5-2008

An Efficient Hardware Implementation of Target Recognition Algorithms and Investigation of Secure Wireless Communication for a Modified Manet

Stephen Barnes
University of Arkansas, Fayetteville

AN EFFICIENT HARDWARE IMPLEMENTATION OF TARGET RECOGNITION ALGORITHMS AND INVESTIGATION OF SECURE WIRELESS COMMUNICATION FOR A MODIFIED MANET

AN EFFICIENT HARDWARE IMPLEMENTATION OF TARGET RECOGNITION ALGORITHMS AND INVESTIGATION OF SECURE WIRELESS COMMUNICATION FOR A MODIFIED MANET

A thesis submitted in partial
fulfillment of the requirements for the degree of
Bachelor of Science

By

Stephen Barnes

ABSTRACT

This paper presents a scheme for effective wireless security of a open broadcast mobile ad-hoc network, MANET, network without significant loss of bandwidth and data integrity through a double tiered encryption scheme, and the feasibility of reducing the target tracking algorithm in [1] into a compact and efficient hardware package. Due to the open nature of MANET, modifications are necessary to secure wireless data in a potential hostile environment. Furthermore, due to power and processing limitations of small unmanned aerial vehicles (UAVs) and the processing intensive calculations of image processing, a sample hardware implementation of key functions of the target tracking algorithm is described. Using hardware simulation and modeling to implement key elements, results are compared against identical function blocks in a software environment. The results of this research allow for further work in open broadcast MANET security and target tracking hardware implementation to be confidently pursued; it also suggests the tools, methodology, and overall architecture for a larger project.

This thesis is approved for recommendation to the Graduate Council.

Thesis Director:

_____

Dr. Jia Di

Thesis Committee:

_____

Dr. Patrick Parkerson

_____

Dr. Haiying Shen

THESIS DUPLICATION RELEASE

I hereby authorize the University of Arkansas Libraries to duplicate this thesis when needed for research and/or scholarship.

Agreed _____

Refused _____

ACKNOWLEDGEMENTS

I thank Dr. Jia Di for his direction during my research and helpful revisions to my work, my thesis committee for volunteering their expertise to the betterment of my research, and both Carnagie Mellon and FastVDO for their work on this project.

I also thank Kyle White for his help during long hours of research and my family for their continual support.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 Problem

Effective wireless security without loss of bandwidth and data integrity is essential to operations in a potential military environment where location, image data, and overall battlefield awareness cannot be shared outside of the network. MANET, mobile ad-hoc networking, is a robust and commonly used scheme for connecting nodes without the need of a central server. However, by its very nature, MANET is vunerable to intrusion by malicious nodes. Additionally, a MANET that is designed to be broadcast oriented needs each message to be efficient as data is repeated multiple times across the network. Most work on determining the threat of nodes is based upon specific identification nodes [2] [3] [4]. However, an open broadcast MANET required a different approach to network security.

When dealing with unmanned vehicles, typically power and processing speed are not as readily available as it is when developing in other environments. Unmanned aerial vehicles, UAVs, have many jobs to consider such as navigation, communication, and physical hardware management. Secondary tasks, such as sensing and computer vision, must be as efficient as possible so that primary tasks have adequate power and processing time. This thesis deals mainly with the problem of minimizing the costs of the target tracking algorithm in [1] and showing the feasibility of implementing it within hardware. The computational intensity of visual target tracking necessitates that this algorithm is reduced.

1.2 Objective

A broadcast-oriented MANET may be made secure through a combination of unique trust identifiers, encryption, and communication organization. Futhermore, this thesis will show that it is feasible to implement the target tracking algorithm [1] in hardware to provide a low power, fast alternative to software emulation.

1.3 Approach

The wireless portion of this paper approaches the problem of data security through a combination of synchronous and asynchronous encryption. In addition, a unique sequence for proving trust and specific broadcast oriented MANET issues are discussed. Finally, particular encryption schemes are suggested based upon hardware encryption and decryption speeds.

The target tracking and ATR section of this paper is approached through rapid hardware design and simulation. Using tools such as MATLAB, Simulink, and ModelSim, key components are simulated using predefined data sets. Additionally, analysis of data integrity is also included.

1.4 Potential Impact

Successful implementation of both wireless security and hardware implementation of key target tracking and ATR components could provide important progress in UAV development. Reliable image results from automated vehicles with automated tracking could improve response times to critical events such as search and rescue or natural disaster missions. Furthermore, robust wireless security could allow for

for UAV's to be deployed, without risk from electronic interference, into almost any environment.

1.5 Organization of this Thesis

This paper is organized as follows. In Section 2, the background and key issues are described; this includes the specifications of an open broadcast MANET, public and private key encryption, and the Yue and Chellappa [1] target tracking and automatic target recognition algorithm. Section 3 contains the high level overview of the wireless security scheme and the overall description of the implementation of the target tracking algorithm. In Section 4, the implementation details and suggestions are described, while in Section 5 the implementation details are analyzed. Finally, Section 6 contains concluding remarks and suggestions for future work.

2. BACKGROUND

2.1 Key Concepts

2.1.1 Open Broadcast MANET and Public/Private Key Encryption

Mobile ad-hoc networking, MANET, is a widespread protocol for networking various nodes without the addition of a central server. VANET, vehicular ad-hoc networking, is a similar concept that focuses on ground based networks between vehicles. Due to the high flexibility and speed of movement of UAVs, MANET is a far better choice.

Security for MANET and VANET is an issue that has been investigated by several sources, but none have developed a unique scheme for dealing with an open broadcast system. In this system, every message from a node is broadcast to all nodes that are within range, which is then forwarded again by each node that receives it up to a certain number of times. This concept increases fault tolerance at the cost of communication bandwidth. One node in the swarm cannot directly communicate solely with another node within the swarm.

Public and private key encryption is introduced in Chapters 3 and 4 as part of the overall security scheme. Due to the nature of each, private key encryption is also known of synchronous encryption while public key encryption is also known as asynchronous encryption. Private key encryption establishes key pairs between individuals nodes while public key encryption contains a large public key, with local smaller, private keys to decrypt.

2.1.1.1 Encryption and Decryption Speeds for Synchronous/Asynchronous Encryption

Synchronous encryption is generally faster than asynchronous encryption due to the nature of the mathematical problems they use and required key lengths. Sample times of hardware implementations of both types of encryption can be seen below in Table 1.

Speed Comparisons for Various Encryption Techniques					
Name	**Type**	**Frequency**	**Encryption Speed**	**Decryption Speed**	**Reference**
AES	Symmetric	58 MHz	390 Mbps	390 Mbps	[5]
RSA	Asymmetric	28 MHz	153 Kbps	17 Kbps	[5]
ECC	Asymmetric	50 MHz	20 Kbps	20 Kbps	[5]
ECC	Asymmetric	-	6.5×10^{-4} bits/ clock cycle	13.1×10^{-4} bits/ clock cycle	[6]

Table 1: An Overview of Private and Public Hardware Encryption Speeds

2.1.2 Visual Target Tracking and Automated Target Recognition

Visual targest tracking and automated target recognition, ATR, is a form of computer vision for automated vehicles. The process is as follows. First, the initial video frame, or image, is analyzed by the ATR recognition algorithm and one or more areas of interest, AOI, is formed. Follow this, the target tracking algorithm takes over and analyzes each following frame to track the progress of the central form within the AOI. The target tracking and ATR algorithms considered within this paper are discussed in [1] and [7].

2.2 Literature Review

2.2.1 Wireless Security

Work in the field of encryption and trust authentication is readily available. Particularly, work on Elliptical Curve Cryptography, which is discussed in Section 4.1.1,

can be viewed in [8], [6], and [9]. Furthermore, RSA and AES encryptions are discussed in [5].

Several topics discussed within Section 3.1 and 4.1, such as trust, certificate tables, and swarm intelligence models are discussed in [10].

2.2.2 Target Tracking and Automatic Target Recognition

This thesis is based upon the work covered in [1] and [7] . The work presented within [7] covers visual tracking and recognition using appearance-adaptive models within particle filters. This work is applicable to many areas, including vehicle tracking, aerial tracking, and facial recognition. The work presented in [1] summarizes visual tracking and novel views for UAVs. Specifically, the work presented on image fusion is valuable in a swarm environment for UAVs.

Topics discussed within Sections 3.2 and 4.2, such as visual tracking, image gallieries, and particle filters are also covered in [7].

3. ARCHITECTURE

3.1 Overview: Wireless Security

In order to provide data encryption while also sustaining a low encryption and decryption time cost, this thesis proposes the use of asymmetric key encryption for trust authentication and symmetric key encryption for data transfer after trust has been established. The use of a proprietary, low level hardware key may be needed to ensure that the MANET stay closed to the UAV system when authenticating using the asymmetric key encryption. Because of the limitations of a broadcast system (typical private key encryption cannot be used because nodes are not allowed to directly communicate between one another), modifications must be made to the private key encryption scheme. Figure 1 provides an overview of generating, maintaining, and managing keys between nodes in an open, broadcast oriented MANET.

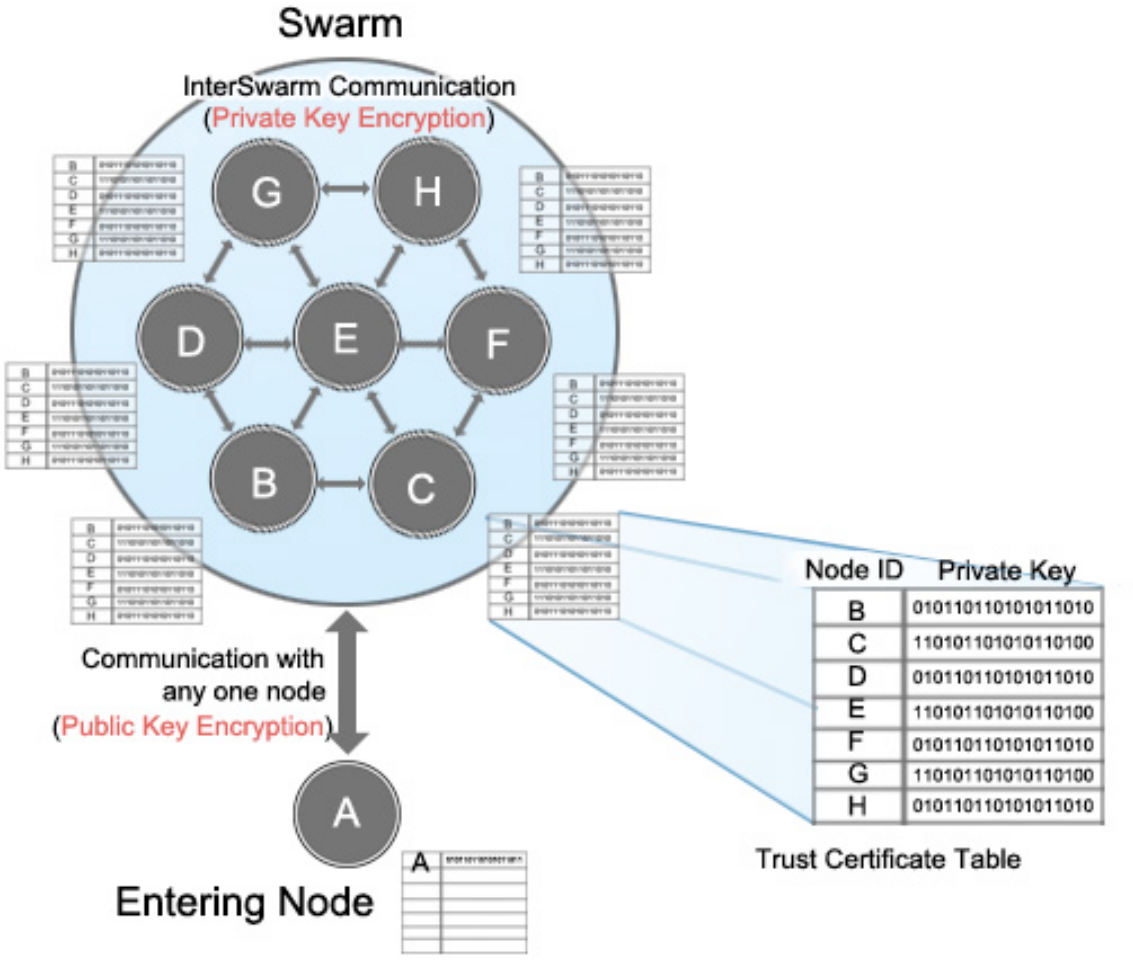


Figure 1: Overview of the proposed wireless security scheme

3.2 Overview: Target Tracking Simulation and Feasability

In order to successfully show the feasibility of the hardware implementation for the target tracking algorithm, two key blocks were designed and simulated for use with an FPGA or ASIC. In order to prove feasibility without rigorous development, software simulation of hardware designs was exploited using MATLAB, Simulink, and ModelSim. A design flow has been developed and is shown in Figure 6, and the detailed steps and results for each phase of the design flow can be seen below.

4. IMPLEMENTATION

4.1 Wireless Security

In Section 4.1, details are discussed for a future implementation of the proposed wireless security scheme. The overall organizational management can be seen in Figure 2.

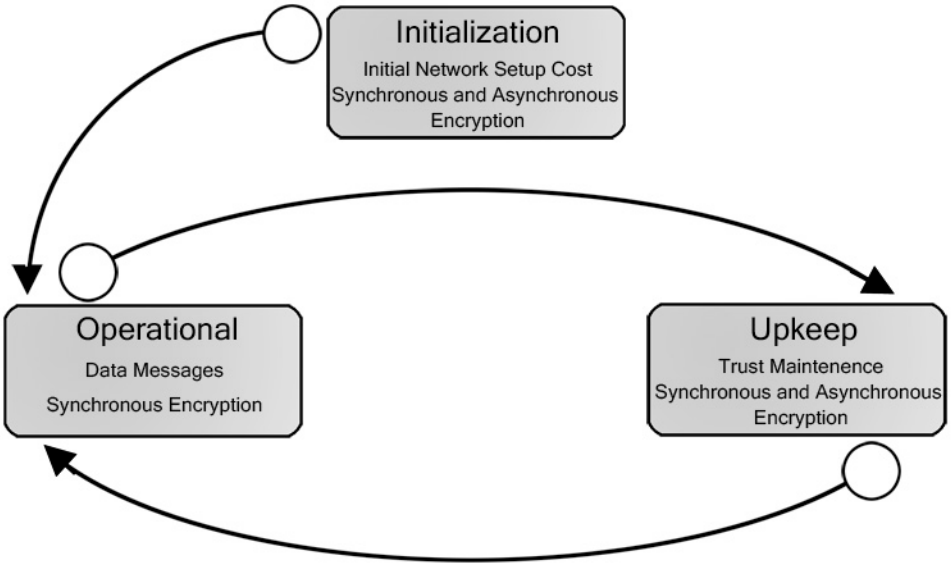


Figure 2: Overview of the proposed wireless security scheme

4.1.1 Asynchronous Encryption and Trust

This thesis proposes the use of either ECC or RSA public key encryption for the public key encryption scheme, with an emphasis on the former. Both provide excellent security that is resilient against a brute force attacks [9]. ECC, Elliptic Curve Cryptography, uses the lliptic curve discrete logarithm problem as opposed to the integer factorization problem used by RSA [9]. Dr. Vanstone sums up ECC's prime advantage

over other public key encryption schemes in his paper “Next generation security for wireless: elliptic curve cryptography”[9]:

> *“Because the best known way to solve the elliptic curve discrete logarithm problem (ECDLP) is fully exponential, you can use substantially smaller key sizes to obtain equivalent strengths. Hence ECC provides the most security per bit of any public key scheme known.”*

This allows ECC to use much smaller key sizes and therefore it is smaller, more energy efficient, more bandwidth conservative, and faster.

In order to provide proof of trust, a UAV entering the swarm must offer a certain code or portion of information that has been been encrypted by public key encryption. This code must also be subject to variances so that outside sources cannot duplicate the communication bit by bit and receive a response from the swarm. A case or time sensitive lookup table could be used to produce this proprietary trust indentification value.

4.1.2 Synchronous Encryption and Data Transfer

For the synchronous encryption and data transfer, research suggests a hardware implementation of AES (Advanced Encryption Standard) symmetric key encryption. AES, also known as Rijndael after its creators, is the current encryption standard for the U.S. government and the successor to DES, Data Encryption Standard [11]. A recommended key length for AES to be protected until the year 2036 is only 128 bits, and should be matched by a 256 bit ECC key or a 3248 bit RSA key [11].

Private key encryption is typically performed by matching one key between a pair of nodes. However, in a broadcast system it is impossible to transmit directly between two nodes. As a result, a copy of the correct private key decryption key must be kept by

each node in the swarm. When transmitting data, a node must first identify itself before sending data. An example of the synchronous encryption and data transfer scheme can be seen in Figure 4.

4.1.3 Implementation and Performance

The method of network encryption and security suggested is kept as simple as possible to eliminate processing costs, transmission quantities, and transmission data sizes, as all are limited. Briefly, a setup cost of establishing trust between nodes is required, and it may be required on certain intervals to reestablish trust between nodes. A description of the process is shown in Figure 3 and is discussed in section 4.1.1. After trust is established between two nodes the communication methodology can switch to a symmetric encryption style as shown in Figure 4. In both methods, a hello message or identification header which contains the node's unique identification is broadcast from the sender node and received by every other node within range. As the recipient node receives this, it checks in its certificate table to establish if the node is already known. If not, the recipient replies with its public key and identification to establish a trustable connection. Alternatively, if the node is known, the recipient merely has to match the identity with the proper private encryption key to decrypt the data that follows. The encryption and decryption speed differences between public and private key encryption is shown above, in Table 1.

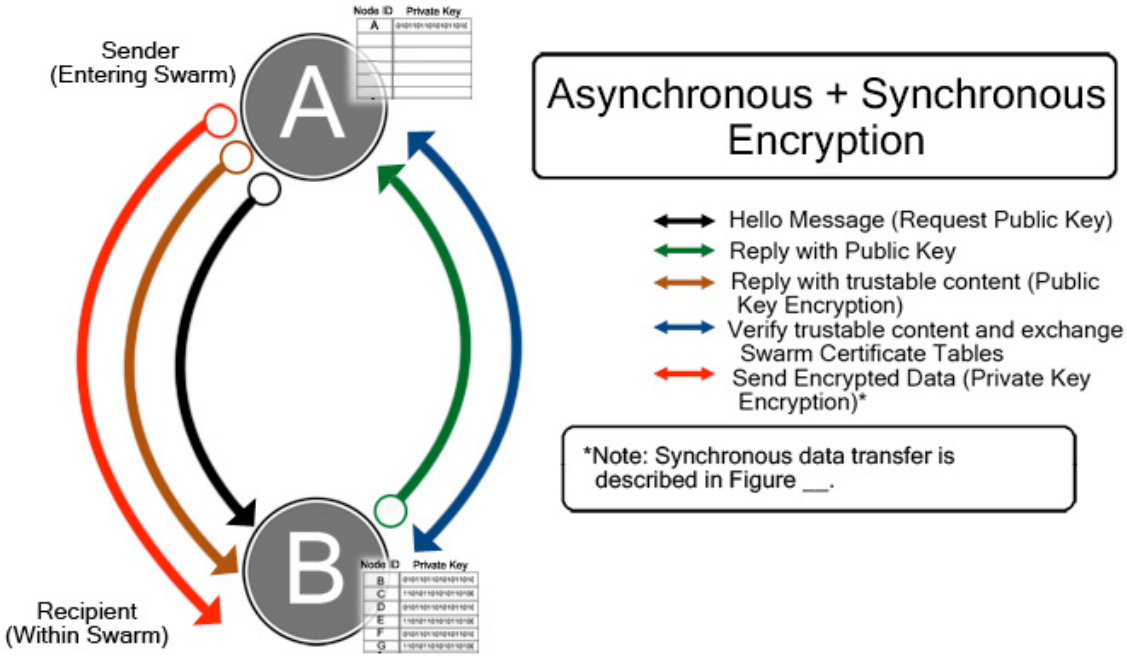


Figure 3: State Diagram for Asynchronous + Synchronous Encryption.

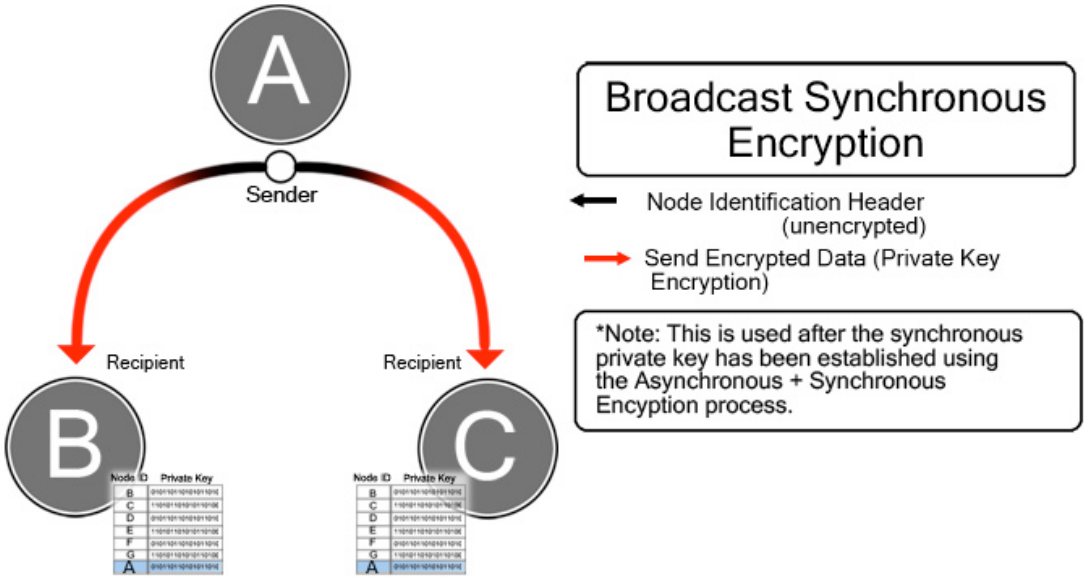


Figure 4: State Diagram for Broadcast Synchronous Encryption

4.2 Target Tracking Simulation and Feasability

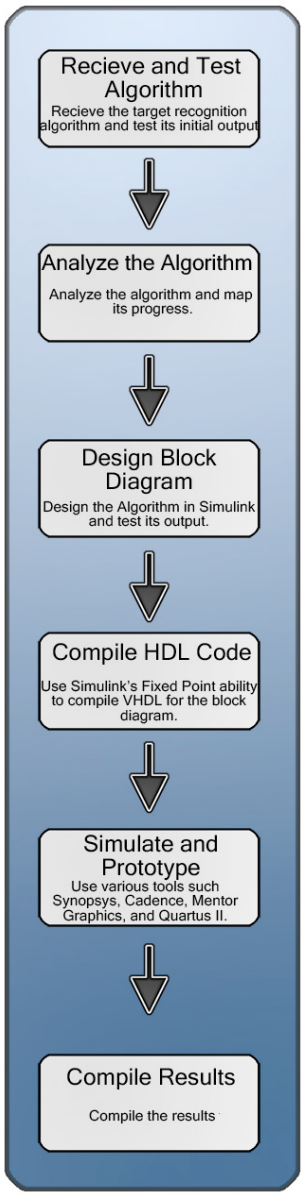


Figure 5: Hardware

4.2.1 Receive and Test Algorithm

The target tracking and recognition algorithm, based off [1] and [7], was written in the MATLAB environment, and was tested successfully with the provided sample image sequence and input values. The MATLAB programming environment allows for a rapid interface with MathWork's Simulink, a tool with which allows an accelerated design flow for transitioning from algorithm design to hardware design. A sample image output from the received algorithm is shown in Figure 5.

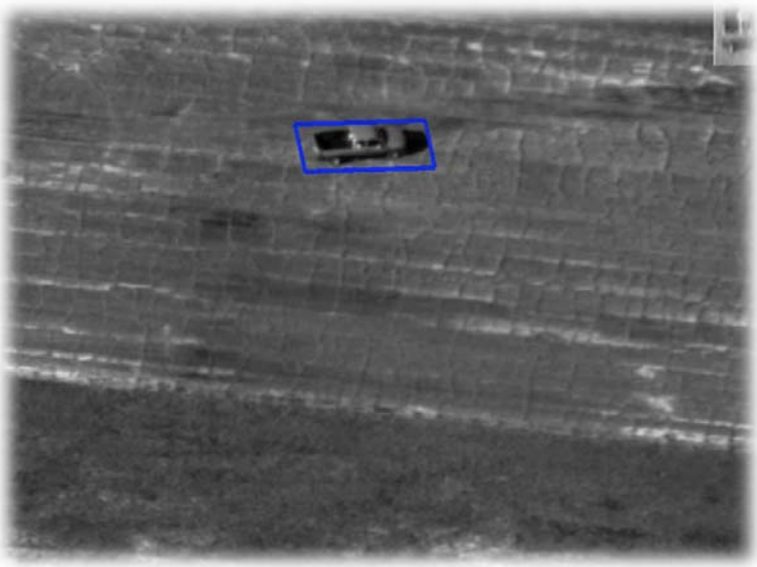


Figure 6: Example output from the target tracking algorithm

4.2.2 Analysis of Algorithm

Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters [7] differs from many image processing analysis techniques by focusing more on data computation as opposed to image processing filters. Due to the large size of data that could be calculated within each major function of the algorithm, sometimes matrices up to 1500 by 720 containing double values, it is important to analyze which functions should be studied and implemented first. Another consideration was the generation of random numbers within certain functions which would overcomplicate the goal of showing feasability.

Particularly, the focus was on demonstrating the feasibility of the target tracking portion of the algorithm. Two blocks were selected as appropriate for demonstrating reasonable gains through optimizing the hardware implementation. The first function selected is used on each frame up to five times to select the values for each particle sample from the current image frame using affine computation. The second function forms a new image gallery and fills it with zero-mean-unit-variance samples. Each function is used primarily when estimating the mean shift in affine transformation between two adjacent frames, a critical part of the target tracking algorithm.

4.2.3 Design Block Diagram

Implementation of the target blocks in Simulink was accomplished with the Fixed-Point Toolbox, Signal Processing Blockset, Image Processing Blockset, and primarily with Embedded MATLAB code for HDL compilation. Embedded MATLAB, a subset of the native MATLAB programming language for embedded C

programming, has a further subset that allows for HDL code generation. The benefits of Embedded MATLAB, otherwise known as EML, encompass faster development time, easier design of sequential logic within hardware, and an easier learning curve for algorithm design architects. In order to properly simulate hardware, input data to each EML block is streamed in serially, and then stored into memory when using persistent variables. After the data has been allocated to the block, more complex operations such as matrix multiplication, reshaping, and other high level functions are available.

Full implementation of creating zero-mean-variance image galleries was constructed and a partial implementation of image warping using affine computation was also constructed. These implementations could then automatically generate VHDL code for hardware simulation. Prior to VHDL generation, a simulation may also be run and checked for proper output with the benefit of stepping line by line through design files. The top and mid level diagram from Simulink for the zero-mean-variance image gallery creation is shown below in Figure 7.

Accurate hardware level operations are modeled by Simulink's Fixed-Point data type. With the fixed-point data type, properties such as bit length, fraction length, product accuracy mode, rounding mode and many others can be specified. In practice, fixed point makes excellent use of optimization, and using the correct settings error percentages for real world output versus simulation can be near 1/1000. The mean difference for result differences between real world values and simulation values for both blocks is shown in Figures 8a and 8b.

Each block was implemented using a finite state machine to control different sequential functions. Primarily, each block had three states: data retrieval, data processing, and data output. During the data retrieval stage the block accepts serial streamed input data, while in the data output stage the block outputs serial streamed results. In both the data retrieval and output states, each block stores data into vectors of EML persistent variables, which act as memory, or delays, within hardware.

Within the data processing state of the image gallery creation block, several calculations have to be performed. First, the serially streamed 180000 bits of data must be reshaped into a matrix of 720 x 250 bits. This matrix contains template data, the 30x24 pixel area of interest, for 250 different samples. After the data is in a more easily modified form a mean and standard deviation is calculated on each template sample, data manipulation is performed on the input using both of these factors, and the output is formed into an outgoing serial stream. The processing state of the image cropping block is currently implemented to a point of reshaping several input streams, multiplying a 1500 by 720 type double matrix, and manipulating this data to analyze which pixel should or should not be cropped.

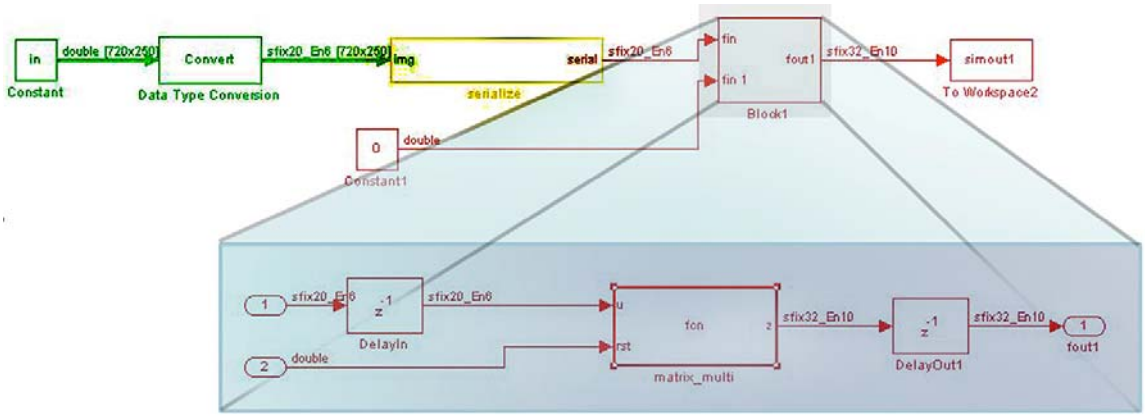


Figure 7: Simulink model for zero-mean-variance galleries

4.2.4 Compile HDL Code

Compilation of Simulink model files into valid VHDL code for hardware simulation and implementation is automatically generated by Simulink's HDL Coder. Also, a test bench for the VHDL code can be automatically generated from the data samples and sample times of the model.

4.2.5 Simulate and Prototype

Simulation and prototyping results are detailed in Section 5.1 and 5.2.

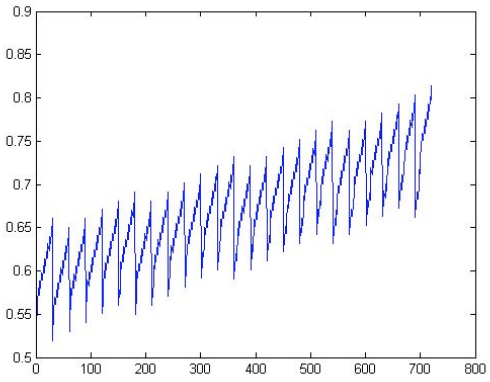
4.2.6 Compile Results

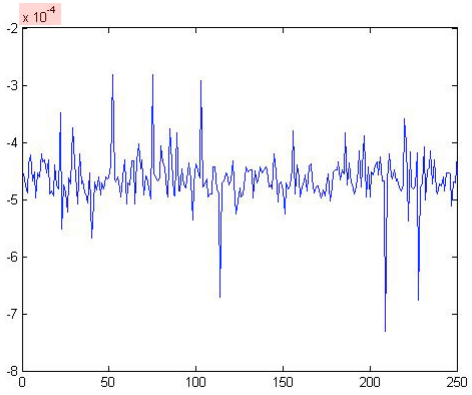
Data comparison and results analysis are detailed in Section 5.1 and 5.3.

5. ANALYSIS AND TESTING

5.1 Hardware Simulation via Simulink

Simulations were performed on both models during the block design process. Sample data for both the Simulink and ModelSim simulations was obtained by running the received algorithm in the MATLAB environment and saving input, output, and intermediate data samples. The largest input data sample for creating image galleries is a 720 by 250 type double input matrix, while the affine computation block accepts three different serial inputs ranging from a 1500 by 3 type double input matrix to the serialized values of the grayscale transformation of the current image frame. Results from the Simulink simulations can be viewed in Figures 9a and 9b.



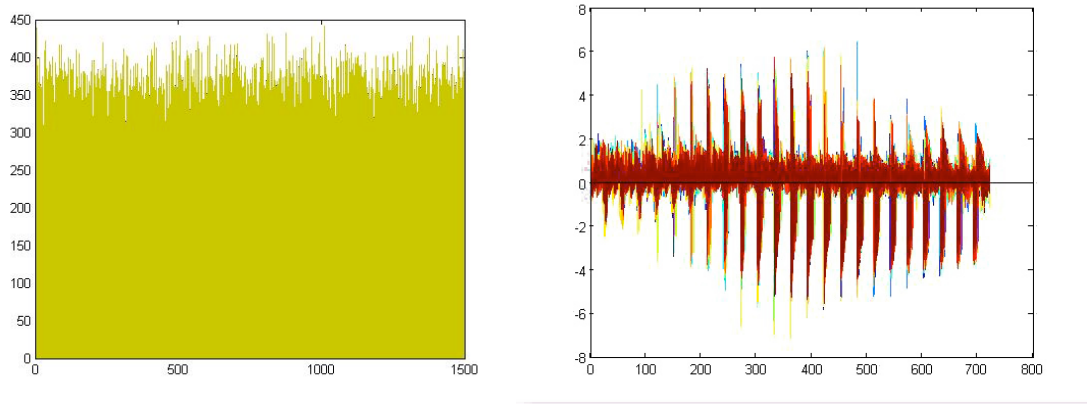


Figures 8a(left) and 8b(right): Data error comparisons

$$\text{Equation: } \frac{\sum_{i=1}^{N}(Simulated\ Output - Real\ Output)}{N}$$

Figure 8a: Mean Fixed Point data error for current affine transformation variable * grid with a combined matrix size of 1500 x 720 type double
Figure 8b: Mean Fixed Point data error for the zero-mean-variance image gallery output with 250 samples

Min Left: 1.000 Min Right: -7.1005
Max Left: 442.6013 Max Right: 6.4238



Figures 9a(left) and 9b(right): Output data samples
Figure 9a: Data output for current affine transformation variable * grid with a combined matrix size of 1500 x 720 type double
Figure 9b: Data output for the zero-mean-variance image gallery output with 250 samples

5.2 Hardware Simulation via Modelsim

Pending VHDL code and test bench generation, the design is tested in Mentor Graphic's ModelSim simulation framework. The testbench generated by such that the clock is forced to five nanoseconds (200MHz) and the hold time is two nanoseconds. The final simulation provides a nanoscale view of each operation as it occurs, as well as validation of the generated VHDL code. Example output from the testbench compilation and simulation within ModelSim is shown in Figure 10.

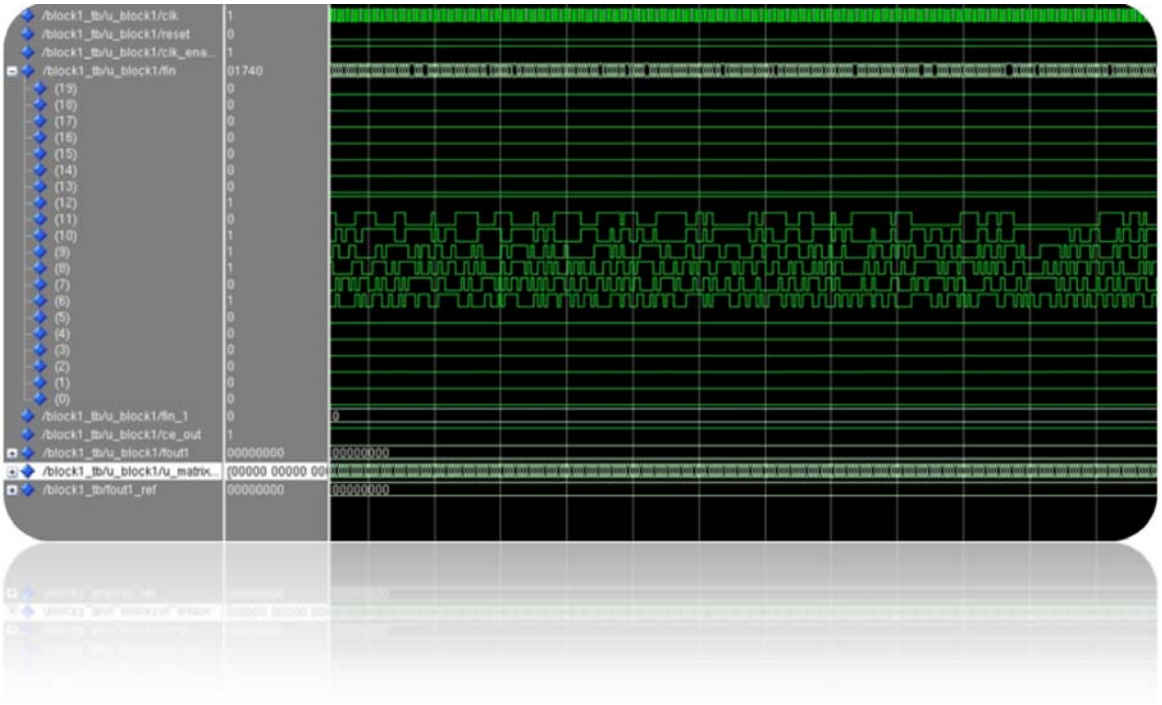


Figure 10: Sample output from the simulated testbench in ModelSim

6. CONCLUSIONS

6.1 Summary

MANET provides an excellent framework for swarm networking, as it allows for nodes to enter and exit without penalty. Similarly, a broadcast oriented MANET creates a fault-tolerant system that improves communication integrity. However, to properly secure a broadcast oriented MANET, the authentication scheme above should be considered.

A full implementation of the target tracking algorithm [1] could be rapidly achieved through the design process described above. Furthermore, this thesis proposes that it is feasible to implement this algorithm using a competitive time schedule and that data integrity would not notably suffer.

6.2 Contributions

This thesis contributes to the following areas:

- Image Processing and Computer Vision
- Target Tracking and ATR
- Wireless Security for Broadcast Oriented MANET networks

6.3 Future Work

6.3.1 Wireless Security

The proposed data encryption plan and fault tolerant design help ensure data concealment, but two problems still exist: resourse attacks, and malicious node intrusion [6]. Resourse attacks are a serious threat that can easily jam a MANET with thousands of

faulty requests for authentication. This preys on the MANET's slowest member, the public key encryption scheme for establishing trust. Malicious node intrusion, on the other hand, is due to the fact that our network has no established structure and allows nodes to enter and exit the network freely. This allows a malicious node to request authentication and trust as it assumes the role of a proper node. If the malicious node is successful in authenticating with just one node, it could possibly receive data or send conflicting messages throughout the network.

6.3.2 Target Tracking

Performance of the current implementation is hampered by the process of serially streaming data from a source, storing it within the block, and serially streaming it out. For a final FPGA or ASIC design, this lag in performance should be ignored since data would be stored in flash memory and modified through memory access structures. Furthermore, data paths can be further reduced as more functions from the original algorithm are built. Along with the aforementioned work, the following should also be considered:

- Detailed, full implementation of both target tracking and ATR
- ASIC design for improved performance
- Inclusion of Image Fusion in performance testing

REFERENCES

[1] **Chellappa, Zhanfeng Tue and Rama.** *Synthesis of Novel Views of Moving Objects in Airborn Video.* College Park, MD : University of Maryland.

[2] **A. Karygiannis, E. Antonakakis, A. Apostolopoulos.** *Detecting Critical Nodes for MANET Intrusion Detection Systems.*

[3] **Anand patwardhan, Jim Parker, and Anupam Joshi.** *Secure Routing and Intrusion Detection in Ad Hod Networks.* s.l. : National Institute of Standards and Technology (NIST).

[4] **Gongjun Yan, Gyanesh Choudhary, Michele C. Weigle, Stephan Olariu.** *VANET'07 Poster: Providing VANET Security Through Active Position Detection.* 2007.

[5] *Design and Implementation of a Private and Public Key Crypto Processor and Its Application to a Security System.* **Lee, Howon Kim and Sunggu.** 2004, IEEE Transactions on Consumer Electronics, Vol. 50. 0098 3063/04.

[6] **Sarwono Sutikno, Andy Surya, and ronny Effendi.** *An Implementation of ElGamal Elliptic Curves Cryptosystems.* 1998. 0-7803-5146-0.

[7] *Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters.* **Shaohua Kevin Zhou, Rama Chellappa, and Baback Moghaddam.** 2004, IEEE Transactions on Image Processing.

[8] **Certicom.** *An Elliptic Curve Cryptography (ECC) Primer: why ECC is the Next Generation of Public Key Cryptography.* s.l. : Certicom, 2004.

[9] **Vanstone, Dr. S.A.** *Next Generation Security for Wireless: Elliptic Curve Cyrptography.* s.l. : Elsevier Ltd, 2003. 0167-4048/03.

[10] *Ant-based Adaptive Trust Evidence Distribution in MANET.* **Baras, Tao Jiang and John S.** s.l. : IEEE, 2004. Proceedings of the 24th International Conference on Distrubuted Computing Systems Workshops. 0-7695-2087-1/04.

[11] **Giry, Damien.** *Keylength.com - Cryptographic Key Length Recommendation.* [Online] [Cited: August 14, 2007.] http://www.keylength.com.

[12] **Hao Yang, Haiyun Luo, Fan Ye, Songwu Lu, and Lixia Zhang.** *Security in Mobile Ad Hoc Networks: Challenges and Solutions.* s.l. : IEEE, 2004. 1536-1284/04.