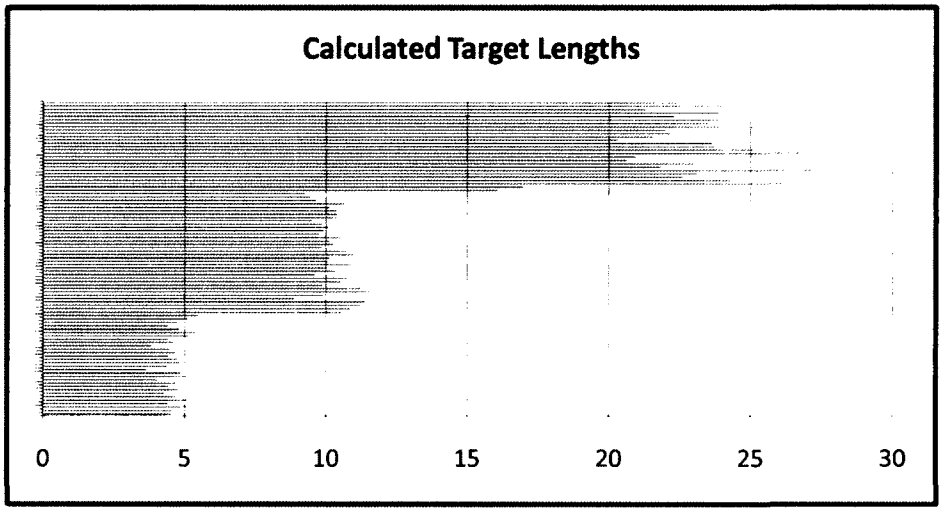
| Duration | | | | Threshold | | | | FAPI | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Tot D | Tot FA | FA D | FA FA | Tot D | Tot FA | FA D | FA FA | Tot D | Tot FA | FA D | FA FA |
| 78.61% | 19.58% | 99.54% | 85.68 % | 83.16% | 29.98% | 97.20% | 85.61% | 91.65% | 48.43% | 91.39% | 48.66% |
| 54.00% | 5.12% | 90.09% | 31.09 % | 76.11% | 23.51% | 92.04% | 77.30% | 86.31% | 62.06% | 85.36% | 62.06% |
| 40.65% | 0.35% | 65.23% | 1.51% | 72.38% | 14.43% | 92.79% | 52.06% | 82.74% | 62.76% | 82.83% | 62.76% |
| 18.80% | 0.05% | 28.90% | 0.25% | 60.16% | 10.38% | 90.75% | 34.61% | 65.05% | 54.52% | 65.11% | 54.52% |
| 17.20% | 0.01% | 23.45% | 0.00% | 48.35% | 5.21% | 74.89% | 14.70% | 51.54% | 41.34% | 51.64% | 41.34% |
| 10.55% | 0.00% | 12.75% | 0.00% | 38.09% | 2.57% | 69.25% | 5.60% | 61.81% | 56.42% | 61.81% | 56.42% |
| 8.91% | 0.00% | 11.77% | 0.00% | 28.00% | 0.35% | 49.90% | 0.11% | 65.01% | 61.12% | 65.01% | 61.12% |
| Vary Noise, N=2 | | | | | | | | | | | |
| 40.65% | 0.35% | 61.00% | 23.50 % | 69.86% | 14.00% | 92.12% | 48.56% | 80.63% | 63.14% | 80.76% | 63.14% |
| 45.62% | 0.00% | 45.62% | 0.00% | 73.48% | 0.00% | 73.48% | 0.00% | 39.27% | 0.00% | 39.27% | 0.00% |
| 34.64% | 0.00% | 46.96% | 0.00% | 70.43% | 0.05% | 80.69% | 0.00% | 59.18% | 40.83% | 58.76% | 40.83% |
| 27.84% | 0.09% | 39.23% | 0.38% | 69.70% | 9.47% | 89.47% | 32.27% | 61.23% | 44.73% | 61.24% | 44.73% |
| 40.25% | 1.78% | 61.61% | 4.10% | 70.71% | 19.09% | 93.05% | 67.68% | 72.60% | 55.35% | 72.68% | 55.35% |
| 30.83% | 3.11% | 51.77% | 8.09% | 63.46% | 25.23% | 87.53% | 81.86% | 76.65% | 61.77% | 76.96% | 61.77% |

# Table B.17 Velocity and Length Testing Results

| S Rate | Pixel X | Pixel Y | Noise T | Info# | N | FAR1 | FAR2 | Analysis | C L | Length | Speed | Actual | length | %Diff L | C V | %Diff V | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 120 | 30 | 30 | 0.01 | 1 | 2.6 | 14 | 50 | T/P | | 4.54 | 30.73 | 29.06 | 4.57 | 0.0066 | | 0.0575 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.8 | 10 | 50 | T/P | | 4.59 | 24.25 | 23.88 | | 0.0044 | | 0.0155 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.8 | 10 | 50 | T/P | | 5.08 | 26.38 | 24.06 | | 0.1116 | | 0.0964 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.4 | 10 | 50 | T/P | | 4.45 | 24.97 | 22.41 | | 0.0263 | | 0.1142 | |
| 120 | 30 | 30 | 0.01 | 1 | 2 | 8 | 50 | T/P | | 5.2 | 24.66 | 22.41 | | 0.1379 | | 0.1004 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.9 | 10 | 10 | T/P | | 4.71 | 18.16 | 16.86 | | 0.0306 | | 0.0771 | |
| 120 | 30 | 30 | 0.01 | 1 | 1.5 | 15 | 50 | D/P | | 4.27 | 19.76 | 18.12 | | 0.0656 | | 0.0905 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.5 | 25 | 50 | T/P | | 4.78 | 17.23 | 15.86 | | 0.0460 | | 0.0864 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.5 | 20 | 50 | T/P | | 4.44 | 19.67 | 18.01 | | 0.0284 | | 0.0922 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.3 | 15 | 30 | T/P | | 4.73 | 25.51 | 23.58 | | 0.0350 | | 0.0818 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.5 | 20 | 50 | T/P | | 4.06 | 29.81 | 28.16 | | 0.1116 | L Avg | 0.0586 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.7 | 9 | 50 | T/P | | 5.18 | 23.02 | 21.23 | | 0.1335 | 4.6862 | 0.0843 | |
| 120 | 30 | 30 | 0.01 | 1 | 2.6 | 15 | 50 | T/P | | 4.89 | 26.03 | 24.01 | | 0.0700 | 0.0254 | 0.0841 | 0.0799 |
| 120 | 30 | 30 | 0.02 | 1 | 2 | 5 | 80 | T/P | | 3.69 | 18.62 | 18.01 | | 0.1926 | % error | 0.0339 | |
| 120 | 30 | 30 | 0.02 | 1 | 3 | 10 | 80 | T/P | | 4.44 | 18.76 | 20.35 | | 0.0284 | | 0.0781 | |
| 120 | 30 | 30 | 0.02 | 1 | 3 | 10 | 50 | T/P | | 4.86 | 16.99 | 16.86 | | 0.0635 | | 0.0077 | |
| 120 | 30 | 30 | 0.02 | 1 | 2.1 | 40 | 80 | D/F | | 4.77 | 21.83 | 22.5 | | 0.0438 | | 0.0298 | |
| 120 | 30 | 30 | 0.02 | 1 | 2.5 | 8 | 50 | T/P | | 4.44 | 18.76 | 18.01 | | 0.0284 | | 0.0416 | |
| 120 | 30 | 30 | 0.02 | 1 | 2.9 | 6 | 50 | T/P | | 4.71 | 26.47 | 23.15 | | 0.0306 | L Avg | 0.1434 | |
| 120 | 30 | 30 | 0.02 | 1 | 2.5 | 6 | 50 | T/P | | 4.48 | 25.33 | 23.15 | | 0.0197 | 4.4013 | 0.0942 | |
| 120 | 30 | 30 | 0.02 | 1 | 3 | 20 | 50 | T/P | | 3.82 | 28.21 | 23.9 | | 0.1641 | 0.0369 | 0.1803 | 0.0761 |
| 120 | 30 | 30 | 0.03 | 1 | 1.1 | 15 | 80 | D/F | | 4.68 | 25.53 | 26.24 | | 0.0241 | % error | 0.0271 | |
| 120 | 30 | 30 | 0.03 | 1 | 4.7 | 30 | 80 | D/F | | 4.45 | 27.17 | 26.24 | | 0.0263 | | 0.0354 | |
| 120 | 30 | 30 | 0.03 | 1 | 2.6 | 20 | 140 | T/F | | 5.21 | 29.06 | 28.92 | | 0.1400 | | 0.0048 | |
| 120 | 30 | 30 | 0.03 | 1 | 2.5 | 35 | 80 | D/P | | 5.53 | 16.28 | 17 | | 0.2101 | | 0.0424 | |
| 120 | 30 | 30 | 0.03 | 1 | 6 | 35 | 80 | D/P | | 4.81 | 25.54 | 23.54 | | 0.0525 | | 0.0850 | |
| 120 | 30 | 30 | 0.03 | 1 | 7 | 60 | 80 | D/P | | 4.45 | 20.9 | 14.77 | | 0.0263 | | 0.4150 | |
| 120 | 30 | 30 | 0.03 | 1 | 2.7 | 20 | 100 | D/P | T1 | 4.74 | 22.49 | 32.3 | | 0.0372 | L Avg | 0.3037 | |
| 120 | 30 | 30 | 0.03 | 1 | 4 | 30 | 80 | T/P | L Avg | 5.12 | 13.49 | 14.77 | avg v | 0.1204 | 4.9456 | 0.0867 | |
| 60 | 30 | 30 | 0.03 | 1 | 2.8 | 25 | 10 | T/P | 4.69 | 5.52 | 36.58 | 28.26 | 1.219 | 0.2079 | 0.0822 | 0.2944 | 0.1438 |
| 60 | 30 | 30 | 0.03 | 2 | 3.2 | 15 | 150 | T/P | | 10.55 | 22.28 | 25.03 | 10.19 | 0.0353 | % error | 0.1099 | |
| 60 | 30 | 30 | 0.03 | 2 | 4.5 | 15 | 80 | D/F | | 10.91 | 26.68 | 25.48 | | 0.0707 | | 0.0471 | |
| 60 | 30 | 30 | 0.03 | 2 | 2.5 | 40 | 150 | T/P | | 11.25 | 14.59 | 14.45 | | 0.1040 | | 0.0097 | |
| 60 | 30 | 30 | 0.03 | 2 | 1.5 | 25 | 80 | D/F | | 11.45 | 20.38 | 19.08 | | 0.1237 | | 0.0681 | |
| 60 | 30 | 30 | 0.03 | 2 | 4.2 | 33 | 80 | T/P | | 8.91 | 18.18 | 19.08 | | 0.1256 | | 0.0472 | |
| 60 | 30 | 30 | 0.03 | 2 | 3.3 | 20 | 40 | T/P | | 9.91 | 21.74 | 21.51 | | 0.0275 | | 0.0107 | |
| 60 | 30 | 30 | 0.03 | 2 | 2.6 | 33 | 150 | T/F | | 11.61 | 17.65 | 18 | | 0.1394 | | 0.0194 | |
| 60 | 30 | 30 | 0.03 | 2 | 4.5 | 50 | 150 | T/F | | 11.24 | 22.1 | 22.22 | | 0.1030 | L Avg | 0.0054 | |
| 60 | 30 | 30 | 0.03 | 2 | 4.5 | 10 | 150 | T/F | | 9.92 | 17.29 | 16.78 | | 0.0265 | 10.6290 | 0.0304 | |
| 60 | 30 | 30 | 0.03 | 2 | 3.3 | 10 | 80 | T/F | | 10.54 | 22.31 | 22.22 | | 0.0343 | 0.0431 | 0.0041 | 0.0352 |
| 60 | 30 | 30 | 0.02 | 2 | 3.3 | 15 | 55 | T/F | | 10.81 | 21.29 | 22.22 | | 0.0608 | % error | 0.0419 | |
| 60 | 30 | 30 | 0.02 | 2 | 1 | 22 | 55 | D/P | | 9.63 | 14.86 | 15.38 | | 0.0550 | | 0.0338 | |
| 60 | 30 | 30 | 0.02 | 2 | 1 | 5 | 10 | D/P | | 10.34 | 14.77 | 15.38 | | 0.0147 | | 0.0397 | |
| 60 | 30 | 30 | 0.02 | 2 | 1.3 | 15 | 55 | D/F | | 10.04 | 17.16 | 19.95 | | 0.0147 | | 0.1398 | |
| 60 | 30 | 30 | 0.02 | 2 | 1.3 | 15 | 55 | D/F | | 10.87 | 17.39 | 16.86 | | 0.0667 | | 0.0314 | |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 60 | 30 | 30 | 0.02 | 2 | 1.3 | 15 | 55 | D/F | | 10.06 | 17.35 | 16.86 | | 0.0128 | | 0.0291 | |
| 60 | 30 | 30 | 0.02 | 2 | 2.4 | 15 | 55 | D/P | | 10.14 | 17.04 | 15.82 | | 0.0049 | | 0.0771 | |
| 60 | 30 | 30 | 0.02 | 2 | 4 | 25 | 0 | T/P | | 11.07 | 18.9 | 15.82 | | 0.0864 | | 0.1947 | |
| 60 | 30 | 30 | 0.02 | 2 | 3.5 | 15 | 15 | T/F | | 10.72 | 25.99 | 25.03 | | 0.0520 | | 0.0384 | |
| 60 | 30 | 30 | 0.02 | 2 | 1 | 15 | 55 | D/F | | 9.87 | 22.15 | 22.79 | | 0.0314 | L Avg | 0.0281 | |
| 60 | 30 | 30 | 0.02 | 2 | 3 | 15 | 55 | D/F | | 10.31 | 20.67 | 19.25 | | 0.0118 | 10.3325 | 0.0738 | |
| 60 | 30 | 30 | 0.02 | 2 | 1.5 | 15 | 55 | D/F | | 10.13 | 20.31 | 20.69 | | 0.0059 | 0.0140 | 0.0184 | 0.0622 |
| 60 | 30 | 30 | 0.01 | 2 | 1 | 15 | 55 | D/F | | 10.52 | 22.51 | 20.69 | | 0.0324 | % error | 0.0880 | |
| 60 | 30 | 30 | 0.01 | 2 | 2.3 | 10 | 30 | T/P | | 9.77 | 13.84 | 13.41 | | 0.0412 | | 0.0321 | |
| 60 | 30 | 30 | 0.01 | 2 | 2.5 | 30 | 0 | D/P | | 9.93 | 25.37 | 22.83 | | 0.0255 | | 0.1113 | |
| 60 | 30 | 30 | 0.01 | 2 | 1.5 | 11 | 55 | D/F | | 10.14 | 15.54 | 15.66 | | 0.0049 | | 0.0077 | |
| 60 | 30 | 30 | 0.01 | 2 | 2.1 | 22 | 55 | T/P | | 9.89 | 21.46 | 22.78 | | 0.0294 | | 0.0579 | |
| 60 | 30 | 30 | 0.01 | 2 | 3 | 30 | 0 | D/P | | 9.52 | 25.16 | 22.78 | | 0.0658 | | 0.1045 | |
| 60 | 30 | 30 | 0.01 | 2 | 1 | 15 | 0 | D/P | | 10.41 | 14.41 | 14.45 | | 0.0216 | | 0.0028 | |
| 60 | 30 | 30 | 0.01 | 2 | 1.5 | 11 | 55 | D/F | | 10.41 | 15.35 | 15.26 | | 0.0216 | | 0.0059 | |
| 60 | 30 | 30 | 0.01 | 2 | 2.6 | 11 | 55 | T/F | | 10.4 | 20.2 | 21.51 | | 0.0206 | | 0.0609 | |
| 60 | 30 | 30 | 0.01 | 2 | 2.4 | 11 | 55 | T/F | | 10.15 | 16.17 | 17.78 | | 0.0039 | | 0.0906 | 0.0561 |
| 120 | 30 | 30 | 0.01 | 2 | 3 | 30 | 15 | T/P | | 10.72 | 16.21 | 15.54 | | 0.0520 | | 0.0431 | |
| 120 | 30 | 30 | 0.01 | 2 | 3 | 10 | 0 | T/P | T2 | 9.65 | 25.84 | 24.28 | | 0.0530 | L Avg | 0.0643 | |
| 120 | 30 | 30 | 0.01 | 2 | 2.1 | 5 | 55 | T/P | L Avg | 9.46 | 23.53 | 23.88 | avg v | 0.0716 | 10.0179 | 0.0147 | |
| 120 | 30 | 30 | 0.01 | 2 | 3 | 30 | 0 | D/P | 10.3 | 9.28 | 15.24 | 14.46 | 0.186 | 0.0893 | 0.0169 | 0.0539 | 0.0527 |
| 60 | 30 | 30 | 0.03 | 3 | 5 | 15 | 15 | T/P | | 16.12 | 19.91 | 20.12 | 24.89 | 0.3524 | % error | 0.0104 | |
| 120 | 45 | 30 | 0.03 | 3 | 4 | 30 | 15 | T/P | | 17 | 20.45 | 20.12 | | 0.3170 | | 0.0164 | |
| 60 | 45 | 30 | 0.03 | 3 | 4 | 30 | 15 | T/P | | 26.19 | 20.94 | 20.12 | | 0.0522 | L Avg | 0.0408 | |
| 60 | 45 | 30 | 0.03 | 3 | 4 | 30 | 15 | T/P | | 24.29 | 11 | 10.92 | | 0.0241 | 21.2460 | 0.0073 | |
| 60 | 45 | 30 | 0.03 | 3 | 4 | 15 | 30 | T/P | | 22.63 | 14.57 | 13.42 | | 0.0908 | 0.1464 | 0.0857 | 0.0321 |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 15 | 15 | T/P | | 23.15 | 10.66 | 10.21 | | 0.0699 | % error | 0.0441 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 15 | 15 | T/P | | 27.58 | 23.3 | 19.85 | | 0.1081 | | 0.1738 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 15 | 30 | T/P | | 21.85 | 11.45 | 11.12 | | 0.1221 | | 0.0297 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 15 | 0 | T/P | | 23.07 | 15.26 | 13.79 | | 0.0731 | | 0.1066 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 15 | 55 | T/P | | 20.65 | 18.67 | 20.12 | | 0.1703 | | 0.0721 | |
| 60 | 45 | 30 | 0.02 | 3 | 4 | 10 | 50 | T/P | | 20.96 | 12.7 | 12.13 | | 0.1579 | | 0.0470 | |
| 60 | 45 | 30 | 0.02 | 3 | 4 | 10 | 0 | T/P | | 27.82 | 13.14 | 12.57 | | 0.1177 | | 0.0453 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 8 | 55 | T/P | | 25.26 | 13.66 | 12.57 | | 0.0149 | | 0.0867 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 8 | 55 | T/P | | 23.78 | 12.83 | 12.78 | | 0.0446 | L Avg | 0.0039 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 8 | 55 | T/P | | 23.66 | 19.87 | 12.78 | | 0.0494 | 23.4645 | 0.5548 | |
| 60 | 45 | 30 | 0.02 | 3 | 3 | 15 | 50 | T/P | | 20.33 | 20.75 | 19.18 | | 0.1832 | 0.0573 | 0.0819 | 0.1133 |
| 60 | 45 | 30 | 0.01 | 3 | 2 | 10 | 80 | T/P | | 21.58 | 20.93 | 19.18 | | 0.1330 | % error | 0.0912 | |
| 60 | 45 | 30 | 0.01 | 3 | 1 | 1 | 20 | T/P | | 22.23 | 11.5 | 11 | | 0.1069 | | 0.0455 | |
| 60 | 45 | 30 | 0.01 | 3 | 1.3 | 1 | 20 | T/P | | 22.01 | 20.87 | 18.63 | | 0.1157 | | 0.1202 | |
| 60 | 45 | 30 | 0.01 | 3 | 3 | 10 | 50 | T/P | | 26.85 | 17.63 | 16.49 | | 0.0787 | | 0.0691 | |
| 60 | 45 | 30 | 0.01 | 3 | 1.8 | 15 | 40 | D/P | | 23.5 | 19.47 | 18.95 | | 0.0558 | | 0.0274 | |
| 60 | 45 | 30 | 0.01 | 3 | 3 | 10 | 55 | T/P | | 23.91 | 10.13 | 10.12 | | 0.0394 | | 0.0010 | |
| 60 | 45 | 30 | 0.01 | 3 | 3 | 15 | 0 | T/P | | 22.32 | 10.53 | 10.12 | | 0.1033 | | 0.0405 | |
| 60 | 45 | 30 | 0.01 | 3 | 3 | 15 | 0 | T/P | | 23.88 | 20.9 | 18.55 | | 0.0406 | | 0.1267 | |
| 60 | 45 | 30 | 0.01 | 3 | 3 | 15 | 0 | T/P | T3 | 21.3 | 11.94 | 11.5 | | 0.1442 | L Avg | 0.0383 | |
| 60 | 45 | 30 | 0.01 | 3 | 3 | 15 | 0 | T/P | L Avg | 24.09 | 17.53 | 15.85 | avg v | 0.0321 | 23.1027 | 0.1060 | |
| 60 | 45 | 30 | 0.01 | 3 | 1.5 | 10 | 50 | D/P | 22.9 | 22.46 | 17.7 | 18.63 | 1.017 | 0.0976 | 0.0718 | 0.0499 | 0.0651 |
| | | | | | | | | | | | 19.81 | 19.0500 | | | % error | | |

| | | | | | |
|---|---|---|---|---|---|
| Average delta for HUMMWV | 0.1180 | m | 1.219 | m/s | 2.7306 miles/hour |
| Average delta for HEMTT | 0.1025 | m | 0.186 | m/s | 0.4169 miles/hour |
| Average delta for M1070 | 1.9837 | m | 1.017 | m/s | 2.2790 miles/hour |
| Total delta | 0.7347 | m | 0.808 | m/s | 1.8088 miles/hour |



## Table B.18 Sensor Data Threshold Detection

| Run Information | Threshold | N | γ | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SumPixel(Multi05_1(3:3,3:5,:),1,1); | | 2 | 1 | 3 | 85 | 105 | 3 | 3 | 85 | 105 | 5 | 1 | 85 | 105 |
| Multi05_1(3,4,:) | | 2 | 1 | 13 | 91 | 100 | 3 | 13 | 91 | 100 | 5 | 1 | 91 | 100 |
| SumPixel(Multi05_1(3:3,3:4,:),1,1); | | 2 | 1 | 4 | 85 | 100 | 3 | 4 | 85 | 100 | 5 | 1 | 85 | 100 |
| EnlargePixel(Multi05_1,3,4,.5,.5); | | 2 | 1 | 4 | 85 | 103 | 3 | 4 | 85 | 103 | 5 | 1 | 85 | 103 |
| TwoPix05_1(1,2,:); | | 2 | 1 | 2 | 83 | 96 | 3 | 2 | 83 | 96 | 5 | 1 | 83 | 96 |
| SumPixel(TwoPix05_1,1,1); | | 2 | 1 | 4 | 83 | 95 | 3 | 4 | 83 | 95 | 5 | 1 | 83 | 95 |
| OnePix05_2; | | 2 | 1 | 3 | 83 | 111 | 3 | 3 | 83 | 111 | 5 | 1 | 83 | 111 |
| EnlargePixel(Multi05_1,3,4,.5,1); | | 2 | 1 | 12 | 92 | 100 | 3 | 12 | 92 | 100 | 5 | 1 | 92 | 100 |
| EnlargePixel(Multi05_1,3,4,1,.5); | | 2 | 1 | 5 | 85 | 105 | 3 | 5 | 85 | 105 | 5 | 1 | 85 | 105 |
| OnePix96_2; | | 2 | 1 | 8 | 159 | 179 | 3 | 8 | 159 | 179 | 5 | 1 | 155 | 185 |

## Table B.19 Sensor Data Duration Detection

| Run Information | Duration | N | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SumPixel(Multi05_1(3:3,3:5,:),1,1); | | 1 | 17 | 85 | 106 | 3 | 17 | 85 | 106 | 2 | 17 | 85 | 106 |
| Multi05_1(3,4,:) | | 1 | 1 | 92 | 100 | 3 | 1 | 92 | 100 | 2 | 1 | 92 | 100 |
| SumPixel(Multi05_1(3:3,3:4,:),1,1); | | 1 | 1 | 85 | 100 | 3 | 1 | 85 | 100 | 2 | 1 | 85 | 100 |
| EnlargePixel(Multi05_1,3,4, 5, 5); | | 1 | 3 | 86 | 91 | 3 | 2 | 93 | 100 | 2 | 2 | 93 | 100 |
| TwoPix05_1(1,2,:); | | 1 | 1 | 83 | 95 | 3 | 1 | 83 | 95 | 2 | 1 | 83 | 95 |
| SumPixel(TwoPix05_1,1,1); | | 1 | 6 | 83 | 95 | 3 | 6 | 83 | 95 | 2 | 5 | 83 | 95 |
| OnePix05_2; | | 1 | 16 | 83 | 111 | 3 | 16 | 83 | 111 | 2 | 8 | 83 | 111 |
| EnlargePixel(Multi05_1,3,4, 5,1); | | 1 | 7 | 92 | 100 | 3 | 7 | 92 | 100 | 2 | 1 | 92 | 100 |
| EnlargePixel(Multi05_1,3,4,1, 5); | | 1 | 8 | 85 | 105 | 3 | 8 | 85 | 105 | 2 | 4 | 85 | 105 |
| OnePix96_2; | | 1 | 2 | 160 | 178 | 3 | 2 | 160 | 178 | 10 | 1 | 160 | 178 |

## Table B.20 Sensor Data FAPI Detection

| Run Information | FAPI | N | γ | #T | En1 | Ex2 | FAR1 | #T | En1 | Ex2 | FAR2 | #T | En1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SumPixel(Multi05_1(3:3,3:5,:),1,1); | | 2 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 5 | 0 | 0 | 0 |
| Multi05_1(3,4,:) | | 2 | 1 | 1 | 99 | 100 | 1 | 1 | 99 | 100 | 5 | 0 | 0 | 0 |
| SumPixel(Multi05_1(3:3,3:4,:),1,1); | | 2 | 1 | 1 | 2 | 210 | 1 | 1 | 2 | 210 | 5 | 1 | 2 | 210 |
| EnlargePixel(Multi05_1,3,4,.5, 5); | | 2 | 1 | 1 | 2 | 210 | 1 | 1 | 2 | 210 | 5 | 1 | 2 | 210 |
| TwoPix05_1(1,2,:); | | 2 | 1 | 1 | 90 | 99 | 1 | 1 | 90 | 99 | 5 | 1 | 90 | 99 |
| SumPixel(TwoPix05_1,1,1); | | 2 | 1 | 1 | 2 | 210 | 1 | 1 | 2 | 210 | 5 | 1 | 2 | 210 |
| OnePix05_2; | | 2 | 1 | 1 | 2 | 210 | 1 | 1 | 2 | 210 | 5 | 1 | 2 | 210 |
| EnlargePixel(Multi05_1,3,4, 5,1); | | 2 | 1 | 1 | 2 | 210 | 1 | 1 | 2 | 210 | 5 | 1 | 2 | 210 |
| EnlargePixel(Multi05_1,3,4,1, 5); | | 2 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 5 | 0 | 0 | 0 |
| OnePix96_2; | | 2 | 1 | 1 | 2 | 486 | 1 | 1 | 2 | 486 | 5 | 1 | 2 | 486 |

## Table B.21 Sensor Data PolyFit and FermiFit Analysis Data

| Run Information | Poly | En1 | En2 | Ex1 | Ex2 | Fermi | En1 | En2 | Ex1 | Ex2 | Real | En1 | En2 | Ex1 | Ex2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SumPixel(Multi05_1(3:3,3:5,:),1,1); | | 84.4 | 86.8 | 103 | 105 | | 85 | 91.3 | 98.7 | 105 | | 83 | 86 | 103 | 107 |
| Multi05_1(3,4,:) | | 85.5 | 47.8 | 108 | 107 | | 91 | 93.1 | 97.9 | 100 | | | | | |
| SumPixel(Multi05_1(3:3,3:4,:),1,1); | | 72.7 | 67.7 | 136 | 113 | | 85 | 91.3 | 98.7 | 105 | | 84 | 87 | 97 | |
| EnlargePixel(Multi05_1,3,4, 5, 5); | | 84.7 | 87.3 | 101 | 104 | | 85 | 89.7 | 102 | 108 | | 84 | 87 | 103 | 108 |
| TwoPix05_1(1,2,:); | | 73.2 | 68.4 | 2964 | 116 | | 83 | 87.1 | 91.9 | 96 | | 81 | 86 | 93 | 96 |
| SumPixel(TwoPix05_1,1,1); | | 68.9 | 63.9 | -324 | 142 | | 83 | 86.7 | 91.3 | 95 | | 61 | 71 | 93 | 96 |
| OnePix05_1; | | 61.0 | 66.4 | 73.6 | 83.0 | | 62 | 70.0 | 77.0 | 85 | | 62 | 67 | 80 | 85 |
| OnePix05_2; | | 76.4 | 85.8 | 109 | 114 | | 80 | 85.3 | 106 | 111 | | 81 | 86 | 109 | 115 |
| EnlargePixel(Multi05_1,3,4, 5,1); | | 86.5 | 60.5 | 118 | 111 | | 92 | 93.9 | 98.1 | 100 | | 87 | 94 | 97 | 101 |
| EnlargePixel(Multi05_1,3,4,1, 5); | | 85.4 | 91.8 | 85.9 | 102 | | 85 | 89.9 | 100 | 105 | | 84 | 87 | 103 | 108 |
| OnePix96_2; | | 157 | 165 | 171 | 183 | | 155 | 167 | 173 | 185 | | 156 | 162 | 175 | 180 |

# REFERENCES


[1] Kostka, D., "Moving Toward a Joint Acquisition Process to Support ISR," *Joint Force Quarterly* National Defense University Press, 2009 [Online] Available: http://www.ndu.edu/press/lib/ images/jfq-55/11.pdf.

[2] "Fundamentals of Remote Sensing Satellites and Sensors," *Natural Resources Canada,* Nov. 17, 2009 [Online] Available: Sep. 28, 2010, www.ccrs.nrcan.gc.ca/ resource/tutor/fundam/chapter2/02_e.php.

[3] "Border Patrol Overview," *US Customs and Border Protection* [Online]. Available: http://www.cbp.gov/xp/cgov/border_security/border_patrol/border_ patrol_ohs/overview.xml.

[4] "Transit Zone Interdiction Operations," *Office of National Drug Control Policy,* Oct. 16, 2009 [Online] Available: http://www.whitehousedrugpolicy.gov/ publications/ international/factsht/transit_zone_interdic_op.html.

[5] Maher, H., "Analysts Say U.S. Intelligence System Overloaded, Out of Data," *Radio Free Europe,* 2010, [Online] Available: http://www.rferl.org/content/ Analysts_Say_US_Intelligence_System_Overloaded_Out_Of_Date/1930418.html.

[6] Shachtman, N., "Robert Gates: Overhaul the Pentagon," *Wired News,* Sep. 21, 2009, [Online] Available: http://www.wired.com/techbiz/people/magazine/17-10/ ff_smartlist_gates?currentPage=all.

[7] Zhang, F., Li, C. and Shi, L., "Detecting and Tracking Dim Moving Point Target in IR Image Sequence," *Infrared Phys Tech,* vol. 46, pp. 323-328, Jul. 2004.

[8] Barniv, Y and Kella, O, "Dynamic Programming Solution for Detecting Dim Moving Targets PART II: Analysis," *IEEE T Aero Elec Sys,* vol. 6, pp. 776-788, Nov. 1987.

[9] Blostein, S. D. and Huang, T. S., "Detecting Small, Moving Objects in Image Sequences Using Sequential Hypothesis Testing," *IEEE T on Signal Process,* vol. 19, no. 7, pp. 1611-1629, 1991.

[10] Wei, W. and Jiaxiong, P., "Detection of Moving Small Targets in Infrared Image Sequences Containing Clour Clutter," *Intl J Pattern Recogn and AI,* vol. 18, no. 2, pp. 247-260, 2004.

[11] Sun, Y. Q., Tian, J. W. and Liu, J., "Background Suppression Based on Wavelet Transformation to Detect Infrared Target," *Proc Fourth Int Conf Mach Learning Cyb,* pp. 4611-4615, Aug 2005.

[12] Mei, X., Zhou, S. and Wu, H., "Integrated Detection, Tracking and Recognition for IR Video-Based Vehicle Classification," *IEEE ICASSP,* pp. 745-748, 2006.

[13] Simonson, K. and Ma, T., *Robust Real-Time Change Detection in High Jitter,* Albuquerque: Sandia National Laboratories, 2009.

[14] Badeau, R., David, B. and Richard, G., "Fast Approximated Power Iteration Subspace Tracking," *IEEE T on Sig Process,* vol. 53, no. 8, pp. 2931-2946, Aug 2005.

[15] Caefer, C. E., Mooney, J. M. and Silverman, J., "Point Target Detection in Consecutive Frame IR Imagery With Evolving Cloud Clutter," *Proc SPIE,* vol. 2561, pp. 14-24, 1995.

[16] Tzannes, A. P. and Mooney, J. M., *Point Target Detection in IR Image Sequences Using Spatio-Temporal Hypothesis Testing,* Hanscom AFB, MA: Air Force Research Laboratory, February 1999.

[17] Liu, D., Zhang, J. and Dong, W., "Temporal Profile Based Small Moving Target Detection Algorithm in Infrared Image Sequences," *Int J Infrared Milli,* vol. 28, pp. 272-281, Mar. 2007.

[18] Lim, E. T., Shue, L. and Venkarteswarlu, R., "Adaptive Mean and Variance Filter for Detection of Dim Point-Like Targets," *Proc SPIE: Signal and Data Proc of Small Targets,* vol. 4728, pp. 492-502, 2002.

[19] Tzannes, A. P. and Brooks, D. H., "Detecting Small Moving Objects Using Temporal Hypothesis Testing," *IEEE T Aero Elec Sys* vol. 38, pp. 570-585, Apr. 2002.

[20] Tursun, D., Guiying, X. and Hamdulla, A., "A Particle Filter Based Algorithm for State Estimation of Dim Moving Point Target in IR Image Sequence," *Intel Info Tech App,* vol. 2, pp. 127-131, Dec. 2008.

[21] Zaveri, M. A., Merchant, S. N. and Desai, U. B., "Wavelet-Based Detection and Its Application to Tracking in an IR Sequence," *IEEE T Sys Man Cyb,* vol. 37, no. 6, pp. 1269-1286, Nov. 2007.

[22] Zhang, X. and Ding, X., "A Note on Continuous Wavelet Transform," *J Guangxi Acad Sc,* vol. 23, pp. 4-6, 2007.

[23] Gao, T. and Liu, L., "Moving Video Object Segmentation Based on Redundant Wavelet Transform," *IEEE IC Info Auto,* pp. 156-160, Jun. 2008.

[24] "Army Wheeled Vehicles," *US Army Factfiles* [Online] Available: http://www.army. mi/factfiles/equipment/wheeled.

[25] Simonson, K M and Ma, T J, *Robuts Real-Time Change Detection in High Jitter,* Albuquerque: Sandia National Laboratories, 2009.

[26] Ran, Q., Chi, Y. and Wang, Z., "Property and Removal of Jitter in Beijing-1 Small Satellite," *Int Arch Photo Remote Sens Spat Info,* vol. 37, no. B1, pp. 929-934, 2008.

[27] Reed, I. S., "Optical Moving Target Detection with 3D Matched Filtering," *IEEE T Signal Process* vol. 24, pp. 327-336, 1998.

[28] Brain, Y., "Dynamic Programming Solution for Detection of Dim Moving Targets," *IEEE T Aero Elec Sys.* vol. 21, pp. 144-156, 1985.

[29] Radke, R., et al., "Image Change Detection Algorithms: A Systematic Survey," *IEEE T Image Process,* vol. 14, no. 3, pp. 294-307, Mar 2005.

[30] Rosin, P., "Thresholding for Change Detection," *Comp Vis Image Und,* vol. 86, no. 2, pp. 79-95, May 2002.

[31] Rosin, P. and Ioannidis, E., "Evaluation of Global Image Thresholding for Change Detection," *Pattern Recogn Lett,* vol. 24, no. 14, pp. 2345-2356, Oct 2003.

[32] Poor, H., *An Introduction to Signal Detection and Estimation,* New York : Springer-Verlag, 2000.

[33] Forstner, W., "A Framework for Low Level Feature Extraction," *Lecture Notes in Computer Science,* Stockholm: Springer-Verlag, 1994, vol. 801, pp. 383-394.

[34] Haag, M. and Nagel, H., "Combination of Edge Element and Optical Flow Estimates for 3D Model Based Vehicle Tracking in Traffic Sequences," *Int J Comput Vision,* vol. 35, no. 3, pp. 295-319, 1999.

[35] Hinz, S., "Integrating Local and Global Features for Vehicle Detection in High Resolution Aerial Imagery," *Int Arch Photo Remote Sens Spat Info,* vol. 34, pp. 119-124, 2003.

[36] Hinz, S. and Baumgartner, A., "Vehicle Detection in Aerial Images Using Generic Features Grouping and Context," *Lecture Notes on Computer Science,* Munich : Springer Berlin / Heidelberg, 2001, vol. 2191, pp. 45-52.

[37] Meer, P., Jolion, J. and Rosenfeld, A., "A Fast Parallel Algorithm for blind Estimation of Noise Variance," *IEEE T Pattern Anal.* vol. 12, no. 2, pp. 216-223, 1990.

[38] Yang, B., "Projection Approximation Subspace Tracking," *IEEE T Signal Process,* vol. 44, no. 1, pp. 95-107, Jan 1995.

[39] Abed-Meraim, K., Chkeif, A. and Hua, Y., "Fast Orthonormal PAST Algorithm," *IEEE T Signal Process,* vol. 7, no. 3, pp. 60-62, Mar 2000.

[40] Miao, Y. and Hua, Y, "Fast Subspace Tracking and Neural Network Learning By a Novel Information Criterion," *IEEE T Signal Process,* vol. 46, no. 7, pp. 1967-1979, Jul 1998.

[41] Cavallaro, A. and Ebrahimi, T., "Video Object Extraction Based on Adaptive Background and Statistical Change Detection," *Proc SPIE Vis Image Process.* pp. 465-475, Jan 2001.

[42] Huwer, S. and Niemann, H., "Adaptive Change Detection for Real-Time Surveillance Applications," *Proc Visual Surv,* pp. 75-91, 2000.

[43] Kanade, T. and Collins, R., "Advances in Cooperative Multi-Sensor Video Surveillance," *Proc DARPA Image Understanding Workshop,* vol. 1, pp. 3-24, Nov 1998.

[44] Zelen, M and Severo, N, "Probability Functions ," *Hanbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables,* New York: National Bureau of Standards, 1964, pp. 936-964.

[45] "R2010b Documentation Matlab Hist," *MathWorks,* [Online] Available: http:// www.mathworks.com/help/techdoc/ref/hist.html.

[46] Badeau, R, David, B and Richard, G, Fast Approximated Power Iteration Subspace Tracking, *IEEE T Signal Process,* 2005, vol. 53, 6.

[47] Choi, W. C., et al., "Automatic Tracking System with Target Classification," *Proc SPIE,* vol. 7335, pp. 73350T1-73350T8, 2009.

[48] Nair, D. and Aggarwai, J. K., "Robust Automatic Target Recognition in Second Generation FLIR Images," *IEEE T Signal Process,* pp. 194-201, 1996.

[49] Manikandan, J., Venkataramani, B. and Jayachandran, M., "Evaluation of Edge Detection Techniques Towards Implementation of Automatic Target Recognition," *Int Conf on Comp Intel Multimedia Apps.* , pp. 441-445, 2007.

[50] Clarkson, T., "Automatic Target Recognition Using Neural Networks," *Neural Network World,* pp. 1-9, 1995.

[51] Gupte, S., et al., "Detection and Classification of Vehicles," *IEEE T Intel Trans Sys.* vol. 3, pp. 37-47, Mar. 2002.

[52] Hoffman, J D, "Polynomial Approximatin and Interpolation," *Numerical Methods for Engineers and Scientists,* New York: Marcel Dekker, 2001, pp. 228-231.

[53] Robinson, G., "Test Data," *RP Flight Systems,* Jul. 9, 2010 [Online] Available: ftp://ftp.rpflightsystems.com/.

[54] Melgaard, D and Larson, K, *Practical Considerations for a Velocity Matched Filter Applies to Dim Target Detection,* s.l.: Sandia National Lab Report, 2010.

[55] Boult, T., "Into the Woods: Visual Surveillance of Non-Cooperative and Camouflaged Targets in Complex Outdoor Settings,", *Proc IEEE*, vol. 89, pp. 1382-1402, Oct 2001.

[56] Bhuiyan, S. M., Alam, M. S. and Alkanhal, M., "New Two-Stage Correlation-Based Approach for Target Detection and Tracking in Forward-Looking Infrared Imagery Using Filters Based on Extended Maximum Average Correlation Height and Polynomial Distance Classifier Correlation," *Opt Eng,* vol. 46, no. 8, pp. 864011-8640114, Aug. 2007.

[57] Hinz, S. and Stilla, U., "Car Detection in Aerial Thermal Images by Local and Global Evidence Accumulation," *Pattern Recogn Lett,* vol. 27, pp. 308-315, 2006.

[58] Xue, M., Zhou, S. and Hao, W., "Integrated Detection, Tracking and Recognition for IR Video-Based Vehicle Classification," *IEEE T Acoust Speech.* vol. 5, pp. 745-748, Apr 2006.

[59] Gordon, N. J., Salmond, D. J. and Smith, A. F., "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation." *IEEE Proc,* vol. 140, no. 2, pp. 107-113, 1993.

[60] Doucet, A., Godsill, S. and Andrieu, C., "On Sequential Monte Carlo Sampling Methods for Bayesian filtering," *Stat Comp,* vol. 10, no. 3, pp. 197-208, 2000.

[61] Ling, J., Liu, E. Q. and Yang, J., "Approach of Infrared Small Target Mortion Prediction and Tracking Based on H Filter," *J Infrared Milli,* vol. 24, no. 5, pp. 366-369, 2005.

[62] Caefer, C. E., Silverman, J. and Mooney, J. M., "Optimization of Point Target Tracking Filters," *IEEE T Aero Elec Sys,* vol. 36, no. 1, pp. 496-507, 15-25, 2000.

[63] Caefer, C. E., Mooney, J. M. and Silverman, J., "Point Target Detection in Consecutive Frame IR Imagery with Evolving Cloud Clutter," *Proc SPIE,* vol. 2561, pp. 14-24, 1995.