

1. Report No. SWUTC/09/167651-1		2. Government Accession No.		Recipient's Catalog No.	
4. Title and Subtitle Intelligent Transportation Systems Data Compression Using Wavelet Decomposition Technique				5. Report Date December, 2009	
				6. Performing Organization Code Report 167651	
7. Author(s) Fengxiang Qiao, Hao Liu and Lei Yu				8. Performing Organization Report No. Research Report 167651-1	
				10. Work Unit No. (TRAIS)	
9. Performing Organization Name and Address Department of Transportation Studies Texas Southern University 3100 Cleburne Houston, TX 77004				11. Contract or Grant No. 10727	
				13. Type of Report and Period Covered	
12. Sponsoring Agency Name and Address Southwest Region University Transportation Center Texas Transportation Institute Texas A&M University System College Station, Texas 77843-3135				14. Sponsoring Agency Code	
15. Supplementary Notes Supported by general revenues from the State of Texas.					
16. Abstract Intelligent Transportation Systems (ITS) generates massive amounts of traffic data, which posts challenges for data storage, transmission and retrieval. Data compression and reconstruction technique plays an important role in ITS data procession. Traditional compression methods have been utilized in Transportation Management Centers (TMCs), but the data redundancy and compression efficiency problems remain. In this report, the wavelet incorporated ITS data compression method is initiated. The proposed method not only makes use of the conventional compression techniques but, in addition, incorporates the one-dimensional discrete wavelet compression approach. Since the desired wavelet compression is a lossy algorithm, the balancing between the compression ratio and the signal distortion is exceedingly important. During the compression process, the determination of the threshold is the key issue that affects both the compression ratio and the signal distortion. An algorithm is proposed that can properly select the threshold by balancing the two contradicted aspects. Three performance indexes are constructed and the relationships between the three indices and the threshold are identified in the algorithm. A MATLAB program with the name Wavelet Compression for ITS Data (WCID) has been developed to facilitate the compression tests. A case study on TransGuide ITS data was put into play and a final compression ratio of less than one percent on the trade-off threshold value shows that the proposed approach is practical. Finally, the threshold selection algorithm can be further tuned up utilizing Autoregressive model so that the quality of reconstructed data can be improved with a minor overhead of saving only a few parameters.					
17. Key Words ITS Data Compression, Wavelet Decomposition, Signal Distortion; Compression Ratio, Threshold Selection, Autoregressive Model.				18. Distribution Statement No restriction. This document is available to the public through NTIS: National Technical Information Service 5285 Port Royal Road, Springfield, Virginia 22161	
19. Security Classify (of this report) Unclassified		20. Security Classify (of this page) Unclassified		21. No. of Pages 82	22. Price



Intelligent Transportation Systems Data Compression Using Wavelet  
Decomposition Technique

Fengxiang Qiao, Ph.D.

Hao Liu, M.S.

and

Lei Yu, Ph.D., P.E.,

Texas Southern University

3100 Cleburne Avenue

Houston, TX 77004

Research Report SWUTC/09/167651-1

Southwest Region University Transportation Center

Center for Transportation Training and Research

Texas Southern University

3100 Cleburne Avenue

Houston, Texas 77004

December 2009



## ABSTRACT

Intelligent Transportation Systems (ITS) generates massive amounts of traffic data, which posts challenges for data storage, transmission and retrieval. Data compression and reconstruction technique plays an important role in ITS data procession. Traditional compression methods have been utilized in Transportation Management Centers (TMCs), but the data redundancy and compression efficiency problems remain. In this report, the wavelet incorporated ITS data compression method is initiated. The proposed method not only makes use of the conventional compression techniques, but, in addition, incorporates the one-dimensional discrete wavelet compression approach. Since the desired wavelet compression is a lossy algorithm, the balancing between the compression ratio and the signal distortion is exceedingly important. During the compression process, the determination of the threshold is the key issue that affects both the compression ratio and the signal distortion. An algorithm is proposed that can properly select the threshold by balancing the two contradicted aspects. Three performance indexes are constructed and the relationships between the three indices and the threshold are identified in the algorithm. A MATLAB program with the name Wavelet Compression for ITS Data (WCID) has been developed to facilitate the compression tests. A case study on TransGuide ITS data was put into play and a final compression ratio of less than one percent on the trade-off threshold value shows that the proposed approach is practical. Finally, the threshold selection algorithm can be further tuned up utilizing Autoregressive model so that the quality of reconstructed data can be improved with a minor overhead of saving only a few parameters.

## **DISCLAIMER**

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof. Mention of trade names or commercial products does not constitute endorsement or recommendation for use. Trade and manufacturers' names appear herein solely because they are considered essential to the object of this report.

## **ACKNOWLEDGMENT**

The authors recognize that support for this research was provided by a grant from the U.S. Department of Transportation, University Transportation Centers Program to the Southwest Region University Transportation Center which is funded, in part, with general revenues from the State of Texas.





## EXECUTIVE SUMMARY

While the use of advanced technologies in transportation has been ongoing for many years, the creation of the ITS program has accelerated the pace of innovation and integration of technologies into the transportation system. The ITS program has brought new players into the transportation arena with interests in the application of technologies previously developed for defense, space and other fields. ITS data collection technologies have generated massive amounts of data to improve transportation system performance. More traffic management operators are considering the systematic retention of data generated by traffic monitoring devices. Numerous questions have been generated regarding ITS data storage and management. One of them, and probably the most intractable one, is the compression of ITS data.

Among the most effective methods on image and video compression is the wavelet compression algorithm which presents prominent ability to compress almost any kind of signal data. The ITS data collected by various forms of equipment are mostly in numeric format that can be treated as a signal, and then wavelet compression techniques could apply.

The goal of this research is to propose an ITS compression approach that includes both conventional compression methods and wavelet decomposition technique. To achieve this goal, three objectives need to be reached: First, an ITS data compression framework needs to be established and the methodology needs to be explained in the framework; second, a software program needs to be developed to put the methodology into work; and finally, a case study needs to be carried out to evaluate the methodology and test the program.

In this research, the wavelet incorporated ITS data compression method has been proposed, and a MATLAB GUI program with the name WCID has been developed to facilitate the compression tests. Finally, a case study on TransGuide ITS data was put into play and a final compression ratio of less than one percent on the trade-off threshold value shows that the proposed approach is practical.

Since the desired wavelet compression is a lossy algorithm, the balancing between the compression ratio and the signal distortion is exceedingly important. During the compression process, the determination of the threshold is the key issue that affects both the compression ratio and the signal distortion.

In this research, an algorithm is proposed that can properly select the threshold by balancing the two contradicted aspects. Three performance indexes RE, NZ and RR are constructed and the relationships between the three indices and the threshold are identified. Impact analysis of wavelet forms and decomposition levels to the compression ratios show that there is not too much difference in the selection of wavelet form. However, decomposition levels have significant impacts on the decomposition. Higher decomposition levels normally yield better compression ratios for the same threshold values.

ITS data quality control could be incorporated in the wavelet compression approach. It is found that the result of the compressed ITS data have the effect of de-noising due to the nature of the proposed wavelet compression. Considering that these abnormal data are usually erroneous or inaccurate measurements, data quality control could be well included by recalculating those data in the wavelet decomposition.

It is recommended that the compression processing speed should also be taken into consideration in order to meet the need of the increasingly surging ITS data. With more highway infrastructure put into use, the ITS data increase rate becomes overwhelming. A practical solution on ITS data compression must run faster on prevalent computers than the data-generating speed to be feasible.

## Table of Contents

ABSTRACT .....	v
DISCLAIMER.....	vi
ACKNOWLEDGMENT .....	vii
EXECUTIVE SUMMARY .....	ix
LIST OF TABLES.....	xiii
LIST OF FIGURES .....	xiv
ABBREVIATIONS .....	xv
CHAPTER 1 INTRODUCTION.....	1
CHAPTER 2 LITERATURE REVIEW.....	5
2.1 Existing ITS Data Archival Practices .....	5
2.2 ITS Data Collection Equipments .....	10
2.3 Data Compression .....	11
2.3.1 Lossless Compression: .....	11
2.3.2 Lossy Compression:.....	12
2.4 Wavelet Data Compression.....	13
2.5 Existing Applications on Wavelet Data Compression.....	18
2.5.1 Wavelet/Scalar Quantization (WSQ).....	18
2.5.2 LIGO Data Compression.....	20
CHAPTER 3 DESIGN OF STUDY .....	23
3.1 Framework and Methodology.....	23
3.2 Software Program Development.....	23
3.3 Case Study and Results Evaluation.....	24
CHAPTER 4 RESULTS AND ANALYSIS .....	25
4.1 ITS Data Characteristics Analysis .....	25
4.2 Framework and Methodology.....	25
Step 1: Data Format Compression.....	25

<i>Step 2: Wavelet Compression</i> .....	29
<i>Step 3: Threshold Selection</i> .....	29
<i>Step 4: Further Compression and Reconstruction</i> .....	39
4.3 Software Program Development - WCID .....	40
CHAPTER 5 CASE STUDY AND RESULTS EVALUATION .....	43
5.1 Data Description .....	45
5.2 Data Compression on a Typical Day .....	45
5.3 Data Compression on a Selected Detector Data for a Whole Month.....	51
5.4 Comparison of Different Ratio, Wavelet Forms and Levels .....	54
5.5 Added Benefit for Wavelet Compression .....	58
5.6 Fine Tuning on Signal Details by AR Modeling .....	58
CHAPTER 6 SUMMARY, CONCLUSIONS AND RECOMMENDATIONS .....	65
6.1 Conclusions .....	65
6.2 Recommendations .....	65
REFERENCES .....	67

## LIST OF TABLES

Table 1 Overview of Current ITS Data Archiving Practices.....	9
Table 2 TransGuide Data Archive Compression Ratio from 10-day Data Sample.....	44
Table 3 Compression Ratios under Different Steps by TransGuide and by Proposed Method ....	51
Table 4 Three Indexes and Compression Ration (CR) under Different Wavelets .....	56
Table 5 Statistical Analysis Results.....	57

## LIST OF FIGURES

Figure 1 ITS Data (Freeways) Collected vs. Archived.....	6
Figure 2 ITS Data (Arterials) Collected vs. Archived.....	7
Figure 3 Approximation and Details for a 3-Level Wavelet Decomposition.....	14
Figure 4 Extracting and Storing Approximation and Detail Coefficients.....	15
Figure 5 Demo of 1-D Wavelet Decomposition on a NoisedSine Wave.....	16
Figure 6 Threshold with Retained Energy and Number of Zeros.....	17
Figure 7 Overview of the WSQ Algorithm.....	19
Figure 8 Frequency Support of DWT Subbands in the WSQ Specification.....	20
Figure 9 Extraction Numeric Values in a Typical ITS DataFile.....	25
Figure 10 Wavelet Incorporated ITS Data CompressionFramework.....	27
Figure 11 Compression by Indexing Duplicate Items.....	28
Figure 12 Global vs. Level-dependent Thresholding.....	32
Figure 13 Whole Range Scan of Thresholding for TransGuide One Day Speed Data.....	35
Figure 14 Selection of the Proper Threshold by Proposed Algorithm.....	36
Figure 15 Comparison of Compression Results by Different Thresholding Methods.....	39
Figure 16 Compress the Sample Speed Data in WCID.....	41
Figure 17 Comparing the Original and Reconstructed Sample Speed Data in WCID.....	42
Figure 18 TransGuide Current Traffic Conditions Map.....	43
Figure 19 Reconstructed and Original Data, Compressed by Haar, Level 4.....	47
Figure 20 Performance Measurement Indexes for June 10, 2005 Speed data.....	48
Figure 21 Performance Measurement Indexes for June 10, 2005 Volume data.....	49
Figure 22 Performance Measurement Indexes for June 10, 2005 Occupancy data.....	50
Figure 23 One Month's Volume Data Before and After Compression (June 2005).....	52
Figure 24 Performance Measurement Indexes for a Month's Speed Data (June 2005).....	53
Figure 25 Compression Ratios Comparison between Current Practice and the Proposed Approach.....	54
Figure 26 The Autoregressive Model Algorithm Comparison.....	60
Figure 27 The Variance of the Residual Changes vs. Autoregressive Model Order.....	62
Figure 28 Performance Indexes Comparison Before and After AR Modeling.....	64

## ABBREVIATIONS

AR	AutoRegression
AVI	Automatic Vehicle Identification
ADUS	Archived Data User Service
CalTrans	California Department of Transportation
CCTV	Closed Circuit Television
DWT	Discrete Wavelet Transform
FHWA:	Federal Highway Administration
GUI:	Graphical User Interface
IEEE:	Institute of Electrical and Electronics Engineers
ISTEA:	Intermodal Surface Transportation Efficiency Act
ITE:	Institute of Transportation Engineer
ITS	Intelligent Transportation Systems
IVHS	Intelligent Vehicle Highway Systems
Mn/DOT	Minnesota Department of Transportation
PI:	Performance Index
PSD	Power Spectral Density
TEA-21:	Transportation Equity Act for the 21st Century
TEU:	Twenty-foot Equivalent Unit
TMC	Traffic Management Center
TxDOT	Texas Department of Transportation
UMD	University of Minnesota Duluth
USDOT:	U. S. Department of Transportation
VMT:	Vehicle Miles Traveled
WIM:	Weight-In-Motion
WSDOT	Washington State Department of Transportation
WSQ	Wavelet/Scalar Quantization





## CHAPTER 1

### INTRODUCTION

The Intelligent Transportation System (ITS) adds information technology to transport infrastructure and vehicles, aiming to manage vehicles, loads, and routes to improve safety and reduce vehicle wear, transportation times and fuel costs. Though information technology has been involved in transportation since the 1950s (MDOT, 2005), ITS is believed to have the first complete frameset architecture when the Intermodal Surface Transportation Efficiency Act (ISTEA) of 1991 established the national Intelligent Vehicle Highway Systems (IVHS) program (National Transportation Library, 1991). While the use of advanced technologies in transportation has been ongoing for many years, the creation of the ITS program has accelerated the pace of innovation and integration of technologies into the transportation system. The ITS program has brought new players into the transportation arena with interests in the application of technologies previously developed for defense, space and other fields.

Recently, the United States Department of Transportation (U.S. DOT) recognized the ITS data archival situation and defined it as an urgent problem and began promoting the needs for federal and local level research programs addressing the archiving and multi-agency use of data generated from ITS applications (USDOT, 2009). The Archived Data User Service (ADUS) was then initiated as a joint effort between ITS America and USDOT to meet this data archiving requirement. ADUS' vision is to "improve transportation decisions through the archiving and sharing of ITS generated data" (U.S. DOT ADUS, 2000).

ITS data collection technologies have generated massive amounts of data to improve transportation system performance. With the advent of the ADUS, more traffic management operators are considering the systematic retention of data generated by traffic monitoring devices. Numerous questions have been generated regarding ITS data storage and management. One of them, and probably the most intractable one, is the compression of ITS data.

The massive amount of ITS data has created obstacles for effective data storage, transmission and retrieval. Most of ITS field loop detectors or sensors collect traffic speed, volume, and occupancy data repeatedly at a relatively short time interval, such as 20 or 30 seconds (Turner, 2001). Considering the huge numbers of detectors and their 24/7 continuous operation, with an exception for hardware failures, the data will flood into traffic management

centers' (TMCs) database or data archival facilities at a staggering rate. Upon receiving and archiving the traffic data, TMCs will need to prepare data online for distance uses in that potential users do not necessarily work in these TMCs. Again, problems arise in massive data transmitting with limited internet band width.

For instance, from the 220 detectors located at the initial 26 miles of instrumented highways within the Texas Department of Transportation (TxDOT) TransGuide project, a total of 50-60 megabytes of traffic volume, speed, and occupancy data is gathered per day (TransGuide 2003). This is just the first phase of TransGuide. Currently, the TransGuide system covers 87 miles of San Antonio highways and the ultimate goal is to cover 289 miles of highways (TransGuide 2005). Therefore, it is believed that TransGuide ITS data will continue to increase to around 500 megabytes per day, or 15 gigabytes a month, which challenges the current computer ability in terms of storage, retrieval, and transmission. TransGuide archived the field data in the compressed text file (.z file) format and it can be decompressed by PKZip, WinZip, Solaris® COMPRESS or other equivalences in DOS, Windows, or UNIX environments. Two data sets are offered in these compressed files: the original data set has a 20-second interval, and the other one, derived from the original, has a 15-minute interval calculated on a two-minute running average. Current practices show that a compression ratio (the original file size divided by the compressed size) of 6:1 to 11:1 can be achieved (TxDOT 2009).

Another example (of what?) comes from Minnesota Department of Transportation (Mn/DOT), which manages a network of loop detectors from all metro freeways in and around the Twin Cities (Mn/DOT, 2009). Data is collected at a 30-second interval from about 4,000 loop detectors, seven days a week all year round. The collected data are packaged into a single zip (yyyymmdd.traffic) file on a daily basis and loaded into the University of Minnesota Duluth (UMD) ftp server from TMC. Similar to TransGuide, these files are also in a zip-compressed format and can be uncompressed using common unzipping software such as WinZip.

By these conventional data compression approaches, ITS data can be compressed at a particular rate around 8:1 (Cleary, 1990); therefore archival spaces and transmission times are saved. However, new compression techniques are still needed for two reasons: 1) the conventional data compression approaches are lossless methods, which have a 'ceiling' compression rate according to Shannon's Information Theory (Shannon, 1948). It is impossible to go any further than the information entropy; and 2) it is hard for the conventional data

compression approaches to offer various datasets for various different data requirements. For instance, ITS speed data are essential for both planning and incident detection. For the planning purpose, long-range data are required and an accuracy over 10 mph can be tolerated. However, for incident detection, engineers may only need speed data in a certain range of time period (several hours) but they cannot accept an error of even 5 mph.

Among the most effective methods on image and video compression, the wavelet compression algorithm presents prominent ability to compress almost any kind of signal data. A certain loss of quality being allowed, wavelet compression could reach the goal of storing data in as little space as possible, known as the lossy compression (Debra et al, 2005). The ITS data collected by various equipments are mostly in numeric format, that can be treated as a signal, and then wavelet compression techniques could apply.

The goal of this research is to propose an ITS compression approach that includes both conventional compression methods and wavelet decomposition technique. To achieve this goal, three objectives need to be reached: first, an ITS data compression framework needs to be established and the methodology needs to be explained in the framework; second, a software program needs to be developed to put the methodology into work; and finally, a case study needs to be carried out to evaluate the methodology and test the program.



## CHAPTER 2

### LITERATURE REVIEW

In this chapter, a thorough literature review will be conducted on existing ITS data archival and compression practices. This chapter also intends to explore state-of-the art/practice on data compression techniques and applications. Wavelet compression applications, as the major technological tool of this study, will be examined in great detail.

Section 2.1 provides the practical scan of the current ITS data management and archiving. Section 2.2 introduces general data compression concept and mechanism. Section 2.3 focuses on wavelet compression technique, including its principle and mechanism with in-depth analyses. Finally, Section 2.4 discusses the existing applications utilizing wavelet compression.

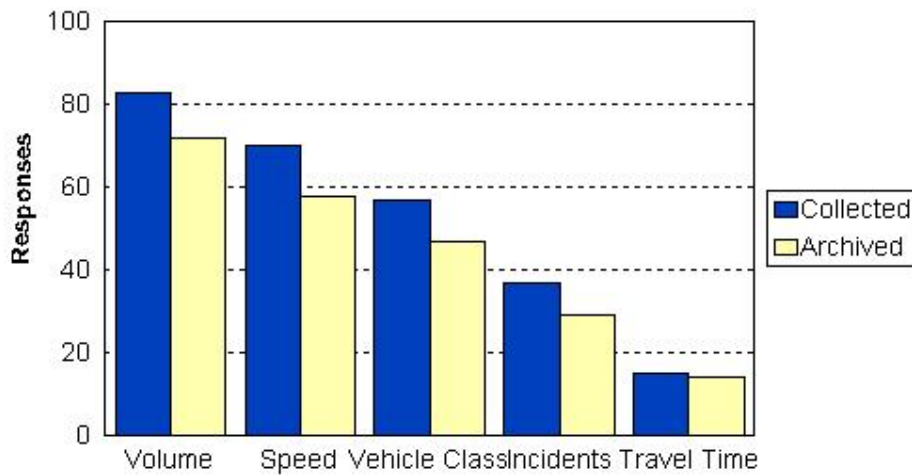
#### 2.1 Existing ITS Data Archival Practices

ITS applications and their sensors and detectors are potentially rich sources of data about transportation system performance and characteristics. Increasing deployment of ITS throughout the nation has brought an awareness that ITS data offer great promises for uses beyond the execution of ITS control strategies. Usually, ITS data refer to data that are typically collected by and/or generated for ITS applications. The most common data sources potentially available from ITS include (Liu et al., 2002):

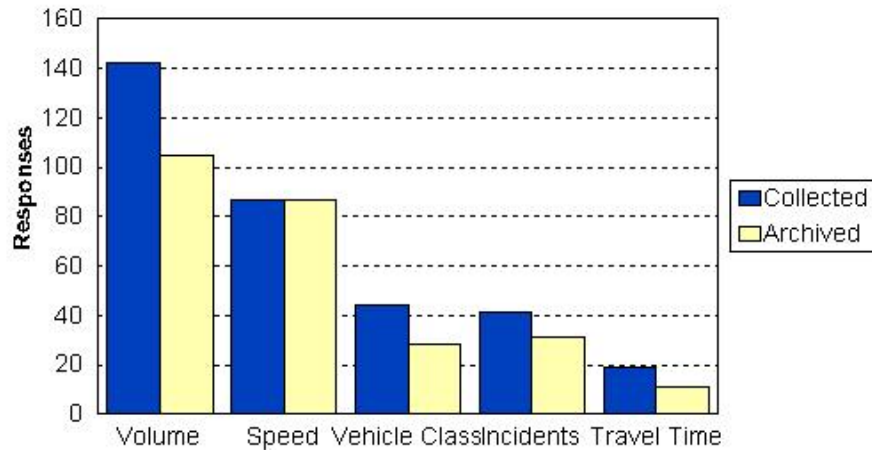
- Traffic Surveillance Data
  - Representative data elements: vehicle volume, speed, travel time, classification, weight, and trajectories;
- Traffic Control Data
  - Representative data elements: time and location of traffic control actions (e.g., ramp metering, traffic signal control, lane control signals, message board content);
- Incident and Emergency Management
  - Representative data elements: location, cause, extent, time history, detection and clearance of roadway incident/emergency;
- Public Transit Data
  - Representative data elements: transit vehicle boardings by time and location, vehicle trajectories, origins and destinations;
- Crash Data

- Representative data elements: location, time, cause, death, injury and clearance;
- Commercial Vehicle Operations Data
  - Representative data elements: cargo type, carrier, O/D, route and time; and
- Environmental and Weather Data
  - Representative data elements: location, time, precipitation, temperature and wind conditions.

The common ITS data collected by TMCs or transportation agencies fall into the following categories: traffic volume, traffic speed, vehicle classification, traffic incidents, lane occupancy, road and weather conditions, and current & scheduled work zones (FHWA, 2005). The USDOT Freeway Management Survey and the Arterial Management Survey showed that all the data collected are not archived (USDOT, 2005). In both surveys, as can be seen in Figure 1 and Figure 2, only 80% of collected ITS data have been archived. Volume data are the most popular collected and archived ITS data type according to this survey, the rest being speed, vehicle classification, incidents, and travel time respectively.



**Figure 1 ITS Data (Freeways) Collected vs. Archived.**  
 (Source: Office of Highway Policy Information, FHWA, 2005)



**Figure 2 ITS Data (Arterials) Collected vs. Archived**  
**(Source: Office of Highway Policy Information, FHWA, 2005)**

TMCs and transportation operations agencies are commonly the major groups for maintaining a data archive because they are responsible for saving their own data. Some operational workgroups only maintain recent data and transfer outdated data to other groups or locations for long-term storage or management. The ITS data are archived in a convenient compressed text format in most cases, but this format is neither easily accessible nor easy to use or analyze. These data archive managers are generally responsible for providing basic data archive functions such as performing quality control, ensuring data accessibility, providing information or documentation on data, providing software applications to analyze the data, etc.

Current ITS data archive approaches could vary from one transportation agency to another. For example, California Department of Transportation (Caltrans) has developed a Performance Measurement System (PeMS) which makes archived data and various data summaries available online (Dept of EE & CS, UC Berkeley, 2009). In Virginia, the Virginia Transportation Research Center stores statewide ITS data for both short-term and long-term use, and also takes the responsibility to distribute these data (USDOT 2009). In Washington, Washington DOT has developed analysis software and publishes an ITS data CD every three months (USDOT 2009).

As Zhang et al. stated in their research (2005), loop detectors data are also used to carry on vehicle type classification. The Washington State Department of Transportation (WSDOT) uses dual-loop detectors to measure vehicle lengths, and then they classify each detected vehicle

into one of four categories according to its length: passenger cars/light trucks, single unit trucks, double unit trucks, and triple unit trucks (WSDOT 2005).

These vehicle length data could be archived for multi purposes, such as a potential real-time truck data source for freight movement studies, or transportation planning and traffic analysis modeling.

As a summary, Table 1 shows the current ITS data archiving practices in several major TMCs or DOTs:



**Table 1 Overview of Current ITS Data Archiving Practices**

Location	Agency	Types of ITS Data
Phoenix, Arizona	Maricopa County DOT & Maricopa Association of Governments	Loop detector data from freeways and arterials (plans underway to archive all “relevant” data used by the Traffic Operations Center)
Los Angeles, California	Caltrans	Freeway loop detector data
Orange County, California	Caltrans	Arterial loop detector data
Berkeley, California (PeMS)	Caltrans & PATH	Freeway loop detector data
Chicago, Illinois	Illinois DOT	Loop detector data from selected freeways
Lexington, Kentucky	NORPASS, Kentucky Transportation Center	CVO (commercial vehicle operations) data, WIM (Weigh-in-Motion) data
Montgomery County, Maryland	Montgomery County DOT & Maryland National Capital Park and Planning Commission	Loop detector data from selected arterials
Detroit, Michigan	Michigan DOT	Loop detector data from freeways
Minneapolis-St Paul, Minnesota	Minnesota DOT	Loop detector data from freeways
TRANSCOM, New York/New Jersey/Connecticut	TRANSCOM	Travel times derived from AVI-equipped vehicles
Houston, TX	TRANSTAR	Travel times derived from AVI-equipped vehicles
San Antonio, TX	TransGuide	Loop detector data from freeways, travel times derived from AVI-equipped vehicles, and incident management data
Seattle, WA	Washington DOT	Loop detector data from freeways
Milwaukee, WI (MONITOR)	Wisconsin DOT	Freeway data collected by electronic detectors, closed circuit television cameras, ramp meters and variable messages signs

Traffic surveillance data are the primary type of data being archived. Since the study focused on data from TMCs, this result is not surprising. However, many transit systems that have deployed electronic fare payment and automatic vehicle location systems also routinely archive these data (including ridership counts by route segment and time of day, and station origin-destination patterns).

Many existing TMCs are either currently archiving or plan to archive traffic surveillance data. However, archiving is often implemented on an informal basis with no storage guidelines or limited access capabilities (Liu, 2002). Most TMCs store the original raw data files; while only a few of them use some popular data compression software techniques to compress the collected data before archiving them. The commonly used compression software tools include Winzip, Gzip, and compress (FHWA, 2005).

The ITS data quality is also an important issue in traffic data archiving for various reasons such as communication failure or hardware error, data error or when an inaccuracy occurs. Therefore, quality control techniques for archived data should be carried out to encompass at least missing data, suspect or erroneous data, and inaccurate data. The key way of knowing erroneous data from inaccurate data is the plausibility of the data. Erroneous data values do not fall within expected ranges or meet established principles or rules, while inaccurate data values are systematically inaccurate but within the range of plausible values. Basic quality checks, based on minimum and maximum flow thresholds, are often used for the detection of erroneous data. Weijermars and Van Berkun's research (2006) proposed quality checks methodology based on the principle of conservation. These quality checks are introduced for the detection of inaccurate data. The principle of conservation of vehicles implies that flow measurements have to be consistent between upstream and monitoring detectors within one intersection. Weijermars and Van Berkun (2006) argued that the quality checks based on the principle of conservation of vehicles are a useful addition to basic quality checks, since 95% of the invalid data detected by inconsistencies of flows between upstream and monitoring detectors was not detected by the basic quality checks.

## **2.2 ITS Data Collection Equipments**

ITS Data are generally collected through the following three types of equipment: Road-based Sensors, Closed Circuit Television (CCTV), and Vehicle Probes.

Road-based Sensors-- Mounted beneath the road, loop detectors are the most frequently used road-based sensors. Loop detectors provide vehicle counts, speed, volume, and occupancy data. Travel times data can be obtained by identifying and matching vehicles between adjacent loop detectors. As powerful and convenient as they are, loop detectors are not always accurate and often are non-functional due to software, hardware, or communication problems. Quality

control, therefore, is a critical issue before the loop detector data have been used for any practical use. Similar to loop detectors, RADAR detectors also provide traffic counts and density. They are mounted on the side of the road, and one sensor can monitor several lanes of traffic. The third popular type of road-based sensors is video image detection systems. This is the only sensor type that can read license plates so that vehicles can be re-identified in order to estimate travel time. However, the accuracy of video image detection systems is often seriously decreased in case of traffic jams, bad weather conditions, or poor light surroundings (Middleton et al. 1999).

Closed Circuit Television-- Video cameras provide immediate, intuitive and comprehensive pictures of traffic conditions. They are extremely useful in incident detection and response management because the incident level can be easily identified through picture and video. However, providing communications between the cameras and TMCs can be costly.

Vehicle P --In Houston, most toll road users have their vehicles equipped with electronics toll tags which allow users to pay their toll fee automatically. At the same time, these toll tags enable the Houston TMC Transtar to calculate travel times if the same vehicle is read at two locations. Incidents can be detected quickly by an unexpected travel time drop. The toll tag equipped vehicles are also seen in San Antonio and on a limited basis in New York (Dahlgren, et al, 2002). Vehicles with cell phones, though in early development status, are expected to provide large amounts of low-cost travel time information. GPS-equipped vehicles also have the ability to send a signal to data collecting devices when they pass locations of interest.

## **2.3 Data Compression**

Data compression has been investigated in the field of digital communication for a long time. Generally, data compression techniques can be divided into two major families: the lossless and lossy compression (Nelson, 1995).

### *2.3.1 Lossless Compression:*

Lossless compression consists of the techniques guaranteed to generate an exact duplication of the input dataset after a compress/decompress cycle. Lossless compression is essentially a coding technique. There are many different kinds of coding algorithms, such as Huffman coding (Huffman, 1952), run-length coding (Storer, 1988), and arithmetic coding (Witten et al., 1987).

### *2.3.2 Lossy Compression:*

Lossy data compression concedes a certain loss of accuracy in exchange for high compression ratio. Lossy compression proves effective when applied to digitized representations of analog phenomena. By their very nature, these representations are not perfect to begin with, so the idea of output and input not matching exactly is somewhat more acceptable. Most lossy compression techniques can be adjusted to different quality levels, gaining high accuracy in exchange for less effective compression. Lossless compression is a necessary component of every lossy compression approach.

Most of the lossy data compression algorithms follow similar methodology: the original data are mathematically transformed to a new domain in which they are better organized for data compression than in the normal spatial-temporal domain (Dahlgren et al, 2002). Therefore, the choice of mathematical transformation is crucial to the performance of compression algorithms.

JPEG--the Joint Photographic Experts Group works on a format for still pictures that also allows for compression. The JPEG format allows for lossy as well as lossless compression (Salomon et al, 2007).

MPEG--the Moving Picture Experts Group works on techniques for compressing moving image data. Moving images such as video frames are typically more difficult to compress than still images, because frames are related in time to their predecessors and successors as well, and good compression techniques must note and use this fact (Salomon et al, 2007).

Wavelet Compression--since the 1980s, the traditional study of digital signals based on Fourier transforms has been replaced by wavelet analysis. The "Haar wavelet" and similar ones are used as a basis for image analysis, compression, and regeneration (Salomon et al, 2007).

Scalar Quantization--in this technique, lossy compression is achieved by means of approximating the color values of pixels; for instance, in place of 256 shades of gray, which are unnecessary since the human eye can only distinguish 30 to 40, one can approximate the same image with just 32 or 64 shades of gray (Salomon et al, 2007).

Vector Quantization--in this technique, the compression technique works on an array of independent values, rather than on single pixel values (Salomon et al, 2007).

## 2.4 Wavelet Data Compression

Wavelet compression, which has recently started to become one of the most popular data compression techniques (Salomon et al, 2007), is a form of data compression originally well suited for image compression (sometimes also video compression and audio compression). The goal is to store image data in as little space as possible in a file. A certain loss of quality is accepted (i.e. the compression is lossy).

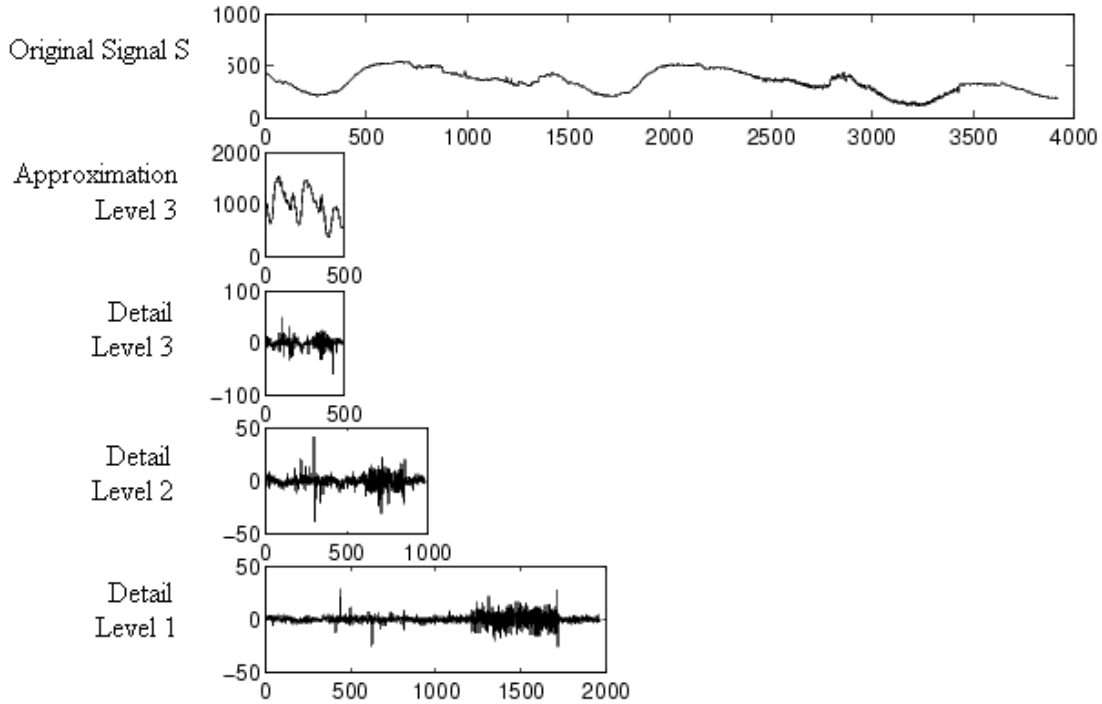
Using a wavelet transform, the wavelet compression methods are better at representing transients, such as percussion sounds in audio, or high-frequency components in two-dimensional images, for example an image of stars on a night sky. This means that the transient elements of a data signal can be represented by a smaller amount of information than would be the case if some other transforms, such as the more widespread discrete cosine transform, had been used.

As powerful and practical as it is, wavelet compression is not good for all kinds of data: transient signal characteristics mean good wavelet compression - smooth, periodic signals are better compressed by other methods (Salomon et al, 2007). ITS data can be well suited in this category because it is considerably smooth and periodic when we look at it in a long range, say, a week or a month. Wavelet compression has also been proven to be suitable for other transportation data, such as time-difference-of arrival data in emitter location finding (Yu et al, 2008).

One big difference between wavelet transform and Fourier transform is how fast these transforms converge to a function. In the Fourier domain, all the elements of the basis are active for all time, i.e., they are non-local. Consequently, Fourier series converge very slowly when approximating a localized function (Cohen, 1992). Wavelet transform makes up for the deficiencies of Fourier transform. Wavelet basis function is a novel basis localizing in both time domain and frequency domain (Daubechies, 1992). Therefore, wavelet basis function can provide a good approximation for a localized function with only a few terms.

The compression features of a given wavelet basis are primarily linked to the relative scarceness of the wavelet domain representation for the signal. The notion behind compression is based on the concept that regular signal components can be accurately approximated using the following elements: a small number of approximation coefficients (at a suitably chosen level) and some of the detail coefficients (Misiti et al, 2001).

The one-dimensional discrete wavelet compression starts with the multilevel decomposition of the original signal  $S$  (i.e., the original ITS data set before wavelet compression.). Figure 3 illustrates a 3-level wavelet decomposition and its coefficients' storage.

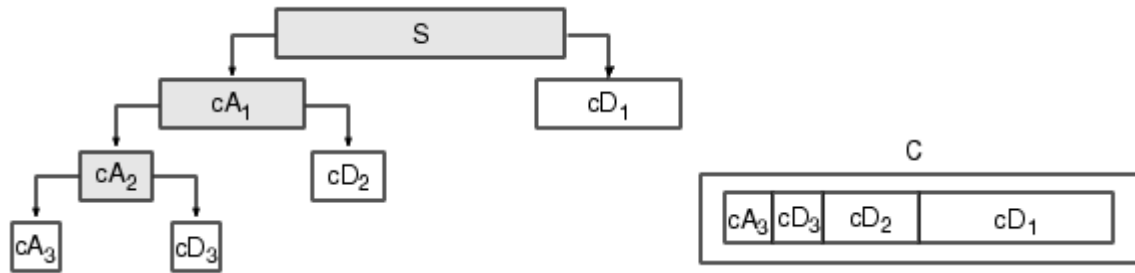


**Figure 3 Approximation and Dfor a -evel Wavelet Decomposition**

**Source: Matlab Wavelet Toolbox, Version 14**

Three details ( $D1$ ,  $D2$ , and  $D3$ ) and one approximation  $A3$  are shown in Figure 8 while compared with the original signal  $S$ . In the case of  $n$ -level decomposition, the original signal  $S$  is currently represented by  $D1$ ,  $D2$ ,  $D3$  and  $A3$ . Or:  $S = D1 + D2 + D3 + A3$ . The size of the approximation and the detail for each level is only half of the previous level, which is obviously illustrated in Figure 8.

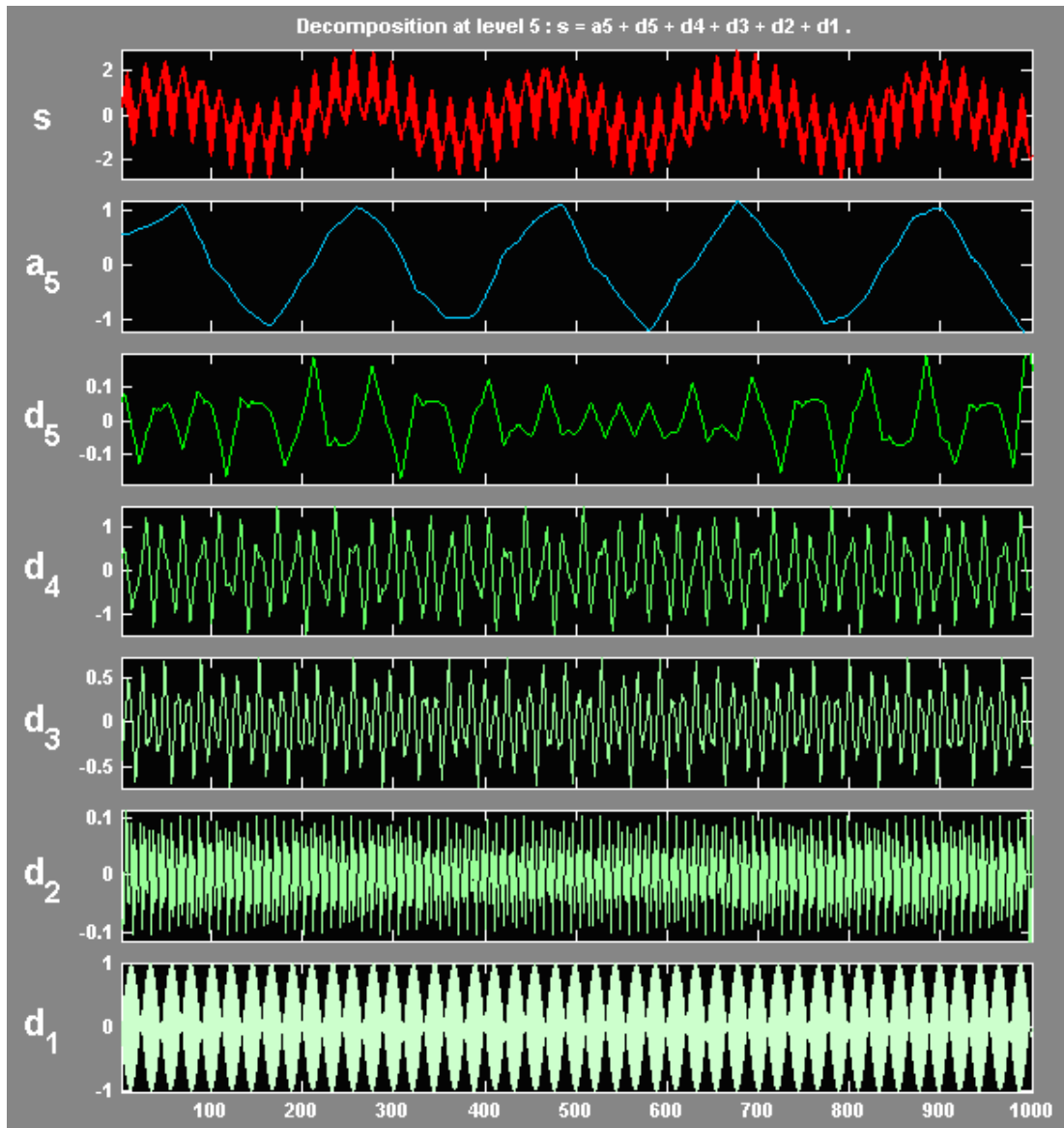
The coefficients of all the components of a 3-level decomposition (that is, the third-level approximation and the first three levels of detail) are returned concatenated into the singular vector  $C$ , as is shown in the Figure 4.



**Figure 4 Extracting and storing approximation and detail coefficients**

**Source: Matlab Wavelet Toolbox, Version 14**

Since the length of the coefficients for each level varies, the matrix is unevenly participated with  $cD_1$ , which represents the coefficients of the detail for the first level of detail, being the longest, and  $cA_3$  as well as  $cD_3$ , which represent the coefficients of the third level of approximation and detail, being the shortest.



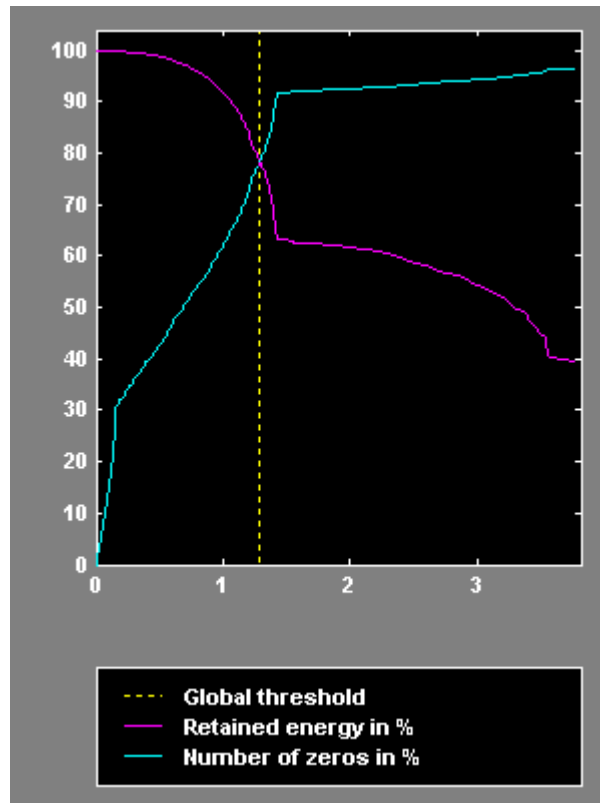
**Figure 5 Demo of 1-D Wavelet Decomposition on a Noised Sine Wave**

Figure 5 shows a noised sine wave decomposed by Wavelet db3 at level 5. The approximation  $a_5$ , together with 5 levels details ranging from  $d_5$  to  $d_1$ , demonstrate the wavelet decomposition's ability of extracting different frequency components from a given signal. The approximation  $a_5$  displays the trend of the signal, which is the lowest frequency part; while the details showed the “noise” – the higher frequency parts,  $d_1$  being the highest.

The mechanism of one-dimensional discrete wavelet decomposition is to set a threshold for each or all details. All values that are below the given threshold(s) are set as zeros. Coding



these zero value coefficients yields a better compression ratio of the signal. In the Section “Fine Tuning on Signal Details by AR Modeling” of this chapter, an effort will be made trying to replace the set-zero algorithm with an autoregressive model.



**Figure 6 Threshold with Retained Energy and Number of Zeros**

Figure 6 shows the relationship between threshold selection, retained energy and number of zeros in the processed coefficients. As the threshold increases, the retained energy of the signal drops down, while the number of zeros, which reflects the compression ratio, grows up. A trade-off balance point being found, we could compress the signal, as well as maintain satisfying information that the signal carries.

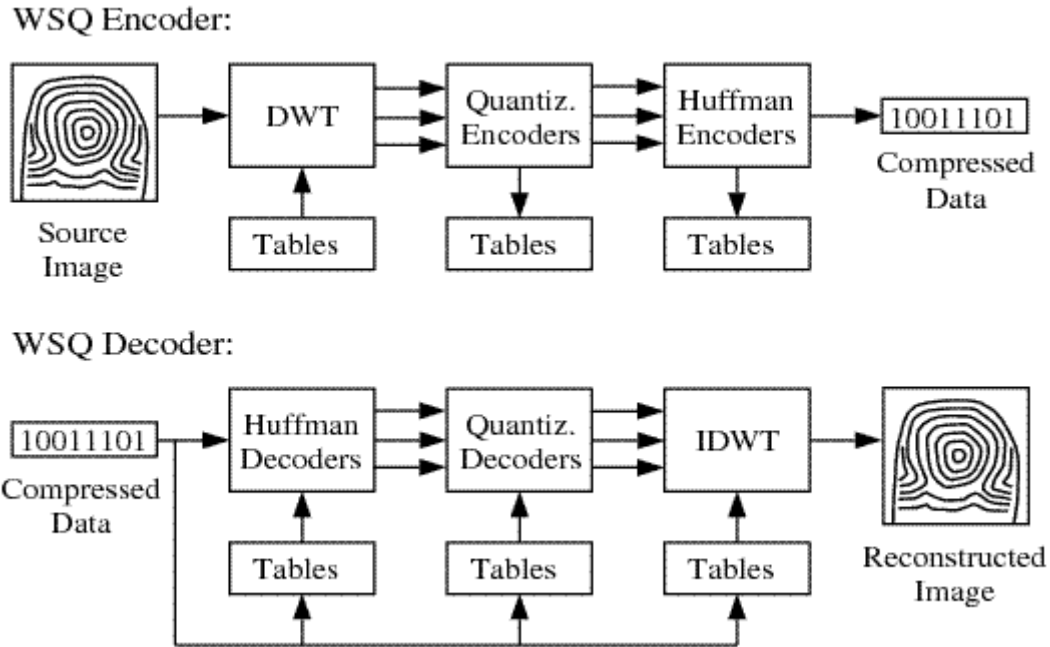
The coefficient sets after thresholding, can be combined together to achieve the reconstructed signal  $S_r$ . Any difference between the reconstructed signal  $S_r$  and the original signal  $S$  is the distortion of the entire wavelet decomposition process.

## 2.5 Existing Applications on Wavelet Data Compression

### 2.5.1 Wavelet/Scalar Quantization (WSQ)

The FBI has formulated national standards for digitization and compression of gray-scale finger print images (Klimenko *et al*, 2002). The compression algorithm for digitalized images is based on adaptive uniform scalar quantization of discrete wavelet transform sub-band decomposition, a technique referred to as the wavelet/scalar quantization method. The algorithm produces archival-quality images at compression ratios of around 15 to 1 and will allow the current database of paper fingerprint cards to be replaced by digital imagery (Klimenko *et al*, 2002). The fingerprint database consists of around 200 million inked fingerprint cards to a digital electronic format. A single card contains 14 separate images. Digitization thus converts a single fingerprint card into about 10 megabytes of raster image data; this, coupled with the size of the FBI's criminal fingerprint database, gives some indication of why image compression was deemed necessary.

Since lossless compression of gray-scale fingerprint images appears to be limited to compression ratios of less than 2:1, the FBI specified a lossy method utilizing a WSQ algorithm for the fingerprint image compression standard (Klimenko *et al*, 2002). The WSQ algorithm produces archival-quality images at compression ratios of about 15:1. The general structure of the compression standard is a specification of syntax for compressed image data and a specification of a "universal" decoder capable of reconstruction compressed images produced by any compliant encoder, as shown in the Figure 7.

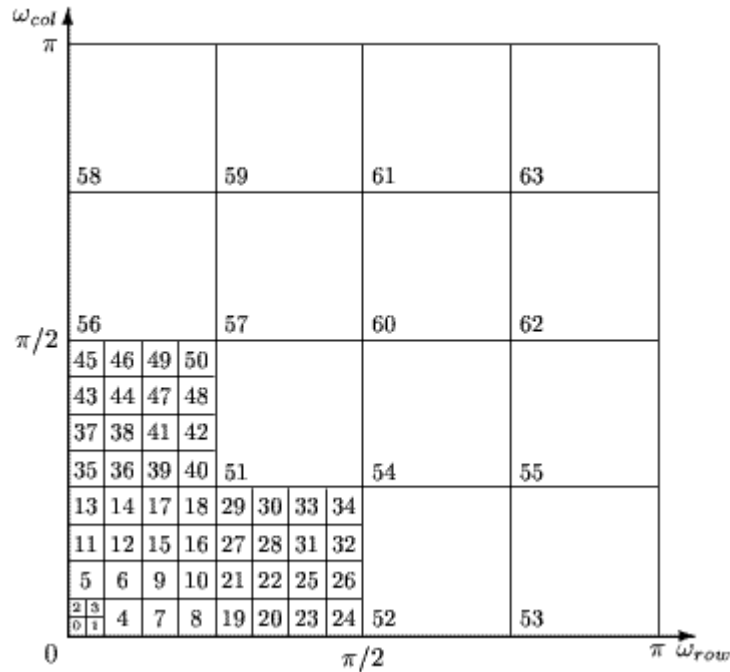


**Figure 7 Overview of the WSQ Algorithm**

(Source: A Lossless Compression of LIGO Data, Klimenko et al, 2002)

Encoding consists of three main processes: discrete wavelet transform (DWT) decomposition, scalar quantization, and Huffman entropy coding. The WSQ decoder must, in turn, be capable of decoding these three processes and all variants of them that are allowed under the general Specification.

The first step to compress the fingerprint images is to decompose them into 64 spatial frequency sub-bands using a two-channel perfect reconstruction multi-rate filter bank (PR MFB), implemented in two dimensions as a separable (or product) filter bank with up to five levels of cascade. A frequency-domain depiction of this decomposition is shown in Figure 8. Note the use of unequal bandwidths, with the low and midrange frequencies partitioned into very narrow bands.



**Figure 8 Frequency Support of DWT Sub-bands in the WSQ Specification**  
 (Source: *A Lossless Compression of LIGO Data*, Klimenko et al, 2002)

The WSQ Specification allows for the potential use of different filters in different encoders to achieve the decomposition. In particular, the WSQ Specification allows for the use of any two-channel linear phase FIR filter bank with filters up to 32 taps long. This class of PR MFB’s divides up into two distinct groups: odd-length filter pairs, in which both impulse responses are symmetric about their center taps (the so-called “whole-sample symmetric,” or WS/WS, filter banks), and even length filter pairs, in which the low pass impulse response is symmetric and the high pass impulse response is anti-symmetric (the “half-sample symmetric/anti-symmetric,” or HS/HA, filter banks) (Klimenko et al, 2002).

### 2.5.2 LIGO Data Compression

LIGO is one of the most data-intensive projects. The expected total bit-rate is 15MB/s and the full two year LIGO data stream will yield about one petabyte (1000 terabytes, or 10<sup>15</sup> bytes) of data. The design of the data reduction procedures, which produce scientific data sets, is one of the important tasks of the LIGO data analysis.

There are four levels of the data, ranging from the full interferometer data (Level 0) to whitened GW strain data (Level 3, 1/1000 of the full data stream) (Klimenko et al, 2000). The full data stream will be available for about 16 hours after acquisition, but will not be archived. The data will be processed to form the Archived Reduced Data Set (Level 1) that will be about 1/10 of the full data stream. At these two stages of analysis, it is important to save all useful data with minimal losses. Thus, fast and efficient lossless data compression can be essential for the generation of the Archived Reduced Data Set.

The idea of using wavelets is to decompose data into components that can be fairly well described as a white Gaussian noise. In other words, wavelets are used to decorrelate the data, which means the representation of data in wavelet domain is more compact than the original representation. A method is represented for the lossless data compression based on the lifting wavelet transform (LWT) that maps integers to integers. The wavelet transform works in combination with the random data compression (rdc) encoder that is optimized for compression of random Gaussian signals.

For lossless compression, an invertible wavelet transform that maps integers to integers is needed (Klimenko et al, 2000). Another requirement is to find the wavelet representation of the data quickly. For these reasons the biorthogonal lifting wavelet transform is used, which can map integers and allow switching between the original data and its wavelet representation in a time proportional to the size of the data.

The engineering run data collected in April 2000 has been used to test the different compression methods. Three encoders were compared: the gzip, eri and the rdc. The encoders were applied to the original data in time domain (TD) and to the decorrelated data. Differentiation, wavelet transform (NP=6) and wavelet binary tree transform (NP=6) are used to decorrelate data. The combination of wavelet + rdc shows better compression ratio than the other methods. Compared to the traditional differentiation + gzip method, the average compression ratio for 16kHz channels is better by  $\sim 20\%$  and for the 2kHz channels the improvement is  $\sim 30\%$ .



## CHAPTER 3

### DESIGN OF STUDY

This chapter describes the design of study in this thesis, including framework and methodology, software program development, and case study. This study, initiated and funded by the Southwest Region University Transportation Center (SWUTC) project No. 167651, also responds to the needs of ADUS by focusing on developing an efficient ITS data compression system that allows efficient archiving and retrieval of large scaled data.

#### 3.1 Framework and Methodology

The first step is to establish the framework for the ITS data compression approach. Then, ITS data characteristics analyzed. The purpose of the analysis is to remove information redundancy and to convert the data to a format which is more suitable for compression. Next, wavelet compression will be applied on the new converted data. The compression process should make the balance between compression effect and data distortion by utilizing proper thresholding. To facilitate proper threshold selection, indices are to be created to quantify the data distortion and compression effect. Different wavelet forms and decomposition levels will be compared to identify the best wavelet combination for ITS data. After the first two steps of the compression process, the ITS data set are further compressed by available conventional compression tools, which will result an additional data size reduction. Finally the compressed data will be reconstructed. The reconstructed data should have no significant difference with the original ones.

#### 3.2 Software Program Development

A software program is to be developed and designed to carry out and formulate the steps of the framework. The basic function of the program is to compress and reconstruct given ITS data set by the proposed approach. Users should be able to input the ITS data file, select wavelet form and level, choose compression options, and obtain the results of compressed and reconstructed data. The program is to have graphic interface so that users have intuitive results of the compression.

### **3.3 Case Study and Results Evaluation**

To evaluate the proposed compression approach and to test the program, a case study is needed. A proper data source needs to be identified as the test-bed for the case study. The collected data will be analyzed and redundancy will be removed. The processed data will then be compressed by the program and the results will be evaluated. A number of wavelet forms, levels, and compression options will be compared to achieve the best results. The proposed compression results will be compared with those from the traditional compression techniques and conclusions will be drawn.



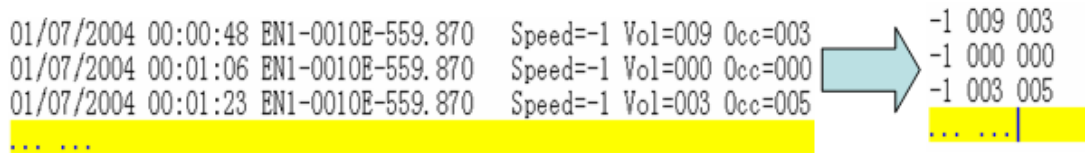
## CHAPTER 4

### RESULTS AND ANALYSIS

#### 4.1 ITS Data Characteristics Analysis

This research is to study ITS data compression. First, existing data archives and management practices are summarized. Common ITS data collecting equipments and techniques are also discussed in this section. Finally, ITS data characteristics are presented and will indicate how these data will be used as signals.

The ITS data collected by various equipment are mostly in numeric format, which can be treated as a signal, and then signal compression techniques could apply. A typical ITS data file contains tens of thousands of lines, each line being with the same format. A line usually starts with date and time information then gives the detector location and name along with the measured values (See Figure 5). The date and time show at each record, and there is usually a fixed amount of time difference between one record and the next as the 16 or 17 seconds shown in Figure 5. The long detector name duplicates itself for each record as well.



```
01/07/2004 00:00:48 EN1-0010E-559.870 Speed=-1 Vol=009 Occ=003
01/07/2004 00:01:06 EN1-0010E-559.870 Speed=-1 Vol=000 Occ=000
01/07/2004 00:01:23 EN1-0010E-559.870 Speed=-1 Vol=003 Occ=005
... ..
```

The figure illustrates the extraction of numeric values from a typical ITS data file. On the left, a sample of raw data lines is shown, each containing a date, time, detector name, and measured values. A blue arrow points from the raw data to the extracted numeric values on the right, which are highlighted in yellow. The extracted values are: -1 009 003, -1 000 000, -1 003 005, and ... ..

**Figure 9 Extraction Numeric Values in a Typical ITS Data File**

To extract the numerically measured values only for the data set in Figure 9, and to index the other redundant information (such as date, time and detector name) would not only have compressed the data set to some extent, but it would also have better represented the data set in the way that is similar to digital signals.

#### 4.2 Framework and Methodology

The proposed approach is designed to indisputably make full use of the existing conventional data compression techniques that are ready to use including the suitable data format

converting technique, coding techniques, WinZip, etc. In addition, the advanced wavelet decomposition based data compression technique is employed as a novel perfection.

This framework includes four major steps: Data Format Conversion, Wavelet Compression, Threshold Selection, and Further Compression and Reconstruction. The first two steps prepare the original data to proper format and the appropriate wavelet type is selected. Step 3, Threshold Selection, makes trade-off between compression rate and data conservation rate. Thus, data requirement from different user groups can be met by adjusting the threshold value. Step 4 seeks data compression techniques other than wavelet to further increase the efficiency, and discusses methods by which the original data is reconstructed with minimum loss. The framework of this wavelet incorporated ITS data compression approach is constructed and illustrated in Figure 10.



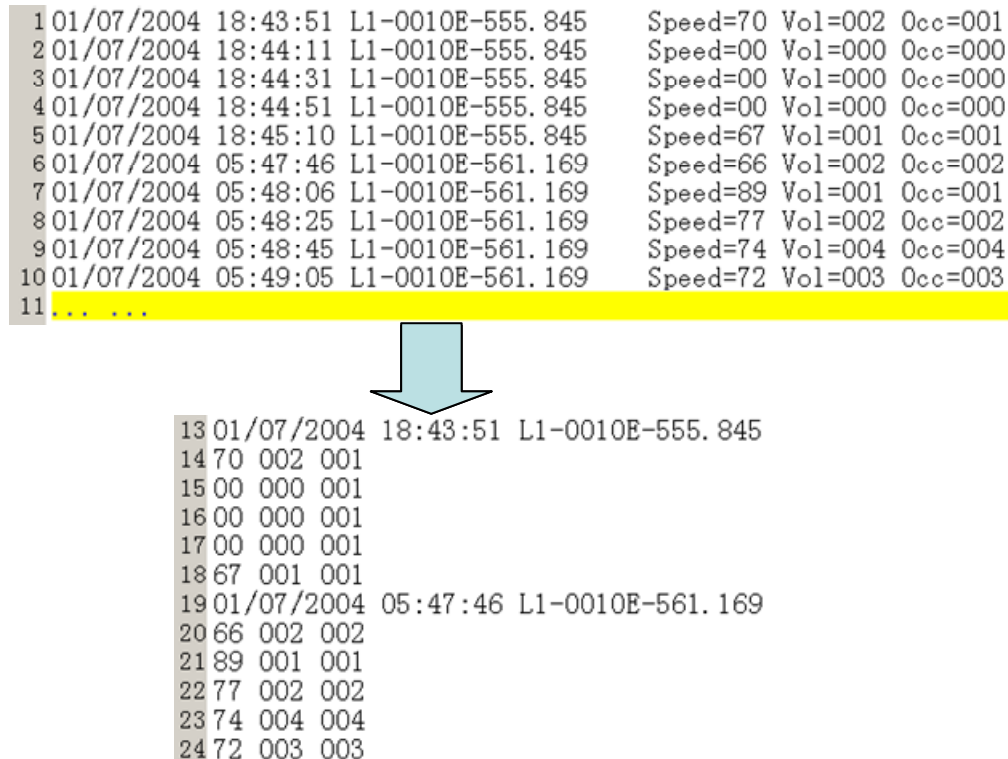
**Figure 10 Wavelet Incorporated ITS Data Compression Framework**

*Step 1: Data Format Conversion*

In the first step, the ITS data format will be analyzed carefully. All redundant data items (such as time, date, detector name, etc.) are then eliminated, because they repeatedly occur on each data item and thus can be easily identified and reconstructed from either the file name or

additional tiny marking files. This saves a large portion of space before any formal data compression.

Having investigated the ITS data characteristics, it is quite clear that these data could be compressed by traditional compression techniques. The basic idea is to reduce the file size by extracting and indexing the duplicate data items. Figure 6 is the demo showing how this is done.



**Figure 11 Compression by Indexing Duplicate Items**

As can be seen in Figure 11, only the starting time for one detector is saved in the compressed text since the time for next record can be calculated by adding the same time period. For example, as shown in Figure 6, the starting time is 18:43:51 for detector L1-0010E-555.845, and the time interval is 20 seconds, thus the following 5 records were taken at time 18:43:51, 18:44:11, 18:44:31, 18:44:51, and 18:45:11 respectively. Due to the accuracy of the detectors, the calculated time may not be exactly the record time by the detector. For example for detector L1-0010E-555.845, the fifth record occurred at 18:45:10, while the calculated time is 18:45:11. This issue could be addressed in the following way:

For a day's data, count the first data entry and last entry, and arrange a plan for the all middle data entries. For example, the first entry is at 00:00:01, and the last entry is at 11:50:20, therefore, there are a total of 42,620 seconds in this time period. If we have 2200 reported data entries, the time interval between two consecutive entries would be  $42,620/2200 = 19.37$  seconds. Then, all the 2200 data entries can be distributed into the time period from 00:00:01 to 11:50:20 every 19.37 second, and round up to the whole second.

After the preparation step, the ITS data are digitalized; that is to say, only the numbers are kept in chunk for the data; the rest, like the detector number, the date and time are indexed and could thus be temporarily not significant compared to the numbers. The following research, therefore, focuses on only the number values of the ITS data. After the compression, the non-number data could be recovered in the reconstruction step.

The ITS data will be converted from “text” to “binary”, which takes much less space for significant space savings.

### *Step 2: Wavelet Compression*

In this step, the ITS data will be further compressed by the wavelet compression technique. The one-dimensional discrete wavelet transform is used to decompose the data into different levels. The approximation, as well as some part of the details, will be kept while other components will be marked as zero if they are beyond the thresholds.

The basic wavelet decomposition is conducted by using the wavelet toolbox in MATLAB (Mathworks, 2001). Proper wavelet form and decomposition levels need to be carefully selected and a program package developed.

### *Step 3: Threshold Selection*

The threshold is important in ensuring a relatively lower compression rate while keeping less distortion. A calculated index named “Retained Energy” (RE for short) is defined to control compression quality. Two other constructed indices are the “Number of Zeros” (NZ for short) and the “Reduced Ratio” (RR for short), and they serve as important parameters in determining thresholds.

**Hard and Soft Thresholding** The threshold selection is a critical issue in the process of wavelet decomposition. As aforementioned, higher thresholds lead to better compression ratio

and greater signal distortions, while lower thresholds keep more signal energy, but the compression rate usually is not satisfied. A well designed algorithm is indispensable for better locating the suitable threshold(s) for ITS data compression.

Hard thresholding and soft thresholding are two predominant thresholding schemes of wavelet decomposition. Hard thresholding sets any coefficient less than or equal to the threshold to zero on a given signal, while keeping the coefficient greater than the threshold unchanged. For this reason, hard thresholding is sometimes referred to as the “keep or kill” scheme. In pseudo code, hard thresholding can be expressed as following:

*For i = 1 to levels*

*if (coef[i] <= thresh)*

*coef[i] = 0.0;*

*Next i*

Soft thresholding also sets any coefficient less than or equal to the threshold to zero, as does hard thresholding. Besides, soft thresholding subtracts the threshold value from any coefficient that is greater than the threshold. Soft thresholding is commonly used if we expect the resulting signal to be smooth. It can be explained by the following pseudo code:

*For i = 1 to levels*

*if (coef[i] <= thresh)*

*coef[i] = 0.0;*

*else*

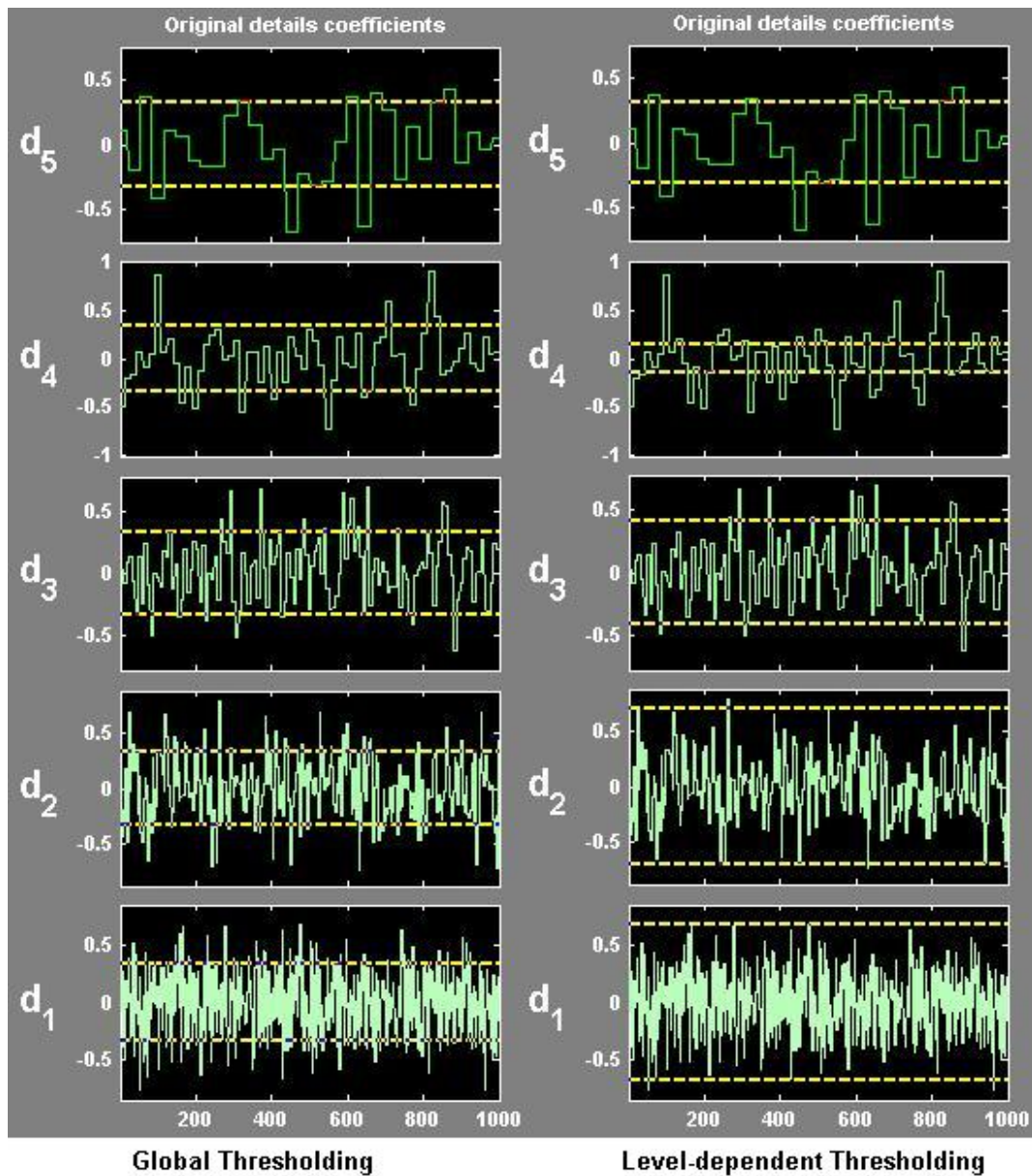
*coef[i] = coef[i] - thresh;*

*Next i*

The soft thresholding has been proven to be better suited for most wavelet de-noising applications because it produces smoother reconstructed signals. However, hard thresholding performs better in processing ITS data. First, hard thresholding is much more efficient than its soft counterpart, as its “keep or kill” method imposes calculation on only the coefficient values less than or equal to the threshold, compared to soft thresholding algorithm’s unavoidable calculation on every coefficient value. Second, the purpose of compressing ITS data is not to

make the signal smooth, but to keep the original signal shape as much as we can, as well as to minimize the compression rate. Soft threshold removes more information from the original signal, thus less retained energy – which will be discussed in greater detail in this section – can be kept. For the sake of conserving data information, hard thresholding is selected in the research.

**Global or Level-dependent Thresholding** When thresholding the coefficient details, two common approaches are usually applied on how to select threshold values between different levels. Global Thresholding is the quick and easy way to conduct the compression with less computational cost. Level-dependent thresholding, on the other hand, offers choices to manually select different thresholding values for each level of detail coefficients. In light of the complex characteristics and various purposes of the ITS data, it is most likely to take a series of trial-and-error steps to decide the optimal threshold value, and even more, to decide the levels of details. In this case, the level-dependent threshold is technically impossible to deal with since comparisons between two sets of thresholding plans on the same date will be impossible due to the various thresholds imposed on different levels. The research thus adopts the global thresholding approach to conduct the ITS data compression. Figure 12 demonstrates the concept of global and level-dependent thresholding.



**Figure 12 Global vs. Level-Dependent Thresholding**

**Three Compression Indexes** It is needless to say that Performance Measurement (PM) is required to systematically assess progress made in threshold selection. As was mentioned in Step 3 of the decomposition framework, three compression indexes are defined as PM indicators to determine the thresholding value: the “Retained Energy” or RE; the “Number of Zeros” or NZ, and the “Reduced Ratio” or RR.



According to signal processing theories, signals have energy. Energy of a signal can be understood as the signal's "ability to work". The energy  $E_s$  of a continuous-time signal  $x(t)$  is defined as the integration of the signal square on all time:

$$E_{s-c} = \int_{-\infty}^{\infty} |x(t)|^2 dt \quad (1)$$

Similarly, the energy of a discrete signal  $x$  is defined as the sum of squared moduli

$$E_{s-d} = \sum_{n=0}^{N-1} |x_n|^2 \quad (2)$$

where  $N$  is the number of total signal sample points.

The Percentage of Retained Energy (RE) is to measure how much energy is retained in the compressed signal out of the original one. RE is measured in percentage as following:

$$RE = 100 \cdot \frac{\sum_{n=0}^{N-1} |x_{nc}|^2}{\sum_{n=0}^{N-1} |x_{nr}|^2} \quad (3)$$

where  $x_{nc}$  is the compressed signal, and  $x_{nr}$  is the original signal, or

$$RE = 100 \cdot \frac{N_c^2}{N_r^2} \quad (4)$$

where,  $N_c$  is the norm of the compressed signal, and  $N_r$  is the norm of the original signal.

The Number of Zeros (NZ) is used to measure the effect of the ITS data compression. The more zeros that occur in the final coefficients, the better result we have achieved. This is simply because the zeros are easy to get compressed in the final signal. NZ is the number of zeros in a given level of decomposition divided by the number of the total number of coefficients in this decomposition:

$$NZ = 100 \cdot \frac{Z_c}{Z_s} \quad (5)$$

where,  $Z_c$  is the number of zeros of the current decomposition,  $Z_s$  is the number of coefficients.

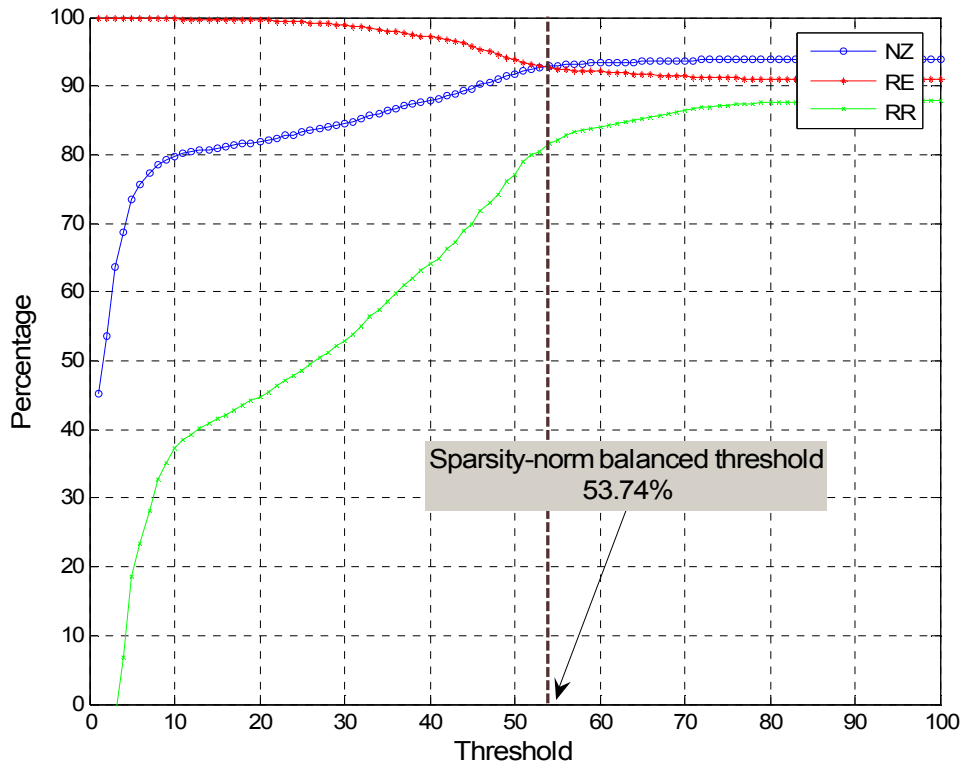
The Reduced Ratio RR is designed to be defined as:

$$RR = 100 \cdot \frac{B_c}{B_s} \quad (6)$$

where,  $B_c$  is the size of original signal subtracted from the binary signal after wavelet compression at the particular threshold, and  $B_s$  is the size of the original binary signal before wavelet compression.

Among the three compression indexes, Retained Energy (RE) represents the congruency of the compressed signal with the original signal, or the reverse of the signal distortion; while the Number of Zeros NZ and the Reduced Ratio RR have similar physical meanings, both reflecting the compressed effects in size and related to each other. NZ is the performance measurement factor coming directly from the wavelet decomposition, therefore it could be used to evaluate the algorithm used; in contrast, RR puts more concern on how the final data size reduces from the original file size.

**Balancing Compression Indices** In order to construct a suitable algorithm for the balanced threshold, the varying of the three compression indices (RE, NZ, and RR) with the whole range of threshold for a typical set of ITS data is plotted in Figure 13. The data were the June 10, 2005, speed set from the detectors in San Antonio, TX downloaded from TransGuide server. While the following analysis uses this set of data as an illustration, the stated phenomenon and the resulted algorithm are suitable for the other ITS data sets that the authors have dealt with in this study.



**Figure 13 Whole Range Scan of Thresholding for TransGuide One Day Speed Data**

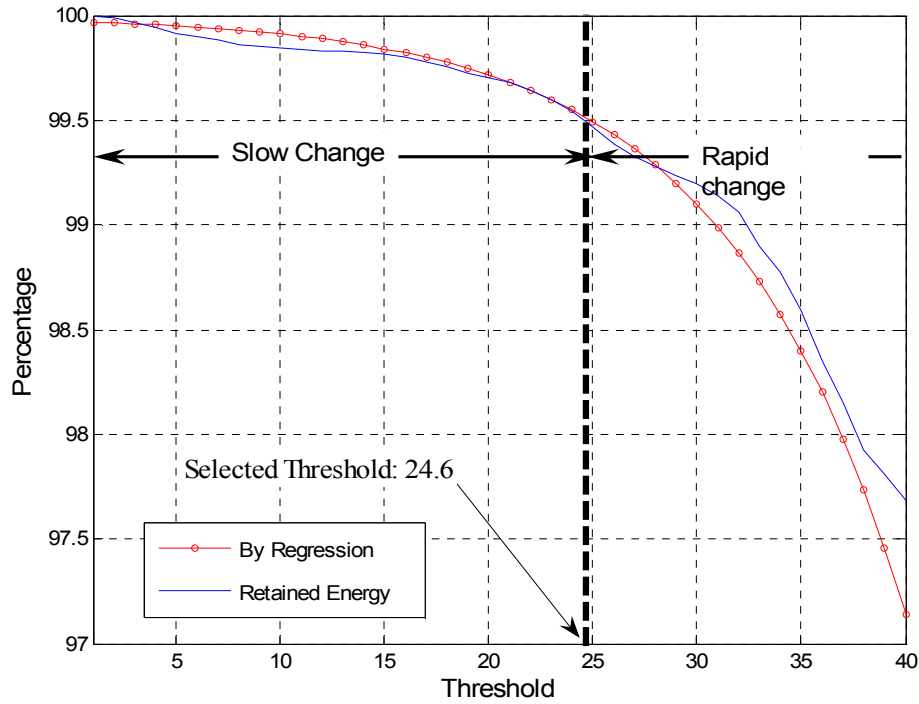
It can be seen from Figure 13 that the RE curve (Retained Energy) is decreasing with the increase of the threshold, meaning that greater thresholds will cause bigger energy losses and large distortions. However, the NZ curve (Number of Zero) and RR curve (Reduced Ratio) are increasing as the threshold goes up, meaning that smaller thresholds can maintain better (smaller) compression ratios. The NZ curve and the RR curve go in the same direction, though they are not parallel, especially when thresholds are small.

The threshold, when the RE curve and the NZ curve intersected each other, is called the “sparsity-norm balanced” threshold. The sparsity-norm balanced threshold is normally used as the global threshold for wavelet compression in many cases to get the trade-off between congruency and compression ratio. However, for the ITS data set, the signal distortion under the sparsity-norm balanced threshold is too high.

To further search for a better threshold that can provide less distortion, focus should be placed on the left hand side of the sparsity-norm balanced threshold since the retained energy is

decreasing with threshold. During this range, the NZ curve and the RR curve increase sharply when threshold is very small, then go up in a smaller, but still high slope. However, the NZ curve keeps a very high value and then drops down faster.

Figure 14 shows a close look of the RE curve when the threshold is less than the sparsity-norm balanced value. It seems the RE curve in this period follows an exponential function, with a slow changing period and a rapid changing period. So, one of the feasible ways is to set the place where the RE curve changes from slow to rapid as the desired threshold.



**Figure 14 Selection of the Proper Threshold by Proposed Algorithm**

The RE curve during this period can be fitted by a kind of attenuation function in the form of a deformed exponential curve:

$$RE = (100 - RE_s) \cdot \left( 1 - e^{-\frac{t-t_s}{\tau}} \right) + RE_s \quad (7)$$

where,  $t$  is the threshold variable;  $\tau$  is the parameter to be calibrated with the name as *time constant*;  $t_s$  is the threshold at the sparsity-norm balanced point; and  $RE_s$  is the Retained Energy at  $t_s$ .

To calibrate the parameter  $\tau$ , Equation (7) can be transformed to:

$$\ln\left(1 - \frac{RE - RE_s}{100 - RE_s}\right) = \frac{t - t_s}{\tau} \quad (8)$$

This equals to:

$$Y = bX \quad (9)$$

where,

$$Y = \ln\left(1 - \frac{RE - RE_s}{100 - RE_s}\right) \quad (10)$$

$$X = t - t_s \quad (11)$$

and

$$b = 1/\tau \quad (12)$$

The calibration of parameter b in Equation (9) can follow any regular routine for the linear equation calibration. The inverse of b returns the parameter  $\tau$  in Equation (7).

According to the theory of signal system analysis, the slow change and the rapid change separate at (Marven 1996):

$$\frac{t - t_s}{\tau} = \lambda \quad (13)$$

where,  $\lambda$  is normally between -2 and -3, which reflect the case when the fitted exponential curve increases from the very bottom point (at the sparsity-norm balanced point) to its  $\exp(-2) = 85.02\%$ , and  $\exp(-3) = 95.02\%$  of the total height (from the sparsity-norm balanced point to the maximum RE point.)

The proper threshold  $t_p$  and the corresponding Retained Energy  $RE_p$  can be determined on the RE curve from the following formula:

$$t_p = t_s + \lambda\tau \quad (14)$$

$$RE_p = (100 - RE_s) \times (1 - e^\lambda) + RE_s \quad (15)$$

Another way to determine the threshold position  $t_p$ , where the slow change and rapid change separate, is to set the exponential part as  $\alpha$ :

$$e^\lambda = \alpha \quad (16)$$

with a typical range of  $\alpha$  setting as  $0.05 \leq \alpha \leq 0.15$ . In exponential functions, this range is usually considered the slope of the curve changes from slowly to rapidly. The physical meaning

of  $\alpha$  is the percentage that the exponential curve has dropped from the top maximum value in the threshold range starting from the smallest to the sparsity-norm balanced point.

The threshold  $t_p$  in this case can be calculated based on (13) and (16):

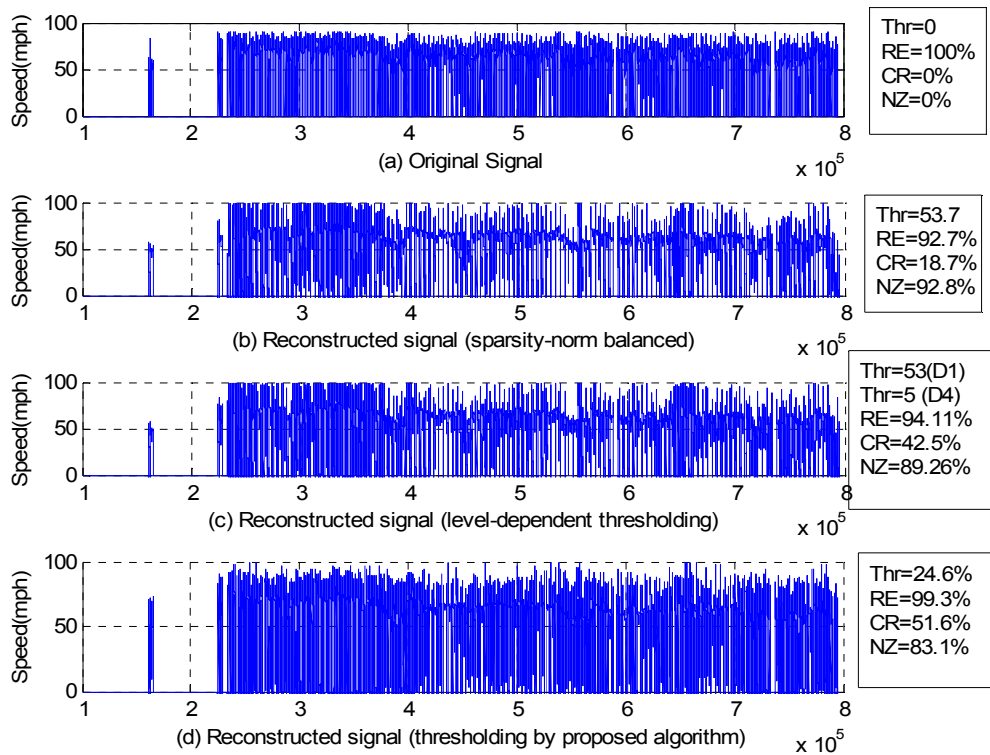
$$t_p = t_s + \lambda\tau = t_s + \tau \cdot \ln \alpha \quad (17)$$

Then,  $RE_p$  can be more explicitly expressed as:

$$RE_p = (100 - RE_s) \times (1 - \alpha) + RE_s \quad (18)$$

In summary, the algorithm of the proper threshold selection is listed as following:

1. Calculate RE, NZ, and RC by changing the threshold within the whole range based on the wavelet compression technique, obtaining the whole range curves RE, NZ, and RC;
2. Locate the sparsity-norm balanced point, which is the intersection between the RE curve and the NZ curve;
3. Obtain the threshold  $t_s$  and the RE value  $RE_s$  at the sparsity-norm balanced point;
4. Calculate the RE value at the proper threshold  $RE_p$  using either Equation (15) or (18). If Equation (18) is used, parameter  $\alpha$  ( $0.05 \leq \alpha \leq 0.15$ ) should be predefined;
5. Trace the threshold value  $t_p$  from Equation (14) or (17).



**Figure 15 Comparison of Compression Results by Different Thresholding Methods**

Figure 15 shows the original ITS signal and the reconstructed signals after wavelet compression under different thresholding methods. It is seen that the proposed thresholding algorithm presents the least distortion and acceptable compression ratio (in (d) of Figure 15) compared with the other thresholding methods in (b) and (c).

*Step 4: Further Compression and Reconstruction*

The ITS data set, after the first three steps, is further compressed by the conventional data compression technique: WinZip. This consideration will engender a supplementary reduction of the size of data sets and attain the finally compressed version.

The finally compressed ITS data set will be stored or archived in TMCs, and transmitted to users as required. At the users' ends, the compressed data sets will be reconstructed following the inverse actions as compression. The objective is that the reconstructed data sets should have no "significant" difference with the original ones. Or at least, the differences are within a range that the users can tolerate.

### 4.3 Software Program Development - WCID

A MATLAB GUI program with the name Wavelet Compression for ITS Data (WCID) was developed for this research. WCID well formulates the basic 4-step framework of the ITS data wavelet compression work by compressing raw ITS data in different thresholds, and make the threshold-norm-compressed-ratio picture. The inputs of the program are extracted ITS single variable data, such as speed. WCID accepts six options before compressing the data: wavelet form, wavelet level, threshold lower limit, threshold upper limit, threshold step value, and thresholding type. WCID can show two types of graph results. The first one is the 3-index result under a range of threshold, intending to help users to see how the performance measures change and to give an estimate of the optimal threshold value. This type of graph uses the x-axis as the threshold value, and the y-axis as the values of the three indexes, presented in percentages, as can be seen in Figure 16. The other type of graph shows the comparisons of the original signal and the reconstructed signal under the threshold value specified as the upper limit of threshold range. Users can have an intuitive picture of how much the signal distortions are under the given threshold value as shown on the graph in Figure 17. The two types of graph can be toggled by simply clicking the “View Indexes” or “Comparing Signals” button. The WCID program is coded in MATLAB GUIDE resulting in two files: a fig file to store the interface and a MATLAB m file to store the code. The core WCID code can be found in Appendix A.



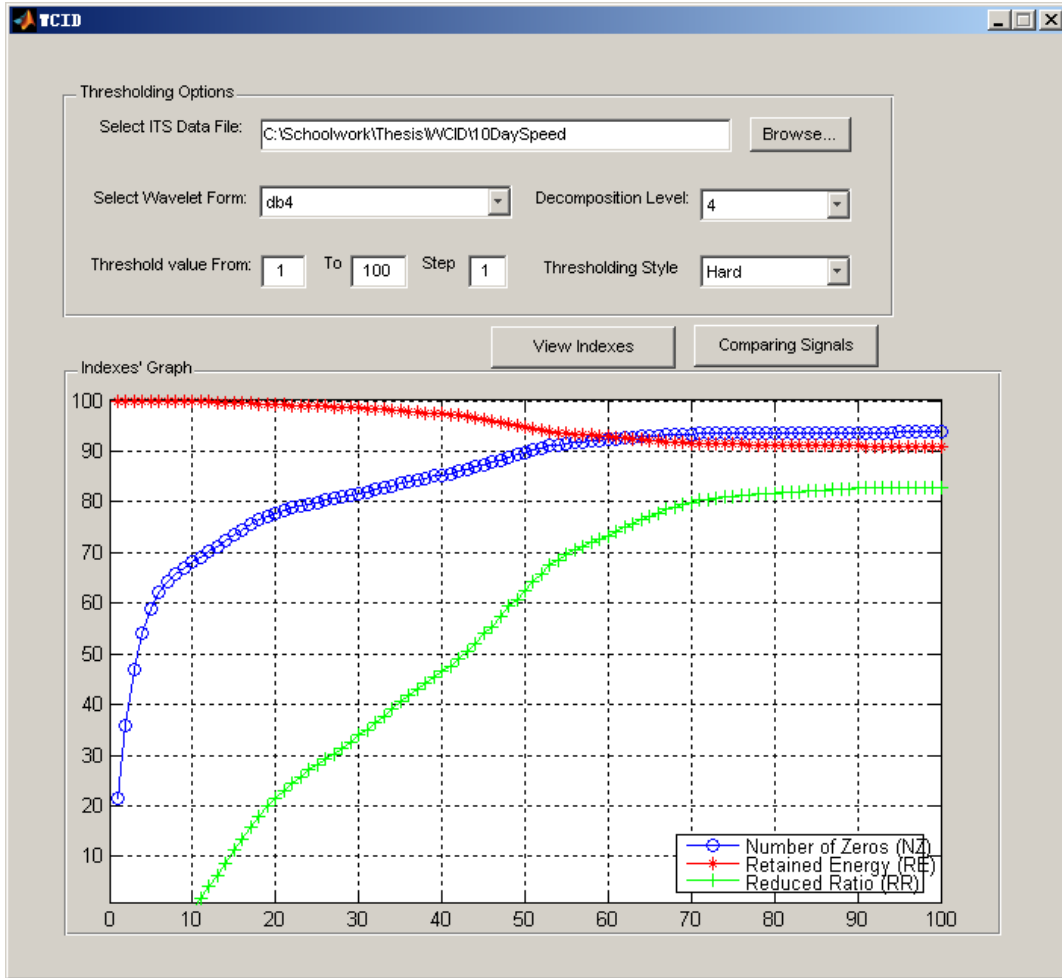
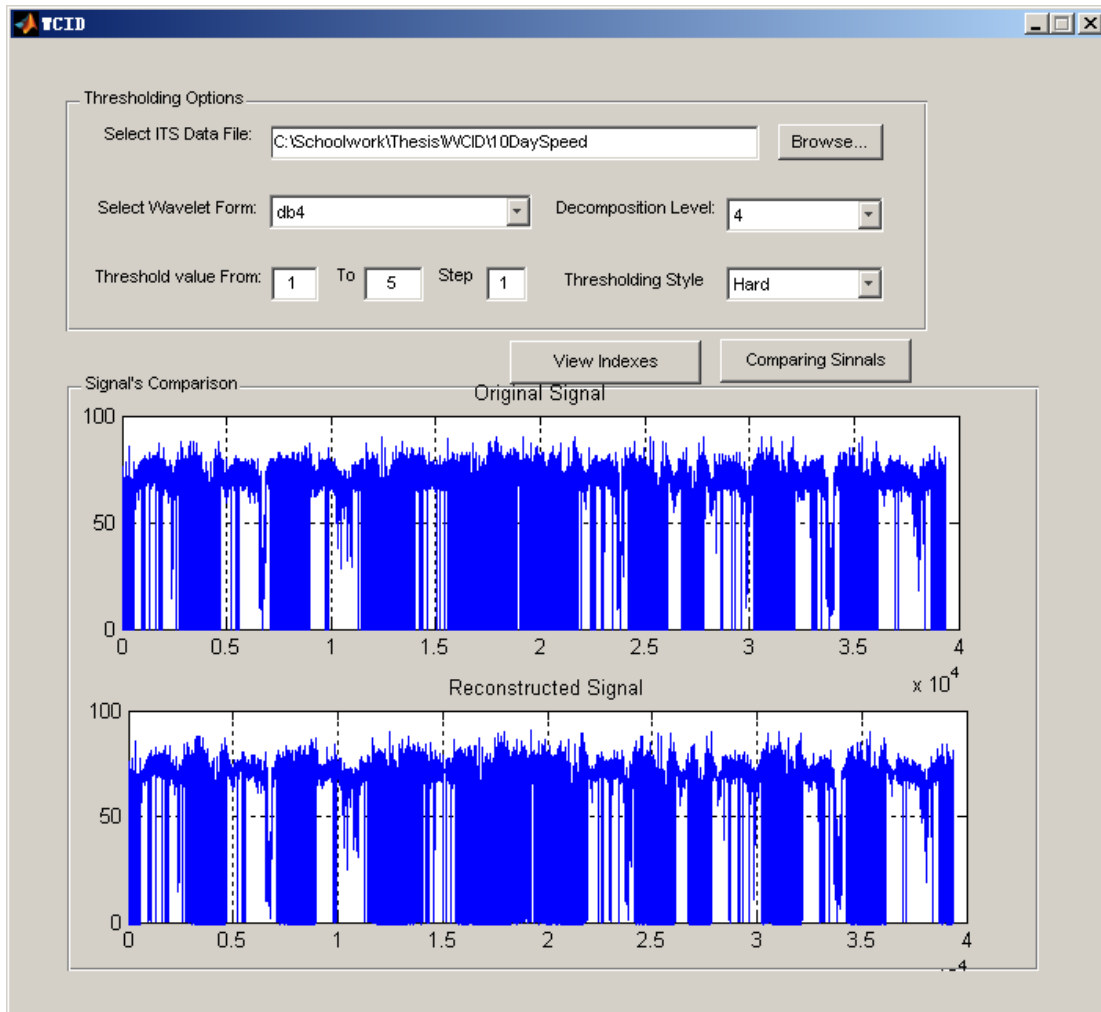


Figure 16 Compress the Sample Speed Data in WCID

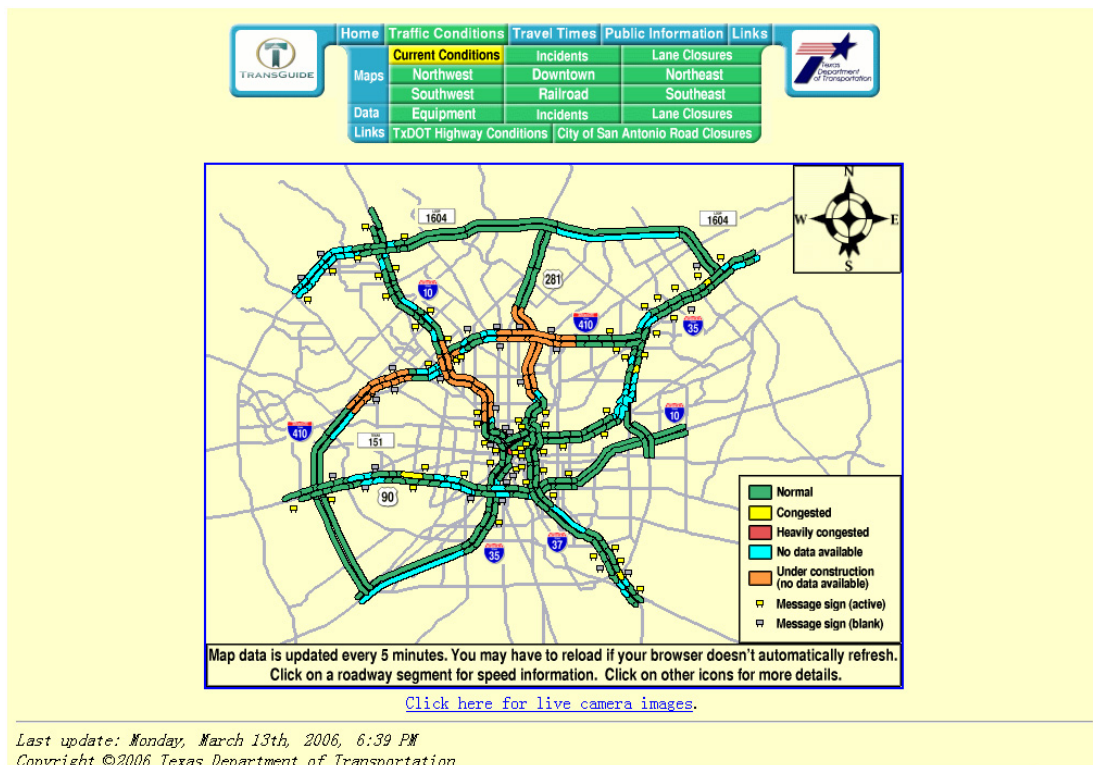


**Figure 17 Comparing the Original and Reconstructed Sample Speed Data in WCID**

## CHAPTER 5

### CASE STUDY AND RESULTS EVALUATION

In this thesis research, the San Antonio TransGuide (TxDOT, 2005) ITS data was selected to illustrate the proposed methodologies and the effectiveness of data compression. TransGuide is an ITS project designed by the San Antonio District of the Texas Department of Transportation (TxDOT). This TransGuide project offers traffic volume, speed, and occupancy data gathered from the initial 26 miles of instrumented highways. Figure 18 demonstrates TransGuide's current traffic conditions map.



**Figure 18 TransGuide Current Traffic Conditions Map**  
(Source: TransGuide Website, Traffic Condition Map, 2006)

TransGuide offers two levels of data sets. The original data has a 20-second interval, and the other one, derived from the original, has a 15-minute interval calculated on a two-minute running average. This proposed research focuses on the original 20-second data set. The

downloadable data contain the compressed text files (.z file) and can be decompressed by PKZip, WinZip, Solaris® COMPRESS or other equivalences in DOS, Windows, or UNIX environments. Current practices show that a compression ratio (the compressed data size divided by the original data size) of 7% to 15% (based on TransGuide’s 21 month’s data from June 2004 to February, 2006) are reached. Table 2 demonstrates the size and compression ratio for these 10 day’s data

**Table 2 TransGuide Data Archive Compression Ratio from 10-day Data Sample**

Date	Original Data Size (KB)	Unzipped Data Size (KB)	Compression Ratio
06/01/2005	6,134	51,451	11.92%
06/02/2005	6,438	53,488	12.04%
06/03/2005	6,582	54,450	12.09%
06/04/2005	6,393	54,019	11.83%
06/05/2005	6,158	53,639	11.48%
06/06/2005	6,524	54,599	11.95%
06/07/2005	6,550	54,854	11.94%
06/08/2005	6,276	52,607	11.93%
06/09/2005	5,764	48,538	11.88%
06/10/2005	6,115	50,504	12.11%
Total:	62,934	528,149	11.92%

The 20-second interval data were selected for this case study. The compression effects under each step in the framework were conducted one after another. During the third step threshold selection, the threshold of wavelet decomposition and the three composition indexes (RE, NZ and RR) were examined, and the optimal threshold was selected based on the proposed algorithm. TransGuide started gathering these ITS lane data from late 2003 on its eight data collecting servers numbering 0 to 7. Each server works separately and generates a data file each day. The data collection had some ups and downs in the early stages, as TransGuide does not have all data files for all servers. It was not until December 2004 that all servers started working on a reliable basis and all eight data files were available for each day. Over 100 data files were analyzed by WCID and three scenarios were selected to demonstrate the performance of WCID. The three scenarios include a single-day-all-detector case (June 10, 2005), a 10-day-single-detector case (June 1-10, 2005), and a 10-day-all-detector case.

## 5.1 Data Description

The retrieved traffic variables include speed, volume and occupancy. The sizes of the data file currently zipped on the TransGuide server are normally around six megabytes per server per day with the unzipped original file size about 50 megabytes. The total zipped size for the 10 day's ITS data is 61.4 megabytes (515 megabytes for decompressed 'raw-text' files.)

In the originally decompressed file, the TransGuide ITS lane data were arranged in the following form:

```
06/10/2005 00:00:42 L2-0010E-557.394    Speed=65 Vol=003 Occ=001
06/10/2005 00:00:42 L2-0010W-557.358    Speed=70 Vol=008 Occ=003
06/10/2005 00:00:42 L3-0010E-557.394    Speed=61 Vol=006 Occ=003
06/10/2005 00:00:42 L3-0010W-557.358    Speed=67 Vol=013 Occ=005
06/10/2005 00:00:42 L4-0010E-557.394    Speed=53 Vol=002 Occ=001
06/10/2005 00:00:43 EN1-0010W-557.926   Speed=-1 Vol=009 Occ=004
```

For the sake of indexing, these data need to be put in an order in which the entries are logical and the traffic trend is easy to observe. After sorting the data first by detector name, then by time, the reformatted data file looks like this:

```
06/10/2005 23:59:05 L1-0010E-559.873    Speed=73 Vol=001 Occ=001
06/10/2005 23:59:24 L1-0010E-559.873    Speed=00 Vol=000 Occ=000
06/10/2005 23:59:44 L1-0010E-559.873    Speed=71 Vol=001 Occ=001
06/10/2005 00:00:44 L1-0010E-560.424    Speed=64 Vol=003 Occ=001
06/10/2005 00:00:55 L1-0010E-560.424    Speed=00 Vol=000 Occ=000
06/10/2005 00:01:15 L1-0010E-560.424    Speed=00 Vol=000 Occ=000
```

## 5.2 Data Compression on a Typical Day

The compression of ITS data, including all three variables- speed, volume and occupancy, was conducted on the June 10, 2005 data. As the sampling interval was 20 seconds, there were a total of 9200,000 pairs of records obtained for that singular day, taking a space of 51,715,560 bytes. After compression by TransGuide, the compressed size was: 6,261,413 bytes with a compression ratio of 12.11%.

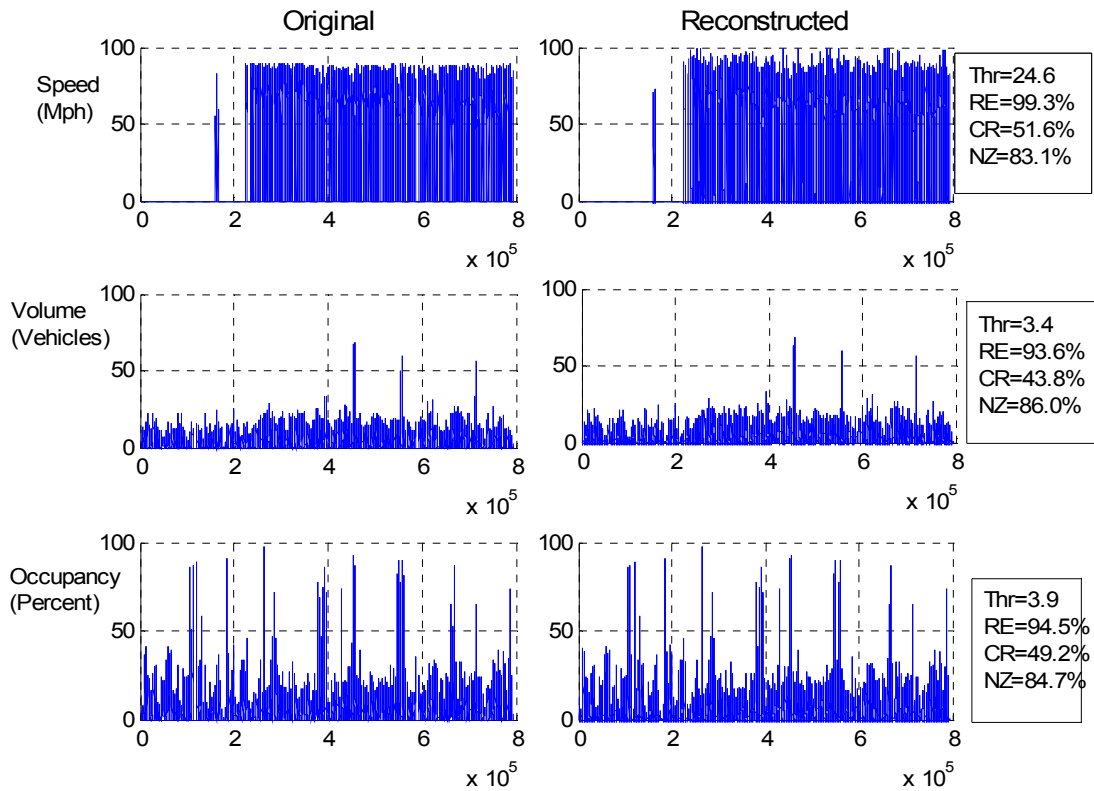
The first step of the compression process was to rearrange the data format in order to reduce redundancy. Information concerning the redundancies was recorded in the extra tiny-size file for the purpose of future decompression. Only the detected values of speed, volume and occupancy were used for further compressions. With this arrangement, the file size was reduced to 11,138,730 bytes, with a compression ratio of 21.53%.

The next step was to convert the file into binary format by using the designed MATLAB program. The file size was sharply reduced to 1,083,943 bytes by this simple action. To this point, the file size was 2.10% of the original raw data file. If the Winzip under the best effects was applied, the file size could further be reduced to 1,082,783, which would be only 2.09% of the original size. This was the best compression ratio we could reach before the wavelet compression technique was applied.

Next, the wavelet compression stated as Step 2 and Step 3 in the framework in Figure 7 was initiated for better compression ratios. The data sets used for wavelet compression were the one after binary format converting to the size 1,083,943 bytes.

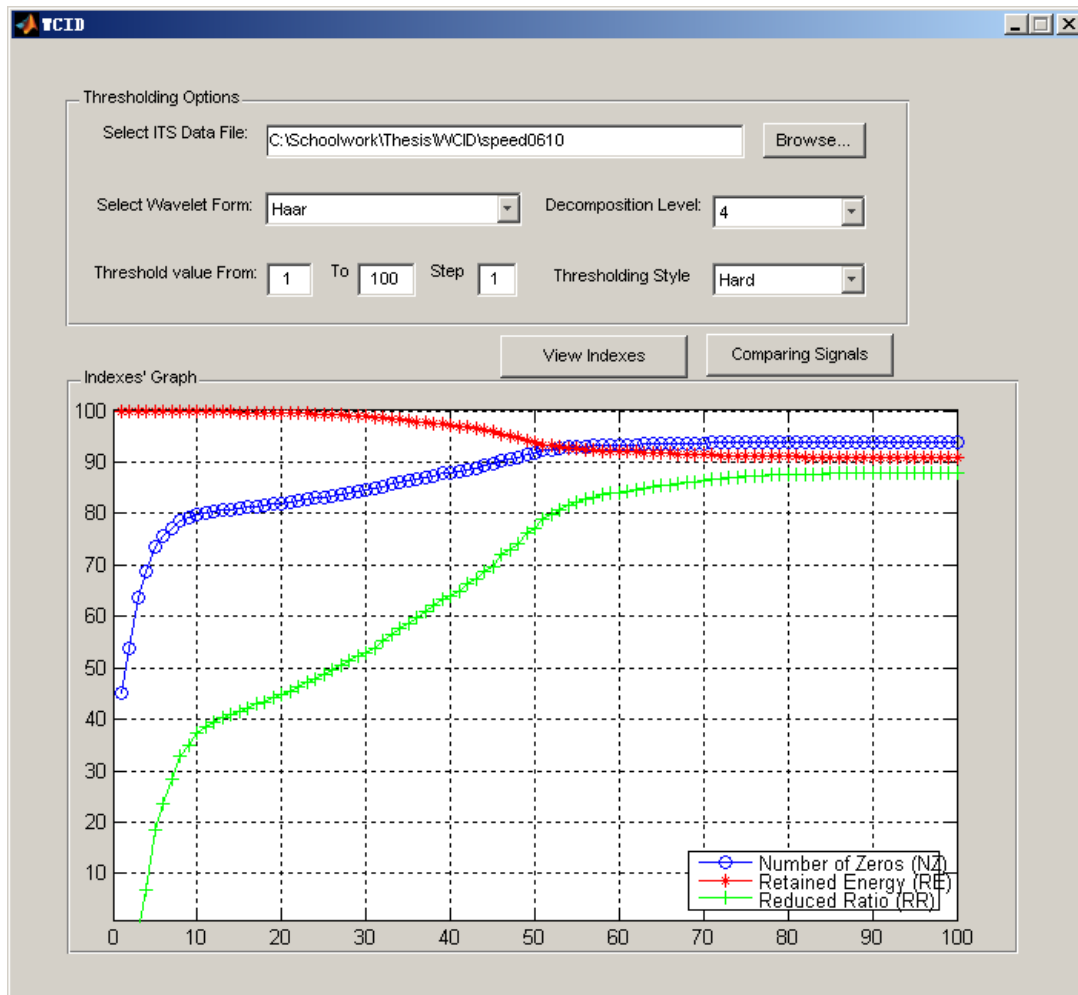
By using the proposed method in the previous sections, the three compression indexes Retained Energy (RE), Number of Zeros (NZ) and Reduced Ratio (RR) were established and used for locating the proper threshold for wavelet compression. A number of wavelet forms and levels were tried for the compression. As a result Haar with decomposition level 4 was selected as the optimal settings.

Figure 19 illustrates the original ITS data and the reconstructed data from wavelet compression. The three traffic variables were listed separately and the associated compression indexes were listed at the right of the plots.



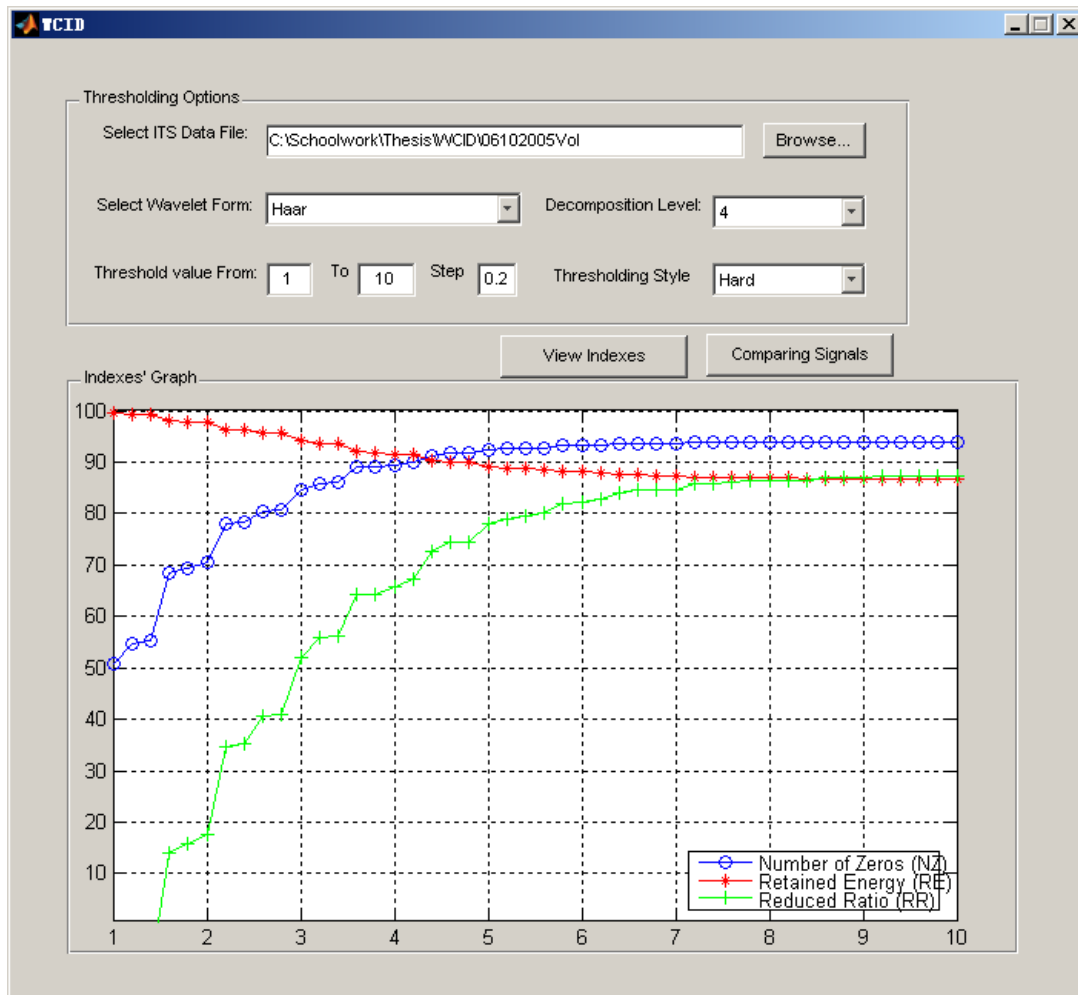
**Figure 19 Reconstructed and Original Data, Compressed by Haar, Level 4**

The optimized threshold that was used for the compression was 24.6, 3.4 and 3.9 for speed, volume and occupancy, respectively; while the Retained Energy was 99.3%, 94.5% and 94.5%, respectively; the Number of Zeros NZ was 83.1%, 86.0% and 84.7%, respectively. Figure 20-22 shows the performance measurement indexes for speed, volume, and occupancy on the given June 10, 2005 data respectively.

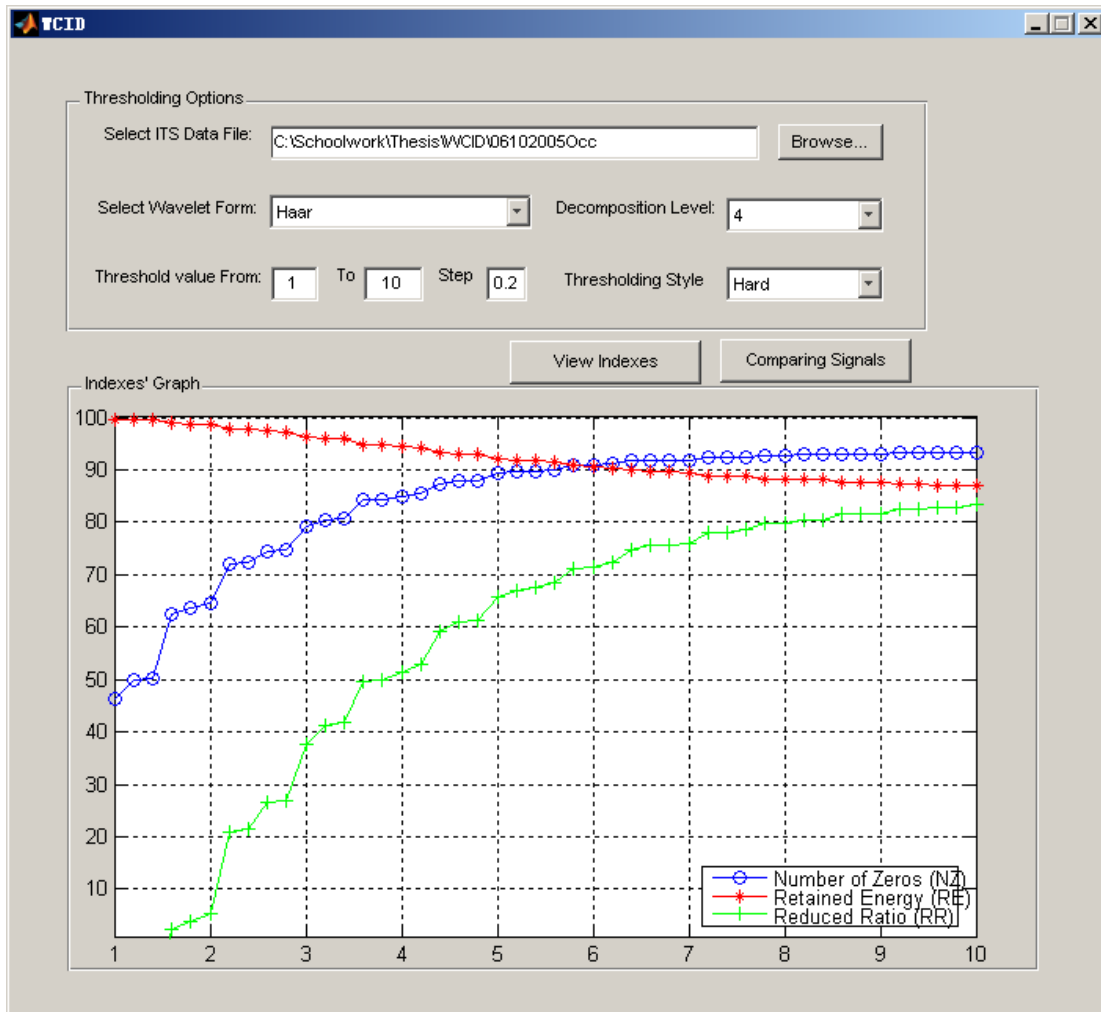


**Figure 20 Performance Measurement Indexes for June 10, 2005 Speed Data**





**Figure 21 Performance Measurement Indexes for June 10, 2005 Volume Data**



**Figure 22 Performance Measurement Indexes for June 10, 2005 Occupancy Data**

The entire compression ratio in this step (for wavelet compression) was 48.6% (more than half) with the wavelet compressed file size being 530,048 bytes. The compression ratio to the original data size was 1.02%.

The wavelet compressed ITS data were zipped by the Winzip with the best effects. The file size after Winzip became 506,014 bytes. This was the final compressed file size for the entire compression process.

Table 3 lists the detailed file size reduction during the entire decomposition. It can be seen that TransGuide compressed this set of ITS data in a ratio of 12.11%; while by the proposed method, the total compression ratio became 0.98%, which is only approximately 8.09% of what TransGuide did. Within this process, the wavelet part compressed almost more than half ( $100\% - 48.6\% = 51.4\%$ ) of its own inputs.

**Table 3 Compression Ratios under Different Steps by TransGuide and by Proposed Method**

*Data Source: June 10, 2005, Server0 lanedata, TransGuide*

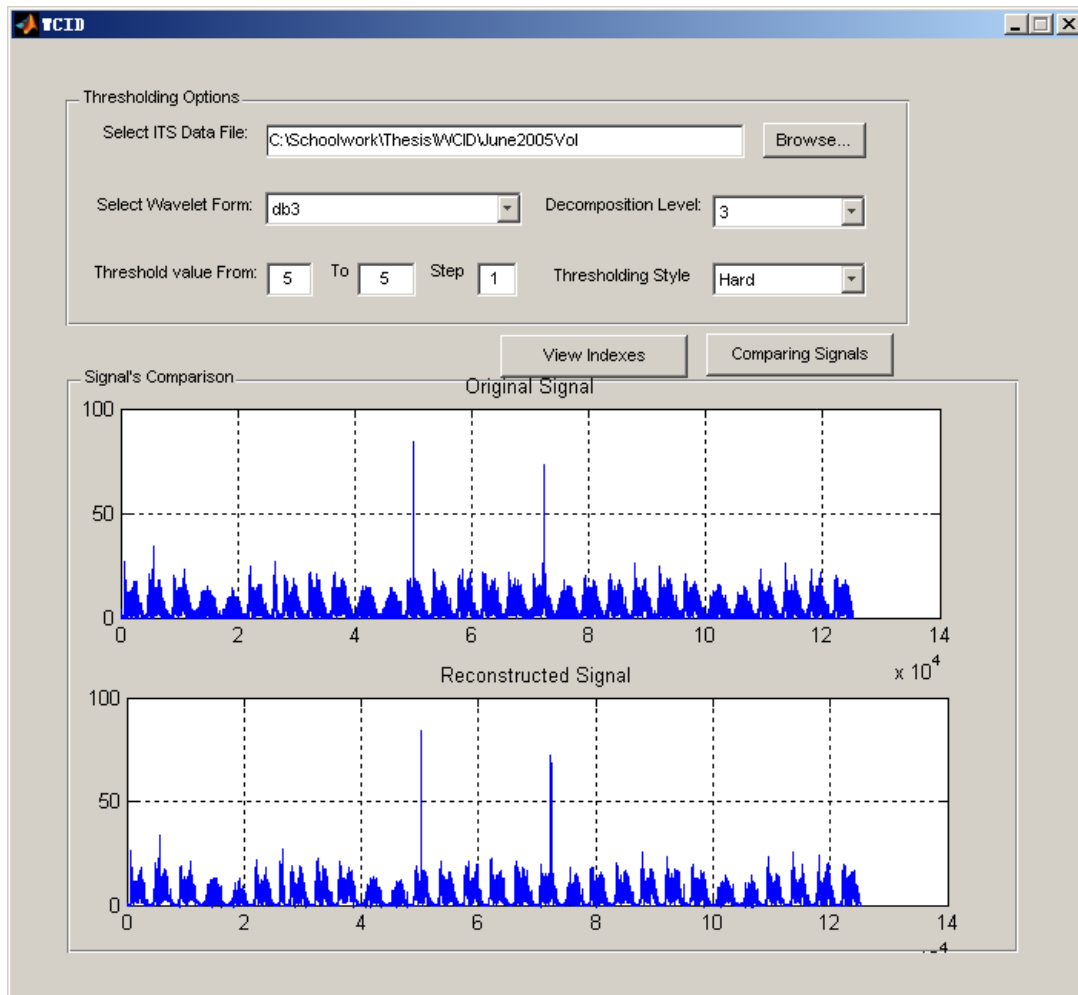
Compression Results	Size(Bytes)	Reduced Size	Compression Rate
<b>From TransGuide</b>			
Raw Text Data	<b>51,715,560</b>	<b>0</b>	<b>100.00%</b>
Raw Text Data Compressed by Gzip	<b>6,261,413</b>	<b>45,454,147</b>	<b>12.11%</b>
<b>Based on the proposed method and algorithm</b>			
Raw Text Data	<b>51,715,560</b>	<b>0</b>	<b>100.00%</b>
Speed + Vol + Occ Text	<b>11,138,730</b>	<b>40,576,830</b>	<b>21.54%</b>
Speed + Vol + Occ Binary	<b>1,083,943</b>	<b>50,631,617</b>	<b>2.10%</b>
[Speed + Vol + Occ Binary +Winzip]	<b>[1,082,783]</b>	<b>[50,632,777]</b>	<b>[2.09%]</b>
Speed + Vol + Occ 1-D DWT Haar Level 4	<b>530,048</b>	<b>51,185,512</b>	<b>1.02%</b>
Speed + Vol + Occ 1-D DWT Haar Level 4 + WinZip	<b>506,014</b>	<b>51,209,546</b>	<b>0.98%</b>

Comparisons between the proposed approach and other ITS data compression methods are hard to make because of the low availability of other methods. However, the approach has not achieved better compression ratio compared with wavelet applications in the image processing area. For instance, the FBI fingerprint compression utilizes WSQ algorithm which is able to produce archival-quality images at compression ratios of about 15:1. In our approach, the wavelet compression step can achieve only roughly 2:1 by itself. This is because the WSQ is applied on images on which the users allow much higher distortion rates.

### 5.3 Data Compression on a Selected Detector Data for a Whole Month

One of the most distinguishing characteristics of traffic data is that it is recursively repeated. If a single detector is studied for a period of consecutive days, the trend will be easily identified: it has the morning peaks and afternoon peaks, and traffic is usually low at nights. It is necessary to take advantage of this well-regulated characteristic especially for transportation planners. For this reason, a selected month, June 2005, was studied to show how the proposed data compression approach could be used on the long-range.

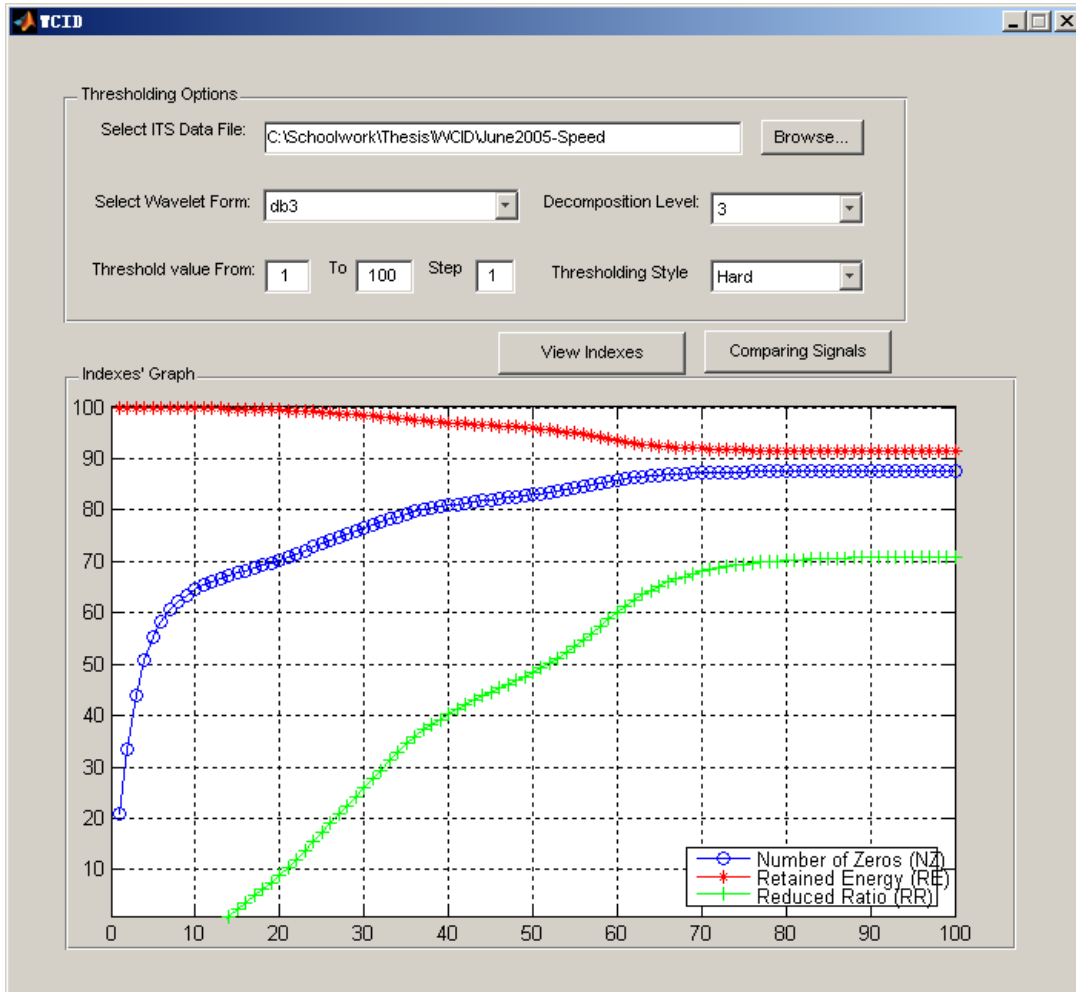
Thirty zipped data files were obtained from TransGuide data server, each for a server (server 0) a day. The total original file size for the entire month ITS data is 187,482,782 bytes, and 1,584,447,236 bytes after decompression. A well-functioned detector with the ID L1-0010E-560.917 was selected to study. The name of the detector indicates that it is on the first lane of the interstate freeway I-10 East at the section with a milepost of 560.917. During the entire month, this detector had 125,216 records, an average of 4174 records per day, or a record per 20.7 seconds. That indicated, this detector had very little “missed” counts.



**Figure 23 One Month's Volume Data Before and After Compression (June 2005)**

Figure 23 shows the wavelet compression volume data comparison with wavelet form db3, level 3. At threshold value 5, the compression can achieve a Reduced Ratio (RR) as high as 64.8%; however, the reconstructed signal keeps the same trend with the original data. Clearly seen are the 30 bumps in the reconstructed signal, each bump with two peaks, representing the

morning and afternoon peak hours. Even the only two abnormal over-50 records were well retained, although they may most likely be erroneous items.



**Figure 24 Performance Measurement Indexes for a Month’s Speed Data (June 2005).**

In Figure 24, the three performance measurement indexes were plotted with different markers. Even for the whole month’s data, the trends of NZ, RE, and RR were still similar to those for the data form of a single day. This is important as the proposed threshold selection algorithm in previous sections was based on the recognition of the trends of NZ, RE and RR as illustrated in Figure 14. If the trends vary with days or the combinations of data sets, then the threshold selection algorithm should be modified.

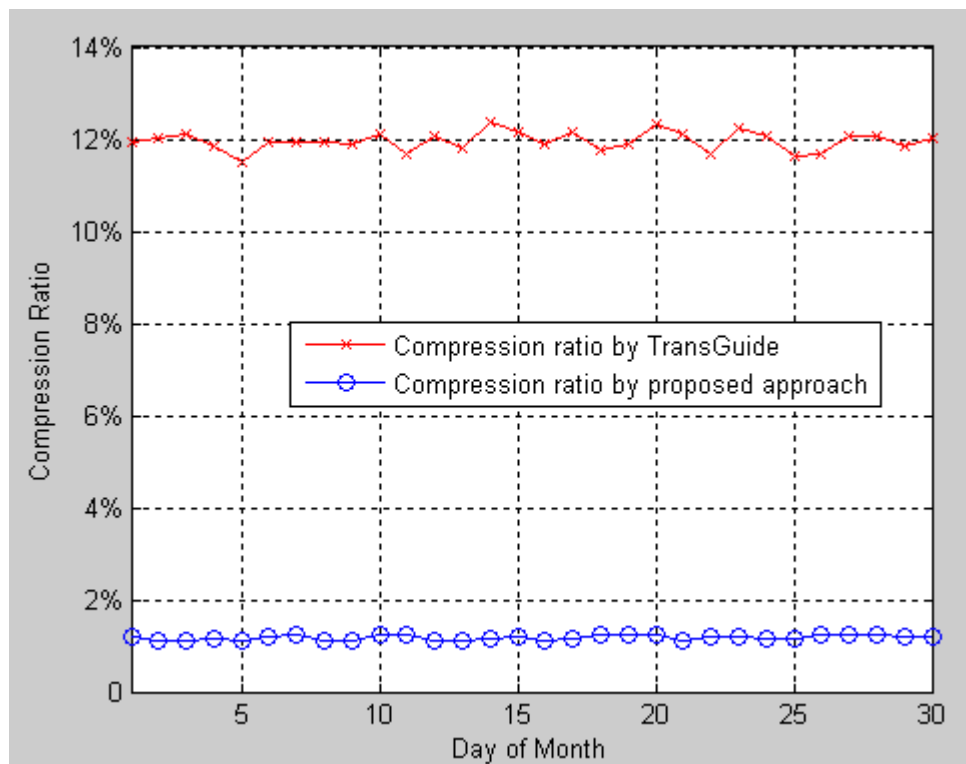
Fortunately, the trends of the three compression indexes related to the threshold remained the same not only for a singer day from all detectors in San Antonio, but also for a combination

of ten days' data from one singular detector. These trends increase the confidence level of the threshold selection algorithm.

#### 5.4 Comparison of Different Ratio, Wavelet Forms and Levels

To test the impact of different wavelet forms and levels on the data compression, the data from June 2005 were compressed using the proposed method, all the thresholds for wavelet compression were based on the proposed algorithm. This means all compressions were balanced between compression ratios and distortions.

The entire compression ratios for the month's data including all steps in the compression framework were retrieved and plotted in Figure 25. For comparison purposes, the corresponding compression ratios by the current TransGuide practice were shown in the same figure as well.



**Figure 25 Compression Ratios Comparison between Current Practice and the Proposed Approach**

**Data are from June 2005, TransGuide Lanedata**

Evidently, the overall compression ratios for the proposed method were much smaller (about 8.1%) than the ones by TransGuide. This means the ITS data can be 91.9% more compressed which will not only save a large space, but increase greatly the transmission rates.

### **Impact of Wavelet Forms and Decomposition Levels**

In order to test the impacts of different selections of wavelet forms and the decomposition levels to the compression ratio, the popular wavelet forms were applied to the compression of the whole day speed data on June 10, 2005, and several typical wavelet forms were selected to present in this paper. The candidate decomposition levels were 2, 4, and 6, while the selected wavelet forms used for testing were db2, db3, db6, Haar, and sym3. The thresholds were determined using the proposed method and algorithm. The three indexes (RE, NZ, and RR) and the compression ratios were compared in Table 4 under five wavelet forms and under three decomposition levels.

**Table 4 Three Indexes and Compression Ration (CR) under Different Wavelets**

Wavelet form	Level	RE*	NZ*	RR*	CR*
<b>Db2</b>	2	99.20%	67.40%	29.80%	70.20%
	4	98.90%	83.40%	47.00%	53.00%
	6	98.80%	87.40%	52.50%	47.50%
<b>Db3</b>	2	99.00%	67.40%	29.00%	71.00%
	4	98.70%	83.70%	46.80%	53.20%
	6	98.60%	87.80%	52.30%	47.70%
<b>Db6</b>	2	98.90%	67.60%	28.20%	71.80%
	4	98.50%	84.00%	46.60%	53.40%
	6	98.40%	88.10%	52.10%	47.90%
<b>Haar</b>	2	99.70%	67.60%	33.60%	66.40%
	4	99.30%	83.20%	48.40%	51.60%
	6	99.20%	87.30%	53.70%	46.30%
<b>Sym3</b>	2	99.00%	67.50%	29.00%	71.00%
	4	98.70%	83.70%	46.80%	53.20%
	6	98.60%	87.80%	52.30%	47.70%

Note: Data used for tests were the one day speeds of all detectors on June 10, 2005;

The thresholds were based on the proposed algorithm;

NZ – number of zeros; RE – Retained Energy; RR – Reduced Ration;

CR – Compression Ratio.

Table 5 presents the statistical analysis results of all the forms and levels. From Table 4 and Table 5, it is seen that there are only slight changes of compression ratios and the three indexes for different wavelet forms. For example, the wavelet form db6 gives the maximum average compression ratio (57.70%), while Haar the minimum (54.77%) This is why Haar is employed in the previous analysis. By the way, for all cases, the Retained Energy (REs) remain very high in value (all >98%), which implies less energy loss.



**Table 5 Statistical Analysis Results**

Comparison index		RE*		NZ*		RR*		CR*	
		Average	Stdev	Average	Stdev	Average	Stdev	Average	Stdev
Wavelet forms	db2	<b>98.97%</b>	<b>0.21%</b>	<b>79.40%</b>	<b>10.58%</b>	<b>43.10%</b>	<b>11.84%</b>	<b>56.90%</b>	<b>11.84%</b>
	db3	<b>98.77%</b>	<b>0.21%</b>	<b>79.63%</b>	<b>10.79%</b>	<b>42.70%</b>	<b>12.18%</b>	<b>57.30%</b>	<b>12.18%</b>
	db6	<b>98.60%</b>	<b>0.26%</b>	<b>79.90%</b>	<b>10.85%</b>	<b>42.30%</b>	<b>12.52%</b>	<b>57.70%</b>	<b>12.52%</b>
	Haar	<b>99.40%</b>	<b>0.26%</b>	<b>79.37%</b>	<b>10.39%</b>	<b>45.23%</b>	<b>10.42%</b>	54.77%	<b>10.42%</b>
	Sym3	<b>98.77%</b>	<b>0.21%</b>	<b>79.67%</b>	<b>10.73%</b>	<b>42.70%</b>	<b>12.18%</b>	<b>57.30%</b>	<b>12.18%</b>
Decomposition levels	2	<b>99.16%</b>	<b>0.10%</b>	<b>67.50%</b>	<b>0.44%</b>	<b>29.92%</b>	<b>2.13%</b>	<b>70.08%</b>	<b>2.13%</b>
	4	<b>98.82%</b>	<b>0.31%</b>	<b>83.60%</b>	<b>0.42%</b>	<b>47.12%</b>	<b>0.73%</b>	<b>52.88%</b>	<b>0.73%</b>
	6	<b>98.72%</b>	<b>0.33%</b>	<b>87.68%</b>	<b>0.42%</b>	<b>52.58%</b>	<b>0.64%</b>	47.42%	<b>0.64%</b>

Note: Data used for tests were one day speeds of all detectors on June 10, 2005;

The thresholds were based on the proposed algorithm;

NZ – number of zeros; RE – Retained Energy; RR – Reduced Ratio;

CR – Compression Ratio.

Big differences between each decomposition level for any chosen wavelet form can also be observed in Table 4 and Table 5. The increase of decomposition levels yield a higher RR, a slight decrease of RE, and the increase of NZ and RR. For example, the overall average compression ratio for Level 2 is 70.08% (with the standard deviation as 2.13%.) while the overall average compression ratio for Levels 4 and 6 are 52.88%, and 47.42%, respectively. Both the standard deviations of compression ratios for Levels 4 and 6 are much smaller (0.73% and 0.64%, respectively.)

These results suggest that the selections of decomposition levels are very important. If possible, higher decomposition levels are recommended. As the differences of decomposition ratios between Level 6 and Level 4 are smaller than the differences between Level 4 and Level 2, probably Level 4 is enough for most of the wavelet forms. This is especially true for Haar as the difference between Levels 6 and 4 are only 5.5%, which is much smaller than the difference between Level 4 and Level 2 (17.8%.) So Haar with Level 4 or Level 6 is the most recommended selection based on the compared pool.

## 5.5 Added Benefit for Wavelet Compression

The nature of the proposed wavelet compression is similar to the nature of signal de-noising; as a result, the compressed ITS data have the de-noised effect on it. That is to say, the “trend” part of the signal is retained, while the noises (basically random abrupt values in a series of smooth data values) are more or less removed. This might not be the desired effect in traffic incident detecting or transportation operation studies, but urban planners would find it helpful because the short-time fluctuations and the random factors are well removed.

Another benefit brought by the proposed wavelet compression is traffic forecasting. Wavelet compression is a form of predictive compression which can be used to estimate the amount of noise in the data set, relative to the predictive function. If a high degree of compression is achieved, then the wavelet algorithm closely approximated the original data set, leaving only small residual values. Turning this around, wavelet compression can be used to estimate the degree of determinism in given traffic data. In other words, it can be told which detector’s data is more predictable by comparing the data from one detector to another. It could help in selecting detectors’ locations for the traffic forecasting process.

## 5.6 Fine Tuning on Signal Details by AR Modeling

The proposed approach achieves data compression by setting the detail levels of the signal to zero where the coefficients of a decomposed signal fall below the pre-set ‘threshold’. We can further tune up this algorithm by modeling the ‘truncated’ signal so the quality of reconstructed data can be improved with a minor overhead of saving only a few parameters.

The characteristics of ITS data determine that the detail level data is likely to have the same amplitude and standard deviation, thus is likely to be stationary. The detail level data does not have seasonality (periodic fluctuations) either. Having met these two conditions, the detailed data after wavelet decomposition could be analyzed and simulated by Autoregressive (AR) Model.

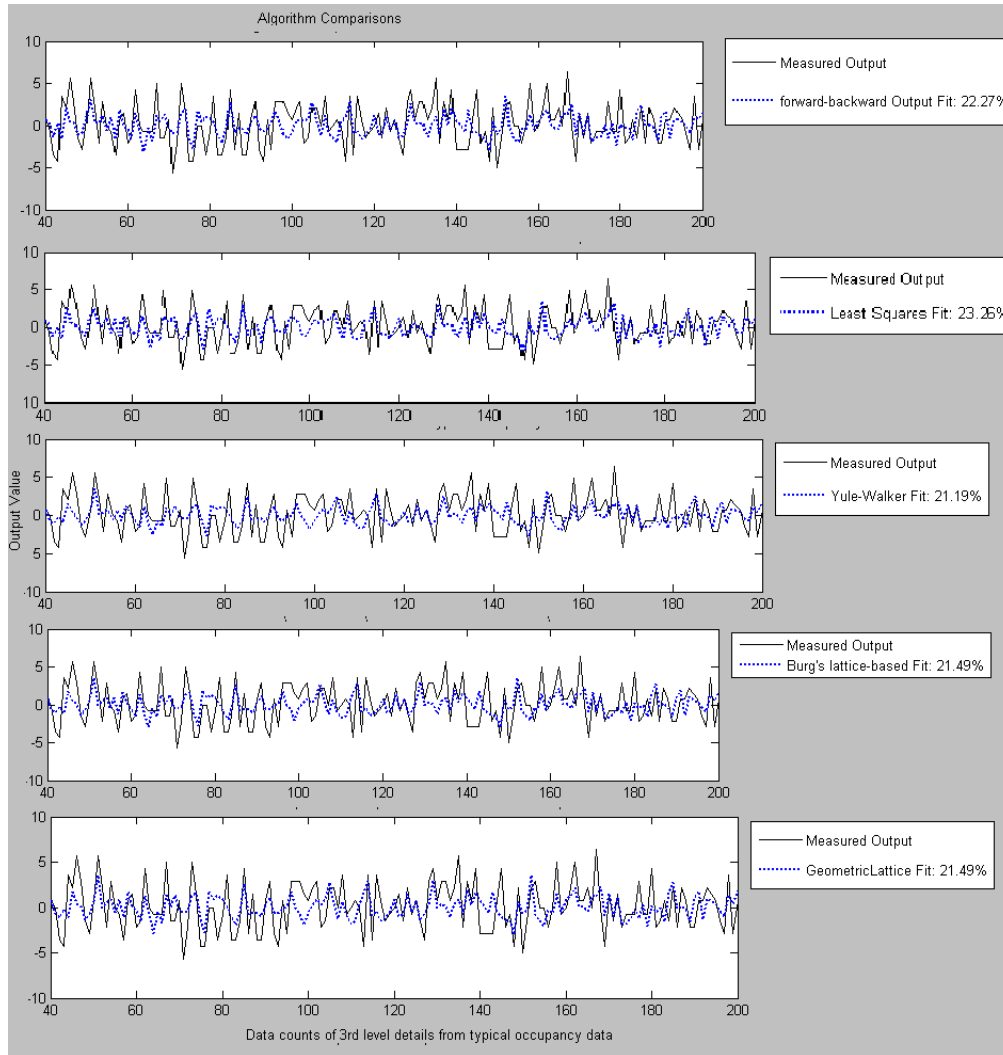
AR Model is usually used in linear prediction formulas that attempt to predict an output  $x[n]$  of a system based on the previous outputs ( $x[n-1]$ ,  $x[n-2]$ ...) and inputs ( $t[n]$ ,  $t[n-1]$ ,  $t[n-2]$ ...). In our case, there is no input because the decomposed ITS signal details that we are trying to model are scalar time series data, therefore the goal is to estimate the model so that it best fits the data.

Various approaches allow us to choose an algorithm from a group of several popular techniques for computing the AR model (Ljung, 2001). The widely used algorithms include the forward-backward approach, the least squares approach, the Yule-Walker approach, Burg's lattice-based method, and the geometric lattice approach. In the forward-backward approach, the sum of a least squares criterion for a forward model and the analogous criterion for a time-reversed model is minimized. In the least squares approach, the standard sum of squared forward prediction errors is minimized. In the least squares approach, the standard sum of squared forward prediction errors is minimized. The Yule-Walker equations, formed from sample covariances, are solved for the Yule-Walker approach. The lattice filter equations are solved using the harmonic mean of forward and backward squared prediction errors for the Burg's lattice-based method. Finally, the geometric lattice approach is similar to Burg's method, except the geometric mean is used instead of the harmonic one. A detailed introduction of autoregressive algorithms can be found in (Ljung 1994). In this section, a comparison on algorithms has been made in order to determine the method that best fits.

The algorithm comparison computes the output  $y_h$  that results when the AR model  $m$  is simulated. The percentage of the output variation is explained by the formula,

$$\text{fit} = 100 \times \frac{1 - \text{norm}(y_h - y)}{\text{norm}(y - \text{mean}(y))} \quad (19)$$

The occupancy data were selected to carry out the comparison. First, as a typical day, the occupancy data on June 1<sup>st</sup>, 2005 was decomposed with Haar Wavelet Level 3. Then, 200 data points from the third level detail was selected as the basic dataset to run the comparison on the model order of 40. This third level of detail has very small values so nearly all of the data points fall within the threshold and thus will be set to zeros in our proposed wavelet compression approach. For each autoregressive method, the measured output (real data) and the model predicted output were plotted in a figure and the fit was calculated.



**Figure 26 The Autoregressive Model Algorithm Comparison  
 (Based on June 1st 2005 Occupancy Data, Harr Wavlet  
 Decomposition Level 3, 3rd Level Detail)**

As can be seen from Figure 26, the Least Squares method has the highest fit 23.26%, while the lowest fit results were from the Yule-Walker approach, which gives a 21.19% fit level. Least Squares method is thus selected to conduct the autoregressive modeling for this research. The notation AR(p) refers to the autoregressive model of order p, it can be written

$$X_t = c + \sum_{i=1}^p \phi_i X_{t-i} + \varepsilon_t \quad (20)$$

where,  $\phi_1 \dots \phi_p$  are the parameters of the model,  $c$  is a constant and  $\varepsilon_t$  is an error term added to the model.

The constant  $c$  can be omitted by a careful model designing process, so the formula could be rewritten as

$$A(q)X(t) = \varepsilon_t \quad (21)$$

where,  $A(q)$  is  $1 - \sum_{i=1}^p \phi_i X_{t-i}$ .

Now, the goal is to estimate the order  $q$  and the AR coefficients  $A(q)$  so as to meet the criterion: to minimize  $\varepsilon_t$ , by the preset algorithm.

The best fit model order  $q$  is found by calculating and comparing the Level 2 Norm Ratio (also called Retained Energy), namely, the Level 2 Norm Ratio of the reconstructed detail signal to the original one. This Level 2 norm value for a one-dimensional signal can be written as:

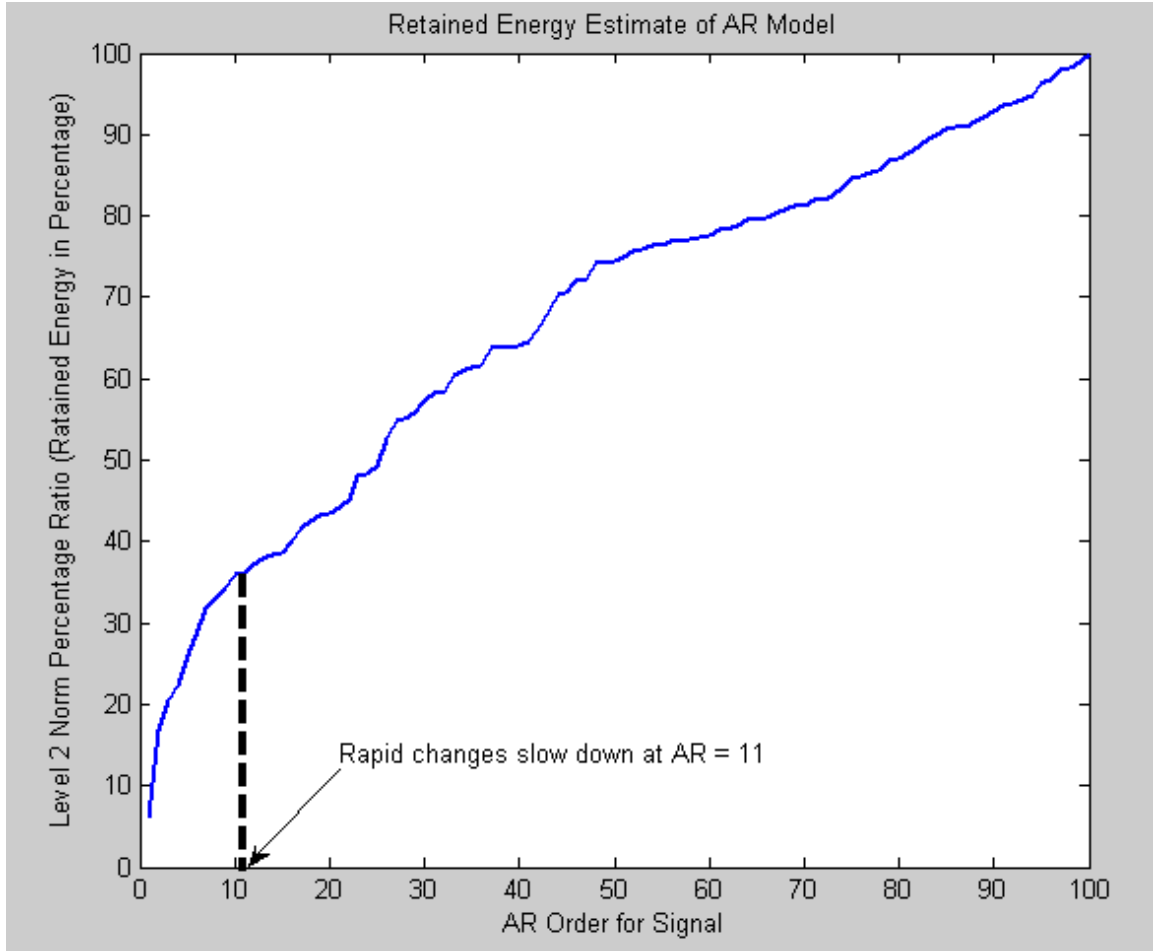
$$Norm(x) = \sqrt{\sum_x |x^2|} \quad (22)$$

Thus, the Level 2 Norm Ratio can be expressed as:

$$Level2NormRatio = 100 \times \frac{norm(reconstructed)}{norm(original)} \quad (23)$$

It is clear that the goal is to make the Level 2 Norm Ratio as big as possible (as close to 100 as possible) so that we get better compression with the model order evaluation algorithm being given. By plotting the Level 2 Norm Ratio, it is found that the increase of Retained Energy slows down dramatically after a certain model order. This order is then chosen to be our model order; orders higher will not give any appreciable increase in the retained energy while increasing the number of AR coefficients (or reflection coefficients) to be archived. The Level 2

Norm Ratio (Retained Energy) as a function of  $q$  (model order) is shown in the following graph:



**Figure 27 The Variance of the Residual Changes vs. Autoregressive Model Order (Based on June 1st 2005 Occupancy data, Harr Wavlet Decomposition Level 3, 3rd level detail)**

As Figure 27 shows, an order of 11 would be enough and increasing the order doesn't change the ratio percentages unless for very high orders.

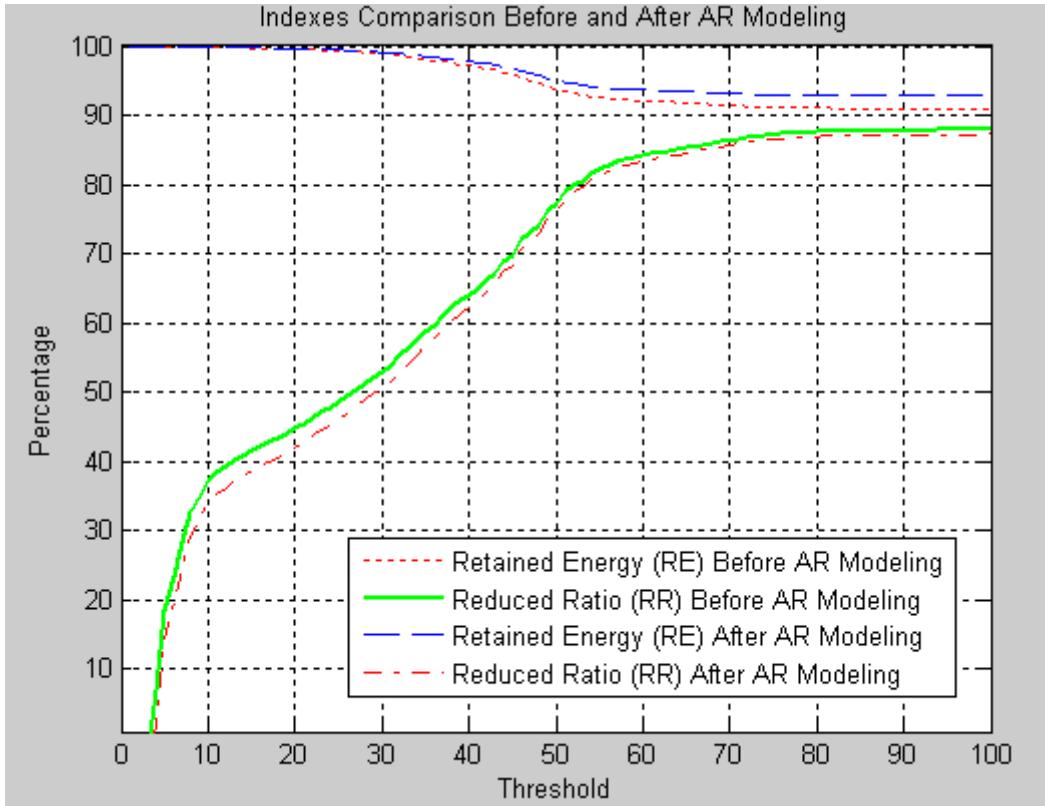
After using AR Model to simulate the selected section of data signal, another problem occurred on how to properly divide the data signal into sections so that the AR Model could be used on each section. Periodogram has been used in time series analysis to provide a check on the randomness of a series, where we consider the possibility that periodic components of unknown frequency may remain in the series (Box and etc, 1994). The representation of

periodogram is used to determine periodicities in the give signal data sets, thus periodogram could be utilized to break the signal down into sections according to the periodicity characteristics. The periodogram for a sequence  $[x_1, \dots, x_n]$  is given by the following formula:

$$S(e^{j\omega}) = \frac{1}{n} \left| \sum_{l=1}^n x_l e^{-j\omega l} \right|^2 \quad (22)$$

This expression forms an estimate of the power spectral density (PSD) of the signal defined by the sequence  $[x_1, \dots, x_n]$ . By this approach, the data signal that may not be stationary in general is to be divided into stationary segments with different lengths. This is called adaptive segmentation; and how to find the boundaries for that segmentation is beyond the scope of this thesis. The other method, which is simpler, is to divide the signal into fixed length short-duration segments and calculate the FFT of each segment and show them in spectrogram. In this technique, the duration of the segments must be short enough to ensure that the signal remains stationary within that duration. The fixed length segments method is used in this thesis to divide the signals, then each segment is simulated by the aforementioned AR Modeling.

To have a comparison of the compression effect before and after AR Modeling, the June 10<sup>th</sup>, 2005 TransGuide occupancy data was once again selected. These data were first decomposed by Haar Level 4. Earlier we did the threshold and reconstruction and then all four levels of details were AR modeled and reconstructed again.



**Figure 28 Performance Indexes Comparison Before and After AR Modeling**

**Note: Data from June 10, 2005 Speed Data**

Figure 28 illustrates the changes of performance indexes before and after AR Modeling. It is very clear that a better Retained Energy can be achieved with the cost of only a small decrease of Reduced Ratio. As can be seen from Figure 28, the Retained Energy was improved by around five percent, while the Reduced Ratio was down by only 2 percent.



## CHAPTER 6

### SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

#### 6.1 Conclusions

In this research, the wavelet incorporated ITS data compression method has been proposed, and then a MATLAB GUI program with the name WCID has been developed to facilitate the compression tests; finally a case study on TransGuide ITS data was put into play and a final compression ratio of less than one percent on the trade-off threshold value shows that the proposed approach is practical.

Since the desired wavelet compression is a lossy algorithm, the balancing between the compression ratio and the signal distortion is exceedingly important. During the compression process, the determination of the threshold is the key issue that affects both the compression ratio and the signal distortion.

In this research, an algorithm is proposed that can properly select the threshold by balancing the two contradicted aspects. Three performance indexes RE, NZ and RR are constructed and the relationships between the three indices and the threshold are identified.

Impact analysis of wavelet forms and decomposition levels to the compression ratios shows that there is not too much difference in the selection of wavelet form. Wavelet form Haar can provide a relatively smaller compression ratio. However, decomposition levels have significant impact on the decomposition. Higher decomposition levels normally yield better compression ratios for the same threshold values.

Finally, the threshold selection algorithm can be further tuned utilizing the Autoregressive model so that the quality of reconstructed data can be improved with a minor overhead saving of only a few parameters. After comparison between several methods, Least Squares method is selected for the Autoregressive model. The case study indicates that a better Retained Energy can be achieved with the cost of only a small decrease of Reduced Ratio.

#### 6.2 Recommendations

Data multiform is crucial for this wavelet incorporated ITS data compression research study. This is because TMCs are currently using incompatible data formats in storing the

collected ITS data. The traffic data variables, the layout of lines and columns in the data file, and the time interval differ from one TMC to another. Future study should test data sets from a various number of TMCs.

ITS data quality control could be incorporated in the wavelet compression approach. During the research, it is found that the result of the compressed ITS data have the de-noised effect due to the nature of the proposed wavelet compression which is similar to signal de-noising. The “trend” part of the signal are retained, while the abnormal data, usually treated as noise in wavelet decomposition, are more or less removed from the result. Considering these abnormal data are usually erroneous or inaccurate measurements, data quality control could be well included by recalculating those data in the wavelet decomposition.

Finally, it is recommended that the compression processing speed should also be taken into consideration in order to meet the need of the increasingly surging ITS data. With more highway infrastructure put into use, the ITS data increase rate becomes overwhelming. A practical solution on ITS data compression must run faster on prevalent computers than the data-generating speed to be feasible.

## REFERENCES

- Box, G. E. P, Jenkins G. M, and Reinsel G. C. (1994). Time Series Analysis Forecasting and Control, Third Edition. Published by Prentice Hall, Inc. Upper Saddle River, New Jersey.
- Cleary, J. G., T. C. Bell, and I. Witten. (1990). Text Compression. Advanced Reference Series, Prentice Hall. Upper Saddle River, New Jersey.
- Cohen, J. K., (1992). Wavelets - a new orthonormal basis: Colorado School of Mines.
- Dahlgren H., Turner S., and Garcia R. C. (2002). Collecting, Processing, Archiving and Disseminating Traffic Data to Measure and Improve Traffic Performance. Transportation Research Board 85th Annual Meeting CD, Paper No: 06-0719. Washington D.C.
- Daubechies, I., (1992). Ten lectures on wavelets: Philadelphia, SIAM
- Debra A. Lelewer and Daniel S. Hirschberg. Data Compression. Computing Surveys. Vol. 19 No. 3, 1987, pp. 261-297. Reprinted in Japanese BIT Special issue in Computer Science, 1989, pp. 165-195. URL: <http://www.ics.uci.edu/~dan/pubs/DataCompression.html>. Retrieved: July 29, 2005.
- Guo H. and Jin J. (2006). Travel Time Estimation Using Correlation Analysis of Single Loop Detector Data. Transportation Research Board 85th Annual Meeting CD, Paper No: 06-0639. Washington D.C.
- Hellinga, B. R. (2001). Improving Freeway Speed Estimates from Single-Loop Detectors. Journal of Transportation Engineering Volume 128, Issue 1, pp. 58-67
- Klimenko, S, Mitselmakher, G, Sazonov. (2002). A Lossless Compression of LIGO Data. LIGO-T000076- 00- D 08/7/2000.
- Liu, H. X., He, R., Tao, Y., and Ran B. (2002). A Literature and Best Practices Scan: ITS Data Management and Archiving. Wisconsin Department of Transportation Project Final Report. Project No.: 0092-02-11. Wisconsin.
- Ljung, L. (2001). System Identification Toolbox for Use with MATLAB: User's Guide Version 2.1. The Math Works, Inc. Natick, MA.
- Ljung, L., and Glad, T. (1994). Modeling of Dynamic Systems, Prentice Hall, Englewood Cliffs, N.J. ISBN 0135970970
- Marven, C., and G. Ewers. (1996). A Simple Approach to Digital Signal Processing. John Wiley and One, In., New York.
- Middleton, D., Jaskek, D., and Parker, R. (1999). Evaluation of Some Existing Technologies for Vehicle Detection. Project Summary Report 1715-S. Texas Transportation Institute.
- Misiti M., Y., Oppenheim, G., & J. M. Poggi. (2001). Wavelet Toolbox for Use with MATLAB: User's Guide Version 2.1. The Math Works, Inc. Natick, MA.
- The Moving Picture Experts Group. The MPEG Home Page. MPEG Website. Retrieved March 18th, 2009 URL: <http://www.chiariglione.org/mpeg/index.asp>
- National Institute of Standards and Technology. (2006). Engineering Statistics Handbook. Chapter 6. Process or Product Monitoring and Control. Retrieved March 3rd, 2006 URL: <http://www.itl.nist.gov/div898/handbook/index.htm>
- National Transportation Library. (2008). Intermodal Surface Transportation Efficiency Act of 1991 – Summary. Title VI Research Part B IVHS Retrieved December 14th, 2008 URL:[http:// http://ntl.bts.gov/DOCS/ste.html](http://http://ntl.bts.gov/DOCS/ste.html)
- Office of Highway Policy, Federal Highway Administration (2005). ITS as a Traffic Data Resource. Retrieved Jan 28th, 2006. URL: <http://www.fhwa.dot.gov/policy/ohpi/hss/presentations/adus.htm>

- PeMS (2009). Freeway Performance Measurement System Retrieved Mar 03rd, 2006. URL: <https://pems.eecs.berkeley.edu/>
- Qiao, F., L. Yu and X. Wang. (2004) Double-Sided Determination of Aggregation Level for Intelligent Transportation System Data. Transportation Research Record, No. 1879. TRB, National Research Council, Washington D.C.
- Salomon, D., G. Motta, and D. Bryant, (2007) Data Compression: The Complete Reference. Spring-Verlag London.
- Souleyrette, R., D. Plazak, T. Strauss, and S. Andrie. (2001) Applications of State Employment Data to Transportation Planning. Transportation Research Record 1768, 2001, pp. 26-35.
- Texas Department of Transportation. Traffic Statistics. TransGuide Website. URL: <http://www.transguide.dot.state.tx.us/docs/statistics.html>. Retrieved: July 29, 2005.
- Turner, S.M. (2001). Guidelines for Developing ITS Data Archiving Systems. Research Project Title: Developing Guidance for Sharing Archived/Warehoused ITS Data. Texas Transportation Institute.
- Turner, S.M., W.L. Eisele, B.J. Gajewski, L.P. Albert, and R.J. Benz. (1999) ITS Data Archiving: Case Study Analysis of San Antonio TransGuide® Data. Report No. FHWA-PL-99-024. Federal Highway Administration, Texas Transportation Institute, College Station, Texas.
- US Department of Transportation (2009). ITS Applications Overview. USDOT Website Retrieved Mar 2nd, 2009 URL: <http://www.itsoverview.its.dot.gov/faq.asp>
- Washington State Department of Transportation. Automated Data Collection & Processing Section. WSDOT Website. Retrieved Feb 3rd, 2006. URL: <http://www.wsdot.wa.gov/mapsdata/tdo/adc.htm>
- Weijermars W. A. M. and Van Berkum E. C. (2006). Detection Of Invalid Loop Detector Data In Urban Areas. Transportation Research Board 85th Annual Meeting CD, Paper No: 06-0719. Washington D.C.
- Yu, L., X. Liu, and X. Chen. (2008). Data Compression for Emitter Location Finding in Sensor Networks. IEEE Proceedings of Information Technology: New Generations, 2008, pp. 1210-1215.
- Zhang X., Nihan N. L., and Wang, Y. (2005). An Improved Dual-Loop Detection System for Collecting Real-Time Truck Data. Transportation Research Board 84th Annual Meeting CD, Paper No: 05-2210. Washington D.C.