ERG4920CM Thesis II

Keyboard Acoustic Triangulation Attack

BY

Au Hiu Yan Fiona (03634573)

A FINAL YEAR PROJECT REPORT
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF BACHELOR OF INFORMATION ENGINEERING
DEPARTMENT OF INFORMATION ENGINEERING
THE CHINESE UNIVERSITY OF HONG KONG

May, 2006

Abstract

It is proven that typing with keyboards can be easily hacked by listening to the sound emanated. Previous researches show that a key can be distinguished by the frequency characteristics of a keystroke. Although the old attacks have an accuracy rate up to 80%, they can be defended by keys with similar sounds. Moreover, the algorithms are complex.

For keys with similar sounds to be distinguishable, we develop a new attack approach, acoustic triangulation attack. Simplicity and easy computation are the advantages. A key can be found by computing the differences in arrival times of a sound wave at two microphones. A series of experiments is performed. Surprisingly, by choosing 3 to 5 keys from a set of 104 keys, the recognition rate can be up to 80% with a 5-minute computation. With similar technique, we are able to identify all the keys in a keyboard. Since the experimental tools and softwares can be easily obtained from shops and the Internet, it is possible for non-technical people to perform the attack. Therefore, keyboard acoustic triangulation attack is ever more threatening.

Table of Contents

Chapter 1 Introduction

"It is only a matter of time before criminals begin using similar techniques!"
--- Bruce Schneider, chief technology officer at Counterpane Internet Security and the author of Applied Cryptography.

Keyboards are the most common inputting devices. Specialized keyboards, such as keypads, are widely used for specialized data input. It is significant that keypads are usually used to protect a security system.

In Hong Kong, keypads are widely used in banking industries and housing estates. Side channeling keypad data may be a potential threat to the security systems. According to a research conducted by Berkeley Doug Tygar [2], a researcher of the University of California, clicks and clacks from a computer keyboard can be transposed into a startlingly accurate transcript.

Imagine that you are typing your password at an automatic telling machine. The ATM produces similar electronic sounds while you are pressing the keys. You think nobody knows what you are typing since the sounds are not distinguishable to human ears. However, you have made a big mistake. In fact, hackers use the mechanical sounds emanated to guess your password!! With some microphones, a computer and a sound-processing software, the keys pressed can easily be retrieved.

The fact that an attacker can use this acoustic emanation from keypads to collect confidential information has been a great concern in security and privacy communities. In this project, we stimulate the keyboard acoustic triangulation attack. Forerunners determine different keys by their frequency components. We introduced a new attack using time difference approach. This approach works even when the mechanical sound is veiled by electronic sound. This implies keyboard emanation attacks are more serious than previous work suggests. Our ultimate goal is to find out ways to prevent the attack.

Below, Chapter 2 briefly reviews previous keyboard acoustic emanation attacks. Chapter 3 presents the details of our attack. It is divided into four parts. The first part presents the progress report. In the second part, we generalize the procedures of a keyboard acoustic attack. In the third part, the details of our experiments, including computational methodologies and experimental procedures, are presented. Finally, the experimental results are discussed. The performance of the computational methodologies is evaluated. We discuss future development in Chapter 4 and conclude with a summary in Chapter 5.

Before going into the project details, let us first define some keywords and state our contribution.

1.1 Side Channel Attack

A side channel attack is an uprising security issue in cryptography. It refers to any attack which gains the information from the physical implementation on a cryptosystem, rather than the theoretical weaknesses in algorithms. [10]

Usually, an attacker is not required to equip a thorough technical understanding in the internal operation of a system in order to perform a side channel attack. Information can be extracted from the physical effects caused by the operation of the cryptosystem, for example, the timing information, power consumption or even sound. Therefore, this kind of attacks is non-invasive. It does not require a physical intrusion into the system.

Another reason that hackers prefer to use side channel attack is that it is generally inexpensive. The cost of recording equipments is quite cheap. For example, to perform an acoustic side channel attack, the only additional hardware required is a microphone.

Some common types of side channel attacks are timing attacks, architectural side-effect attacks, power monitoring attacks and acoustic cryptanalysis. Keyboard acoustic attack is also a kind of side channel attack.

1.2 Acoustic Cryptanalysis

Acoustic cryptanalysis is a type of side channel attack which extracts information unintentionally exploited from sounds produced during a computation or input-output operation. It is a new research area of applied cryptography that has gained more and more interest since the mid nineties.

According to a book “Spycatcher”, written by a former MI5 operative Peter Wright [21], similar attack technique had already been used as early as in 1956. By that time, the attack “ENGULF” is used against the Egyptian Hagelin cipher machines.

Today, hackers collect sounds produced by a computer system during computations or input-output operations. They analysis them by implementing secure mathematical algorithms on the acoustic signals. Experiments show that valuable and distinguishable information can be extracted from those sounds.

An example of acoustic cryptanalysis is the experiment conducted by Adi Shamir and Eran Tromer in 2004 [14]. They demonstrated that it may be possible to conduct a timing attack against a CPU by analyzing the variations in its humming noise.

1.3 Acoustic Triangulation Attack

Triangulation is a process of finding the distance of a point using the concept of a triangle. The distance can be treated as a side of a triangle. It is calculated by measuring angles and sides of a triangle. Triangulation is a common technique for locating an object. It is often used in surveying, navigation, metrology and astrometry.

Acoustic triangulation attack means finding the location of an object based on the measurement of acoustic waves generated by a keystroke. [22] (Figure 1.1) By detecting and measuring the differences in arrival times of the sound wave at two microphones, the impact location can be found uniquely.

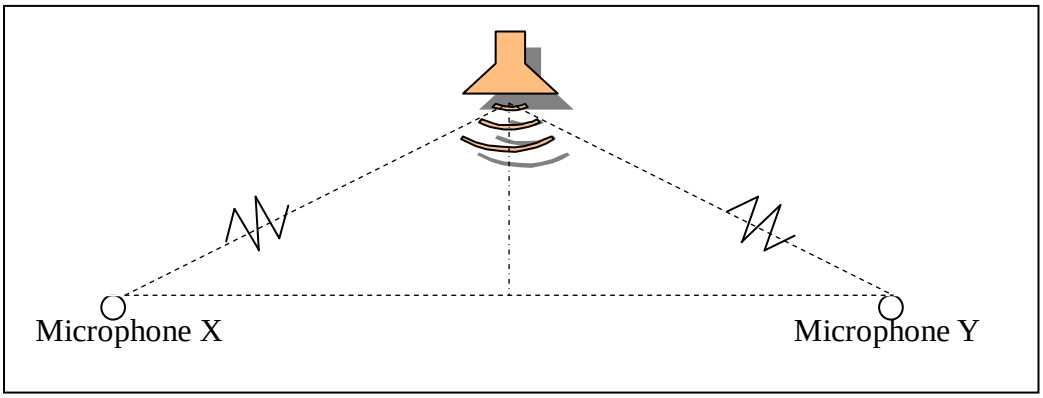

Figure 1.1 An acoustic triangulation attack

1.4 A Keyboard

A mechanical keyboard consists of a number of keys and a circuit board. (Figure 1.2) On the keyboard there are many rubber buttons. Each key corresponds to a button on the board.

Figure 1.2 A typical DELL keyboard

Each key in a keyboard (Figure 1.3) generally consists of three parts: [1]

a) A head -- This is the part where we contact with the key.

b) A bottom rubber part -- The dome-shaped rubber is used to make contact with an electrical switch corresponding to the key.

c) An intermediate plastic part in between the head and the rubber

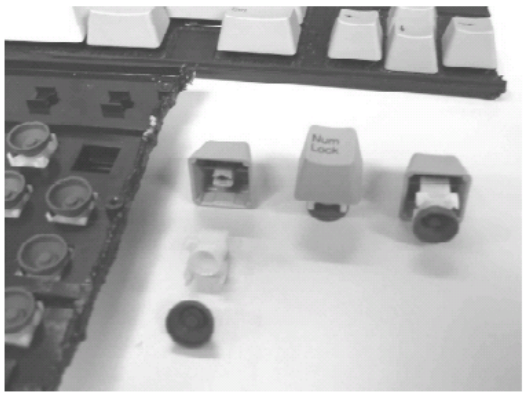

Figure 1.3 The architecture of a mechanical keyboard

When a key is pressed, the dome-shaped rubber is squeezed. It then pushes the electrical switch and closes the circuit. Upon strike, the keyboard plate vibrates and produces a sound.

1.5 Comparison of the Three Approaches

Before going into details, let us understand the difference between the previous attack and the new attack, acoustic triangulation attack.

A simple comparison between the three methods is shown in Figure 1.4. Detail descriptions of the previous attack will be discussed in Chapter 2. The new approach will be discussed in Chapter 3.

	Cost	**Computation Difficulty**	**Algorithm**	**Accuracy**
Acoustic Emanation Attack by D. Asonov and P. Agrawal	Cheap (1 mic)	Difficult	Complex	High
Revised Acoustic Emanation Attack by L. Zhuang, F. Zhou, and J. D. Tygar	Cheap (1 mic)	Very difficult	Very Complex (with dictionary)	Very High (for long type-text)
Acoustic Triangulation Attack	Cheap (2 mics)	Easy	Simple	High

Figure 1.4 Comparison between the previous attacks and the new attack

1.6 Our Contribution

The previous keyboard acoustic attacks make use of the frequency domain characteristics in the keystroke sound. Although the cost of the attack is cheap, it requires a good understanding in feature extraction from similar sounds and classification probability. This attack also involves complex and difficult computations.

To simplify the situation, we propose a new keyboard acoustic attack, ***Acoustic Triangulation Attack***. We measure the time difference in which a keystroke sound travels to two microphones. Different keys may result in different time delay.

Comparing with the previous ones, our attack is easier to compute. Attackers are not required to understand the complex mathematics behind the scene. They just need to compare the waveforms of the sounds collected.

Moreover, the previous attacks can be defended by using keyboards with keys producing similar sounds or electronic sounds. [1] However, under this situation, our attack still works! No computation in frequency domain is required in the new attack. Therefore, we can still get the information in no pain.

The success of this project will be a key milestone in keyboard acoustic side channel attack. This triangulation attack can be easily done by non technical people. The danger for a keyboard to be eavesdropped increases greatly. It is surely a noteworthy security issue.

The new attack will be discussed in detail in Chapter 3.

Chapter 2 Previous Attacks

Researches on speech recognition have been conducted over 70 years. The characteristic of sound in frequency domain is proven to give much vulnerable information about a sound. This piece of information can be used to distinguish between speeches.

Applying the mature speech recognition technology, researches begin their researches on keyboard acoustic attacks in recent years. They investigate the frequency components of the sound emanated by a keystroke. With the use of neural networks, they distinguish keys from each other.

Since keyboard acoustic attacks are relatively new in the security world, we can only find two papers in this field. In this section, we will have a brief review on these papers. And based on these papers, we conduct our research.

D. Asonov, and R. Agrawal, "Keyboard Acoustic Emanations" [1]

Asonov and Agrawal are the first researchers who draw our attention to the sounds emanated from keyboard keys. They assume that the sound of clicks differ slightly from key to key. Thus, they conducted an experiment to differentiate the sound of clicks.

Firstly, they investigate why the clicks produced different sounds. It is found that knocking at different parts of a mechanical keyboard plate produce different sounds. It is because striking a key at different locations provides different timbres.

Next, they recorded the training and the test set for 30 keys on a single PC keyboard. In the experiment, they found out that the fast Fourier Transform values of the press peaks give significant variation for perceptually similar sounds. Then, the frequency distributions were passed into a neural network for classification. Through experimenting with several sets of data, the best recognition rate can be achieved by including the entire active interval in the feature extraction. Moreover, the 300-3400 Hz interval is found to be the most informative.

Using Asonov and Agrawal's algorithm, the accuracy rate for a key to be recognized correctly is approximately 79%. Although different keyboards produce different sounds, the sound emanated is unique for each key. It is also proven that keys can be recognized regardless of the keyboard type and the typing style.

L. Zhuang, F. Zhou and J. D. Tygar, "Keyboard Acoustic Emanations Revisited" [2]

Since Asonove and Agrawal's attack requires neural network training with text-labeled samples, this suggests a limited attack because attacker needs to obtain training sample of significant length beforehand. Zhuang, Zhou and Tygar improve the attack using a combination of machine learning and speech recognition techniques.

In their approach, they used only a sound recording of a user typing to build a keystroke recognizer. They divided the keystrokes collected into K classes, where K was slightly larger than the number of keys on the keyboard. Keystrokes of the same key were sometimes placed in different classes and conversely keystrokes of different keys could be classified in the same class. The letter with the highest probability for each keystroke was picked to yield the best estimation.

For randomly generated sequences, such as passwords, the output of the keystroke classifier for each keystroke is a set of posterior probabilities:

$$P(\text{this keystroke has label } \boldsymbol{i} \mid \text{observed-sound}), \quad \boldsymbol{i} = 1, 2, \ldots, 30.$$

Given the conditional probabilities, the probabilities for all sequences of keys can then be calculated. By sorting the sequences from most probable to least probable, an attacker can try and find the real password.

In addition, Zhuang, Zhou and Tygar apply a spelling and grammar checking and a feedback-based training (Figure 2.1) over the intermediate result to improve the accuracy. Taking a 10-minute sound recording of a user typing English text using a keyboard as the input, up to 96% of typed characters can be recovered. A more shocking result is that 90% of 5-character random passwords using only letters can be generated in fewer than 20 attempts by an adversary.

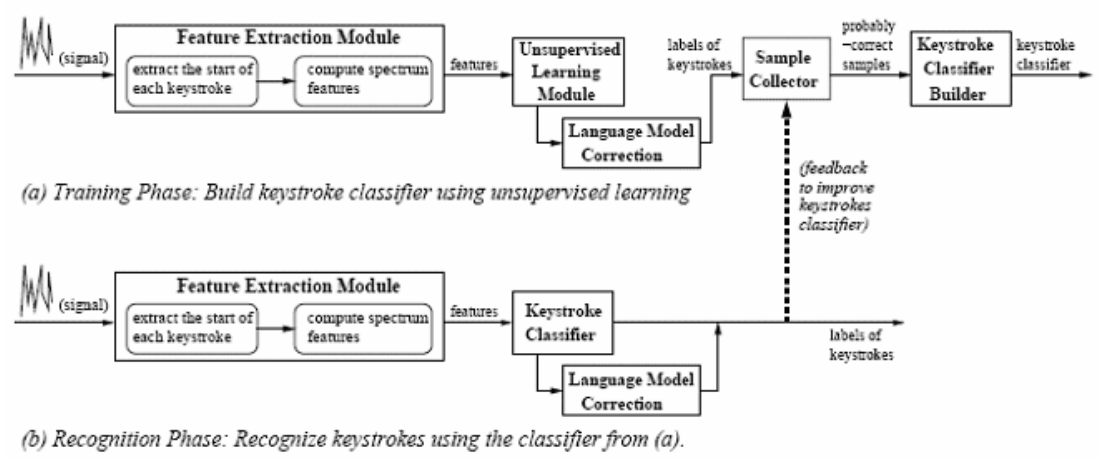

Figure 2.1 Overview classification phrase in the attack of L. Zhuang, F. Zhou and J. D. Tygar

Chapter 3 Project Description

This chapter is divided into five parts. Firstly, the progress of this project is reported. Secondly, the generalized approach for an acoustic attack is presented. Thirdly, we discuss the details of the new attack, the acoustic triangulation attack. This includes the computation methodologies and experimental procedures. Fourthly, we evaluate the results and recognition rates from a set of experiments. Lastly, we suggest some ways to defend the new attack.

3.1 Progress Report

In the second semester, we carry on with our experiments on the keypad acoustic attack using the acoustic triangulation approach.

In January, we have reviewed some papers and books about acoustic characteristics of keys and acoustic classification methodologies. Moreover, we have developed a system using ***Matlab*** for feature extraction and recognition of keystrokes.

From February to April, experiments are carried out to prove the feasibility of finding a key uniquely using two microphones. At the same time, we analysis the results and try to improve the system.

In May, we gather all the materials together and write the thesis.

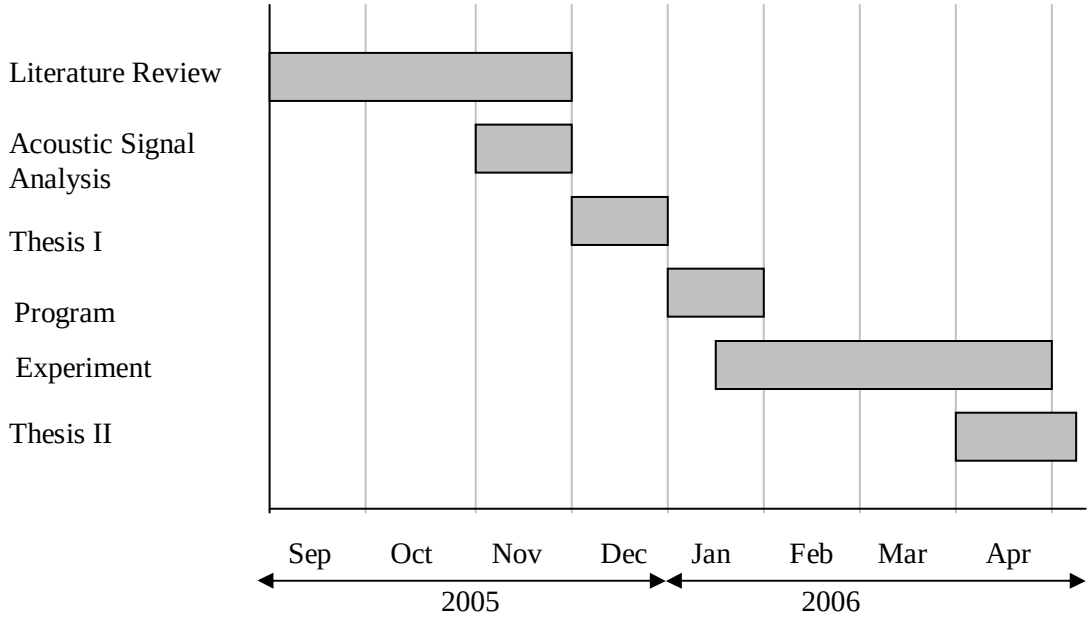

Figure 3.1 Project progress

Since this is a group project, we have divided the project workload evenly. (Figure 3.2) Most of the time, we work and discuss together. Before every meeting, we prepare the materials needed. This enables us to finish our work on time. And hence, our project proceeds in a very good way.

Items		Person in Charge
Literature Review		Both
Program	Feature Extraction	Fiona
	Signal Classification	Eric
Experiment		Both
Data Analysis		Both

Figure 3.2 Distribution of workload

3.2 Generalized Approach

Summarizing different acoustic approaches [4], an acoustic attack can be generalized in three main steps:

1. Sound Collection

The first step of an acoustic attack is to collect sound waves. When a sound is recorded into the computer, analogue sound waves are converted into digital signals. During this process, parameters should be set carefully to generate a digital signal whose waveform is much closed to the original acoustic one. Preprocessing is also required to make the signal more distinguishable. [3]

2. Signal Analysis

After the signal is recorded, a unique characteristic must be extracted to distinguish between different signals. In our proposed new attack, we check the time difference for a sound to reach two microphones.

Analyzing techniques are also important to get accurate and most of the information from a signal. [4] In our experiment, we propose two analyzing methodologies, the maximum peak position approach and the correlation approach.

3. Statistical Classification

Before classification, training should be done to teach the system about the features of a key. It is usually done by inputting a large set of sample data.

To decide which key the signal belongs to, an efficient algorithm is necessary for decision making. Neural networks with the theory of Hidden Markov models are usually used for this purpose. [7]

3.3 Acoustic Triangulation Attack

Since sound is a wave, its traveling time increases when distance increases. By measuring the differences in arrival times of the sound wave at two microphones, we are able to determine the distance between the source and the microphones.

In this experiment, we try to use two microphones to locate a key. The time difference is defined as the time when the first sound is received minus that the second one is received. By proving that the time delay is unique for every key, we can find the location for a particular key.

In this section, the computation methodologies and experimental procedures will be discussed in depth. We will also predict the results and evaluate the errors which may occur.

3.3.1 Hypothesis

The two microphones X and Y are placed according to Figure 3.3.

Figure 3.3 An experimental setup

Let us define the distance between key ***i*** and microphone X be D_{ix}, and the one between key ***i*** and microphone Y be D_{iy}. The keystroke emits a sound which has a velocity V.

When key ***i*** is pressed, the sound wave produced will be received by the two microphones. Let us define x(t) be the time required for the sound wave to reach microphone X and y(t) be the one for sound wave to reach microphone Y.

Therefore, $x(t) = D_{ix} / V$ and $y(t) = D_{iy} / V$

The time delay of the received signal between microphone X and Y is

$$t_i = x(t) - y(t)$$
$$= D_{ix} / V - D_{iy} / V$$
$$= (D_{ix} - D_{iy}) / V$$

For a key to be distinguishable, we assume that the delay t_i is unique for each key. Based on the assumption, we are able to know which key is pressed just by finding the delay difference t_i.

3.3.2 Keyboard Acoustic Characteristics

As mentioned in Section 1.4, mechanical keyboard consists of a number of keys and a circuit board. Each key corresponds to a button on the board. It generally consists of three parts, a head, a bottom rubber part and an intermediate plastic part in between the head and the rubber. [1]

When a key is pressed, the dome-shaped rubber is squeezed. It then pushes the electrical switch (Figure 3.4) and closes the circuit. Upon strike, the keyboard plate vibrates and produces a sound.

Although the sounds emanated by the numeric keys are similar to our human ears, in fact they are different and can differentiate by computational analysis.

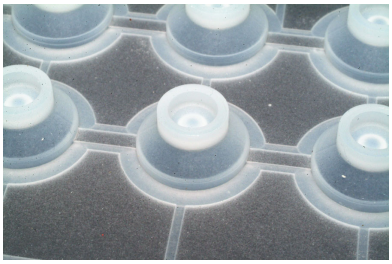

Figure 3.4 An electrical switch

The reason for different sounds is due to different positions of the keyboard plate stroked. Consider the acoustic mechanism of a drum. [1] When it is stroked at different positions on the plastic plate, different timbres can be produced. With similar principle, striking on the keyboard plate at different positions will causes the plate makes different sounds. Thus, the frequency components of different keys are different.

When a key is pressed, it actually produces two sounds. Figure 3.5 shows the acoustic signal of one click. The click lasts for approximately 100ms. We can see that the acoustic signal has two distinct peaks, a press peak and a release peak, corresponding to the pushing and releasing of a key. There is relative silence between the push and release peaks. Since the first peak is more significant than the second one, we will only consider the first one.

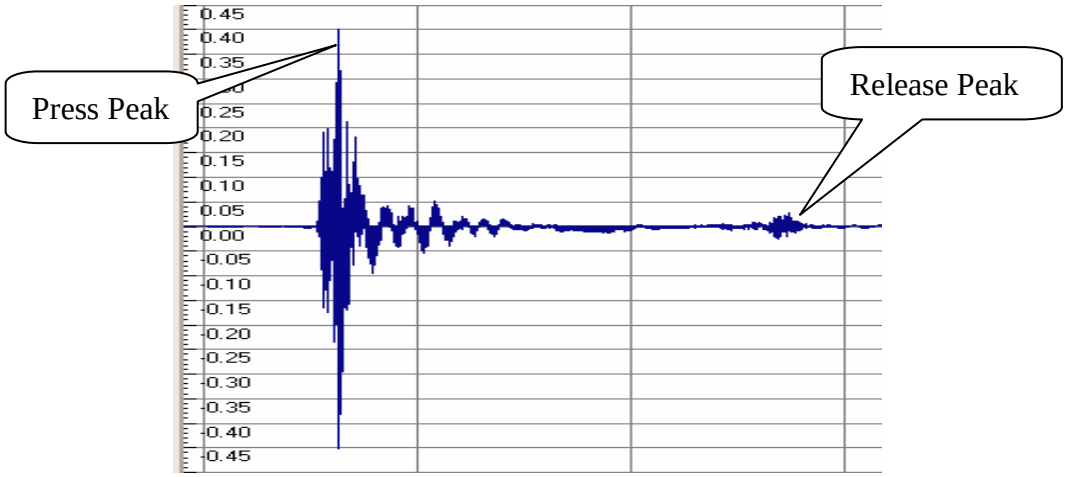

Figure 3.5 The acoustic signal of one click

3.3.3 Computation Methodologies

The goal of our experiment is to develop an easy way for finding the differences in arrival times of the sound wave at two microphones. Therefore, we would like to compute the input signals in simple methods.

The two methods used in our experiments are: the maximum peak position approach and the correlation approach.

3.3.3.1 The Maximum Peak Position Approach

From some simple experiments, we find out that there are some very sharp peaks in a keystroke. Assume that noise is not large enough to interfere the signal significantly. The positions of the sharp peaks are not affected in a great sense. Hence, we are able to find the time difference at receivers by comparing the positions of sharp peaks from the two waves.

A typical example is shown in Figure 3.6. We choose a few sharp peaks as reference points and compare the differences between them. In this case, the difference in received time is (T2-T1).

In the above approach, the comparison is conducted such that noise has a little influence on the sharp peaks. However, in real situation, noise do affects the position of maximum peaks at a random process. Therefore, it is expected that the received time differences vary within a range.

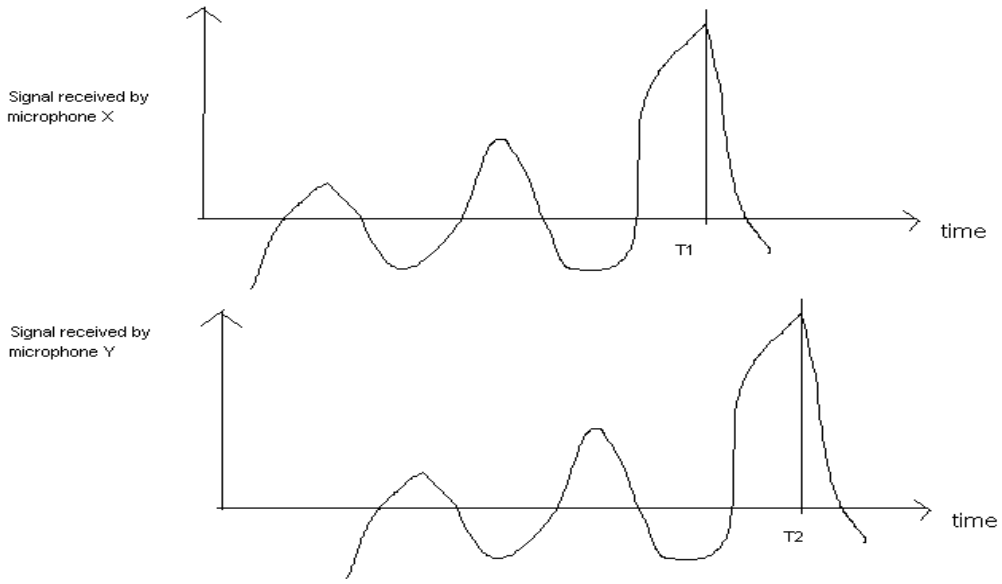

Figure 3.6 Explanation of using peak values as a reference point.

3.3.3.2 The Correlation Approach

The correlation between two signals, cross correlation, is a standard approach to feature detection and pattern recognition. A cross correlation function measures the similarity of two signals. It is commonly used to find features in an unknown signal by comparing it to a known one. [19]

Consider x(***i***) and y(***i***) represents the digital signals received by microphone X and microphone Y respectively, where ***i*** = 0, 1, 2 … N-1. The cross correlation ***r*** at delay ***d*** is defined as

$$r(d) \equiv \frac{\sum_i \left[(x(i) - mx) * (y(i-d) - my) \right]}{\sqrt{\sum_i (x(i) - mx)^2} \sqrt{\sum_i (y(i-d) - my)^2}}$$

r(d) will achieve the greatest value when x(***i***) overlaps with y(***i-d***) to show large amplitudes. (Figure 3.7) In this way, we can achieve the delay ***d***, which is equal to the time difference between sounds received by the microphones. We can also use ***Matlab*** to find d by plotting a graph of r(d) against d.

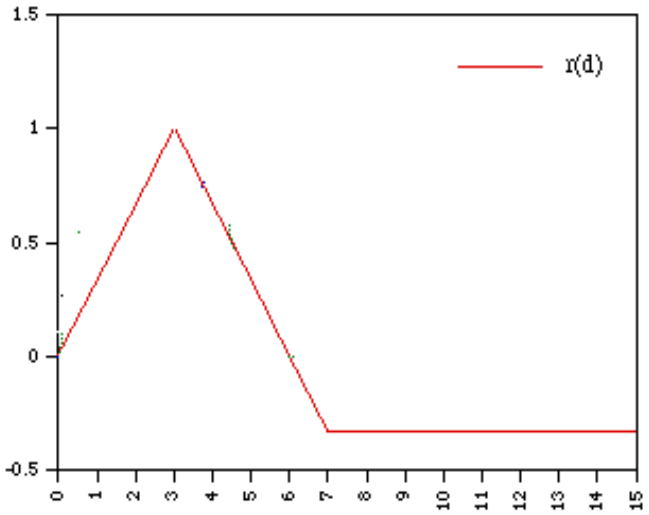

Figure 3.7 A graph of r(d) against d

3.3.4 Advantages of the New Approach

The previous approach relies on different sounds produced by keystrokes. However, such acoustic emanation attack can be prevented by making the keys to sound similarly or by applying an electronic sound of the same frequency. With these defenses, it is harder to analysis the keystrokes using the old approach.

Moreover, analyzing the frequency domain requires very deep knowledge in speech recognition and classification techniques. A lot of research must be done beforehand in order to perform this attack.

However, in the new approach, we do not need to equip with difficult acoustic theories and techniques. The only thing we need to do is to check the press peak and release peak of a keystroke. The time domain properties can be easily found out with many sound processing softwares, e.g. ***GoldWave.*** Then we can use a simple program to distinguish signals among keys by ***Matlab.*** These softwares are common and easy to use.

3.3.5 Experimental Setup

Keyboards. We used a Dell PS2 keyboard P/N 7N242. (Figure 3.8)

Figure 3.8 A Dell keyboard P/N 7N242

Microphones. Multimedia condenser type microphone. Sensitivity: -42dB, 0dB + 1Pa, 1kHz; Frequency response: 20Hz - 20kHz; Impedance: 2000 ohm; 3.5 mm stereojack. (HKD $20)

Computers. We used Dell Computers in IE Computing Lab ERB 1008. They are equipped with Pentium(R) 4 CPUs with 2.8GHz. The sound cards installed are SoundMAX Integrated Digital Audio.

Softwares. The input was digitized using a standard PC sound card. ***GoldWave*** (Free trial version) was used for recording the sound in mono with 96 kHz sampling rate. [23] ***Matlab version 7.0.1*** was used for comparing the waveforms and analyzing the recognition rate. [24]

3.3.6 Procedures

Firstly we placed the microphones on a horizontal line. The microphones were placed approximately 20 cm away from the keypad. (Figure 3.9) Then we recorded the sounds emanated using two computers.

Figure 3.9 The actual experimental setup

For each set of samples, we pressed the same key for a number of times with similar strength. Then we modified the signals using ***GoldWave*** before analysis. (Refer to Section 3.3.6.1)

After that, we fed the signals into a program written in ***Matlab***. (Refer to Section 3.3.6.2) The signals were analyzed in two approaches, the peak difference approach and the correlation approach. (Refer to Section 3.3.6.3)

Finally, we compared the recognition rates resulted from these two approaches.

3.3.6.1 Pre-processing the Recorded Sounds

When the keystrokes were recorded, the background noise was also recorded. Because keystroke sounds are comparatively weak, noise influences the signal in a greater sense.

Since the signals can be barely distinguished in random noise, it is better to reduce them before processing. (Figure 3.10) Due to time limitation, we have not developed our own noise reducing function. Therefore, we apply the "noise reduction" filter in ***GoldWave*** as its performance is quite good. [23] This filter filters away noise using frequency analysis. The result after processing is apparently showing a very clear signal. (Figure 3.10)

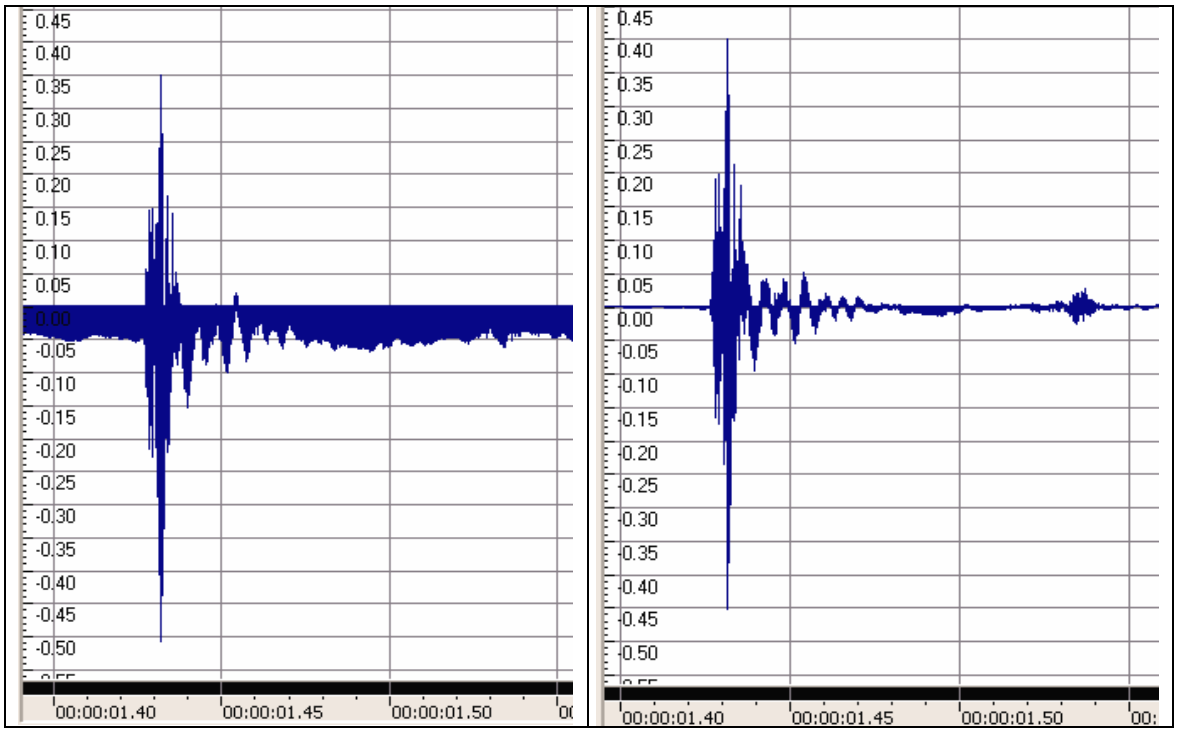

Figure 3.10 A keystroke signal before noise reduction (left) & after noise reduction (right)

3.3.6.2 Extracting the Keystrokes

A sound wave file contains many keystrokes. It is necessary for us to chop them out one by one for comparison. First of all, we read two processed sound waves into ***Matlab***. A sound wave is treated as a single array which stores the wave amplitudes as values. Then, two functions, "chopping" and "compare", are written for the purpose of extracting single keystrokes.

chopping.m

In the function “chopping”, its objective is to find two points representing the beginning and the end of a keystroke signal from a sound wave.

Firstly, this function imports a matrix of sound wave and its initial searching point.

Then, the function checks the values sequentially from the initial point. If the absolute amplitude of a sound wave is greater than a per-set threshold, then it is regarded as the beginning of a keystroke. According to our observation, a key pressed by normal strength can usually be distinguished with a threshold of 0.1. The first point which fulfils the above requirement is regarded as Point A. Then, the start point is set by Point A minus 2000. (Figure 3.11) By doing so, we can ensure that the complete beginning of a keystroke is included.

After the start point is found, the function checks for the end point. Another threshold is set to check the end of a keystroke. It is found that, after noise reduction, waves with absolute amplitudes less than 0.02 are generally not considered in a keystroke signal. Hence, the second threshold is set to be 0.02. When there are 2000 successive points with amplitude lower than 0.02, we set the last point as Point B. The end point is recorded as adding Point B by 2000 to obtain the complete keystroke. (Figure 3.11)

Finally, this function outputs the start and the end points to the function “compare”.

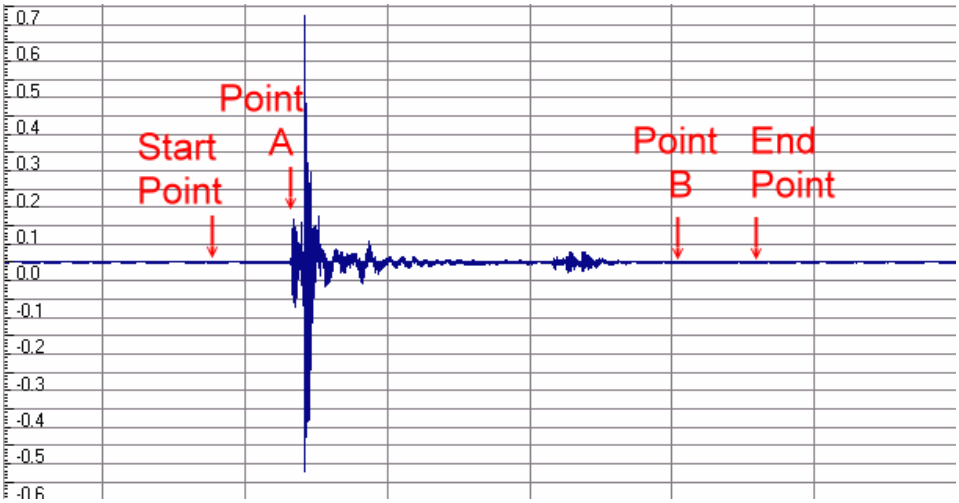

Figure 3.11 The start point and end point of a typical keystroke

compare.m

The objective of this function is to extract slices of chopped signals such that they are synchronized in time.

Firstly, this function reads in two wave matrices. The matrices are passed into the function “chopping” separately.

With the two pairs of start and end points, we compare them to get a common pair. To get a common start point, the start points are compared and the earlier one is chosen. Similarly, the end points are compared and the later one is chosen as the common end point. (Figure 3.12)

Secondly, the data within the selected range is copied from both sound matrices. They are then saved in two independently matrices for later comparison. The initial point for the next comparison is set to be the common end point.

The above process is repeated until all the keystrokes are extracted. Finally, the two processed signal matrices are output to the analyzing functions.

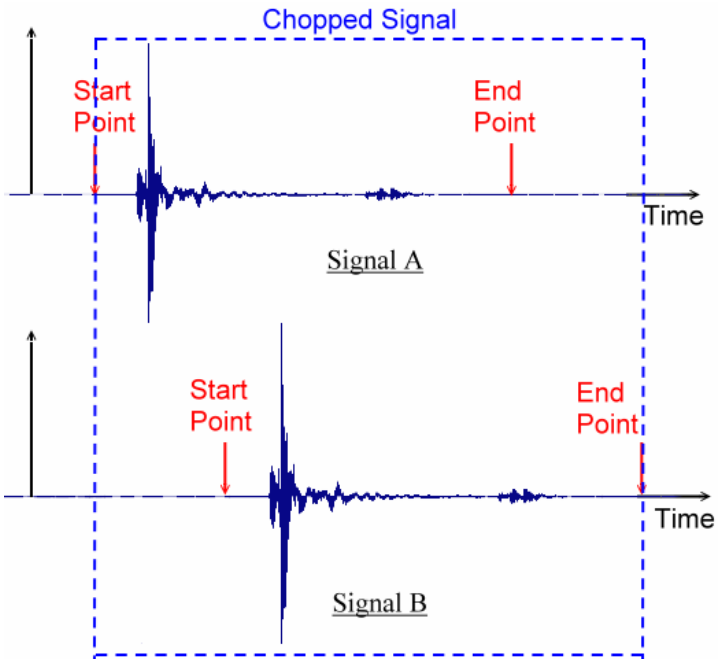

Figure 3.12 The selection of the common start and end point

3.3.6.3 Computing the Received Time Difference

As mentioned, there are two methods of computing the received time difference — the maximum peak position approach and the correlation approach. In the experiment, we process the signals using these two methods separately and evaluate their respective performances.

Two functions are written for each method. Both functions are divided into two parts. The first part is the main computation part. The second part is recognition calculation which is the same for both functions. This will be explained in detail in Section 3.3.6.4.

In this section, we will only discuss the first part of these two functions.

compute_receive_time_difference_by_peak_position.m

From the function "compare", we get two matrices of chopped signals.

For each set of data, we find the position of the maximum peak in each wavelet. Then the time difference is simply found by the difference between the positions of the two maximum peaks. The values are stored in an array for later recognition.

compute_receive_time_difference_by_correlation.m

To perform cross correlation between two signals, we apply the ***xcorr*** function in ***Matlab***. [24] This function returns the cross-correlation sequence in a length 2*N-1 vector, where x and y are length N vectors (N>1). If x and y are not the same length, the shorter vector is zero-padded to the length of the longer vector. By default, ***xcorr*** computes raw correlations with no normalization.

$$\hat{R}_{xy}(m) = \begin{cases} \sum_{n=0}^{N-m-1} x_{n+m} y_n^* & m \geq 0 \\ \hat{R}_{yx}^*(-m) & m < 0 \end{cases}$$

After ***xcorr*** is applied, we find the size and the maximum peak position of the output vector. The time difference is calculated by the following equation.

Time difference = Maximum peak position - (Size of output matrix / 2)

Then the values are stored in an array for later recognition.

3.3.6.4 Computing the Recognition Rate

This section describes the second part of the functions in Section 3.3.6.3.

For doing signal classification and recognition, there exist lots of advance algorithms and statistical models, e.g. the Hidden Markov models. Due to limited time, we only choose the simplest computation, the distance approach.

The recognition phrase can be further divided into two parts, the training and the classification. In the training phrase, the mean received time difference of each key is calculated from a set of training data. It is used as the reference for classification.

After that, we classify keystrokes in a recognition data set. The received time difference of a sample keystroke is compared with the means. The distance between the keystroke and a specific key is given by time difference minus the mean. By finding the minimum distance, we can classify a key for the sample keystroke it belongs to.

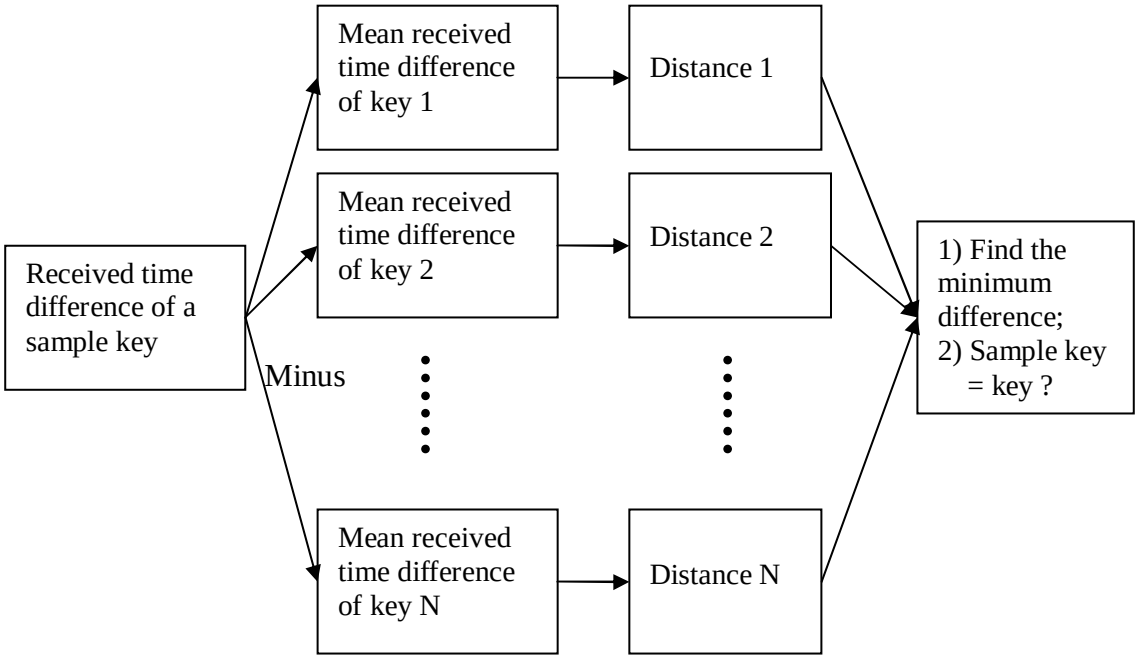

Figure 3.13 Classification of a sample key

3.3.7 Synchronization

Since we are measuring the time domain properties using two microphones, we need to synchronize the signal received in order to set the initial point. Synchronization can be done by using a mixer. The output of a mixer is a single sound wave. Since we need two wavelets to compute correlation, we choose the software approach for analysis.

Consider microphone X starts recording earlier than microphone Y. Let $\boldsymbol{n}$ be the time difference between the starting times of the two recordings.

For a particular key, the time $\boldsymbol{T_x}$ is required for the sound wave to be recorded by microphone X. While the time $\boldsymbol{T_y}$ is required for the sound wave to be recorded by microphone Y.

So the particular delay will be the sum of delay due to difference between $\boldsymbol{T_x}$ and $\boldsymbol{T_y}$. The actual delay $\boldsymbol{d}$ is

$$\boldsymbol{d} = (\boldsymbol{T_x} - \boldsymbol{T_y}) - \boldsymbol{n}$$

The time difference between the sounds received is unique. Since time difference ***n*** is a constant in the recording sample, the time difference between the sounds received under unsynchronized condition is still unique. So the synchronization issue is not a very big problem. However, we usually synchronize the microphones in real practice for easier comparison.

3.3.8 Expected Results

The speed of sound in air varies with the temperature. In IE Computing Lab ERB 1008, the room temperature is approximately 20°C (70°F). Hence the speed of sound is approximately 344 meters/second.

The least value of $(D_{ix} - D_{iy})$ = 2 * separation of two adjacent keys
= 2 * 1.5cm
= 0.03m

The minimum value of delay = The least value of $(D_{ix} - D_{iy})$ / speed of the sound in air
= 0.03 / 344
= 87.2 picoseconds

Since the computers are powerful in IE Lab, we believe that they are able to distinguish small delays. However, there are always many people walking around. This may interfere the speed of sound. It introduces error to the experimental results.

3.3.9 Expected Error

The experimental result is expected to be affected by the following three types of errors:

1. Noise

Noise is always the main source of error for acoustic experiments. Background noise exists no matter where you are. Therefore, it is impossible to eliminate noise perfectly.

In our experiment, the recording of clicks and clacks are conducted in IE lab Rm1008. Noise comes mainly from the air-conditioners, computers, and noise made by students.

Since clicks and clacks are relatively soft sounds, the result will be greatly improved when we can eliminate the background noise. Hence, in the pre-processing stage, we use the noise reduction function in ***GoldWave*** to improve the situation.

2. Hitting variance

In this experiment, we argue that the time difference in keystroke sound received by two microphones can tell the unique position of the key.

However, in reality, the contact point is always differed slightly on pressing. It is impossible for us to press the key at the same position every time. The slight difference in hitting point will lead to a slight variation in the received time difference.

3. Shift of Location of Microphones or Keyboard

In the experimental setup, the keyboard and the two microphones are put flat on the same table. When we hit a key, we are hitting the keyboard and the table simultaneously. Since we are not hitting the key vertically, forces may cause the keyboard to vibrate and move slightly.

Even this small shift in position may cause error. This lead to a slight difference in hitting point and the differences in arrival times of the sound wave at the two microphones.

3.4 Experimental Results

Four types of experiments are taken according to the procedures in Section 3.3.6. Both the maximum peak position approach and the correlation approach will be used in recognition phrase. The accuracy rates will then be evaluated and compared.

The purpose of these experiments is to prove that the clicks and clacks of a keystroke can be uniquely identified using two microphones.

3.4.1 Comparing Keys '1' and '9' in a Keypad

In this experiment, the source sound waves are recorded by pressing '1' and '9' in a keypad each for 25 times. It is then modified with Goldwave. (Refer to 3.3.6.1)

The modified waveforms are processed using the maximum peak position approach and the correlation approach. In both methods, we find out that the two keys are practically indistinguishable. The ranges of the differences in arrival times of the sound wave at two microphones are overlapped. (Figure 3.14, 3.15)

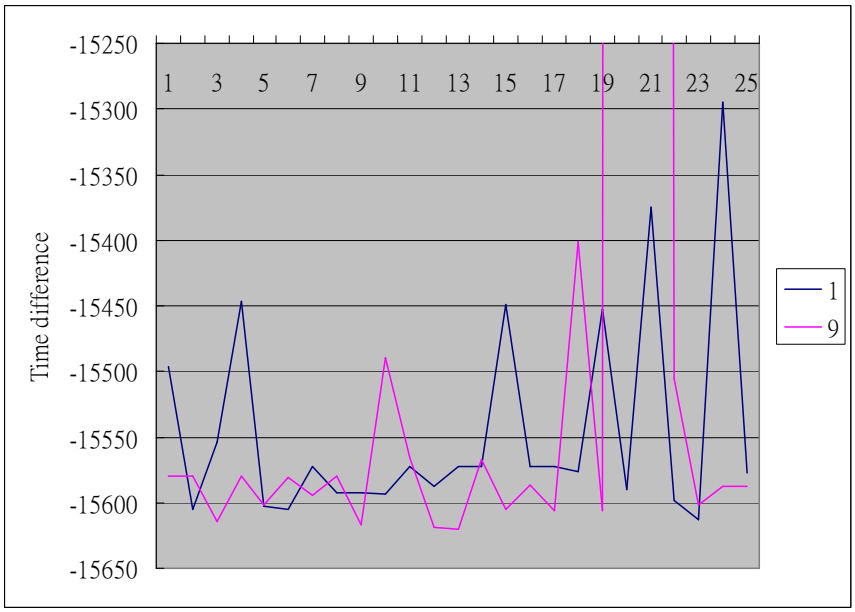

Figure 3.14 A graph showing the time differences of keys '1' and '9' using the maximum peak position approach

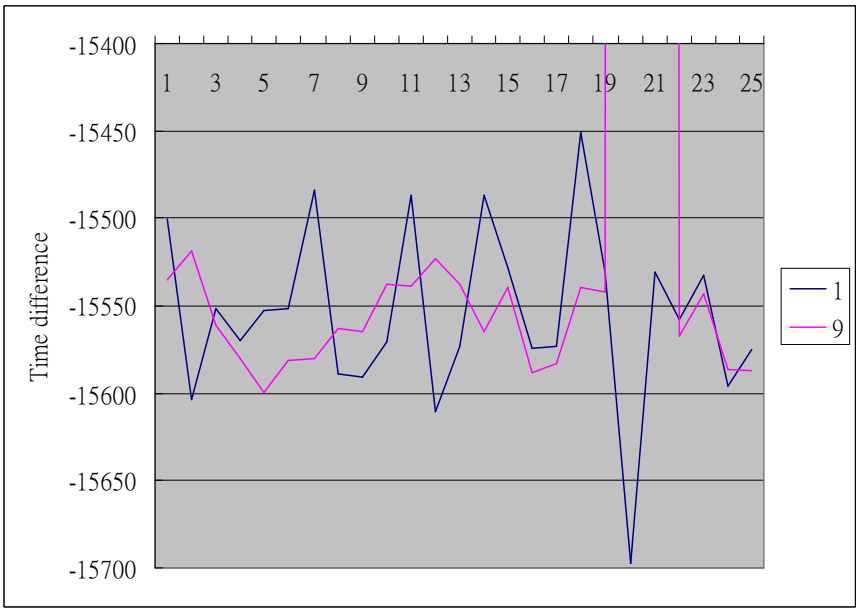

Figure 3.15 A graph showing the time differences of keys '1' and '9' using the correlation approach

There are mainly two reasons for this result. Firstly, as we use only simple tools to record the sound waves, the time differences between two close keys may be too small that it cannot be detected.

Secondly, random noise causes variation in the received time difference. When we measure two close keys, the variation caused by noise is more likely larger than the time difference. And hence, the difference is not very obvious.

Mathematical explanation:

$t_i = (D_{ix} - D_{iy}) / V,$

$t_i - t_j = (D_{ix} - D_{iy}) / V - (D_{jx} - D_{jy}) / V$

$= [(D_{ix} - D_{iy}) - (D_{jx} - D_{jy})] / V$

$= 1/2D_{ij} / V$, where $D_{ij} = 1/2[(D_{ix} - D_{iy}) - (D_{jx} - D_{jy})]$

D_{ij} is the separation of the key i and key j. If the separation is too small, it could not be detected and recognized due to the large noise interference.

Therefore, we try to find the uniqueness in two keys which are further apart in the next experiment.

3.4.2 Comparing Keys 'Z' and '3' in a Keyboard

In this experiment, we choose to check keys 'Z' and '3' in a keyboard. We assume that when the keys are further apart, the variation ranges of received time difference will not overlap. And hence, the two keys can be recognized.

So the experiment begins with pressing each of the two keys 30 times for training. Then each key is pressed for 10 extra times for recognition.

When we apply the maximum peak position approach on the sets of data, we can find out that there is no overlap between the two keystrokes. (Figure 3.16) The recognition rates of 'Z' and '3' are 95% and 100% respectively.

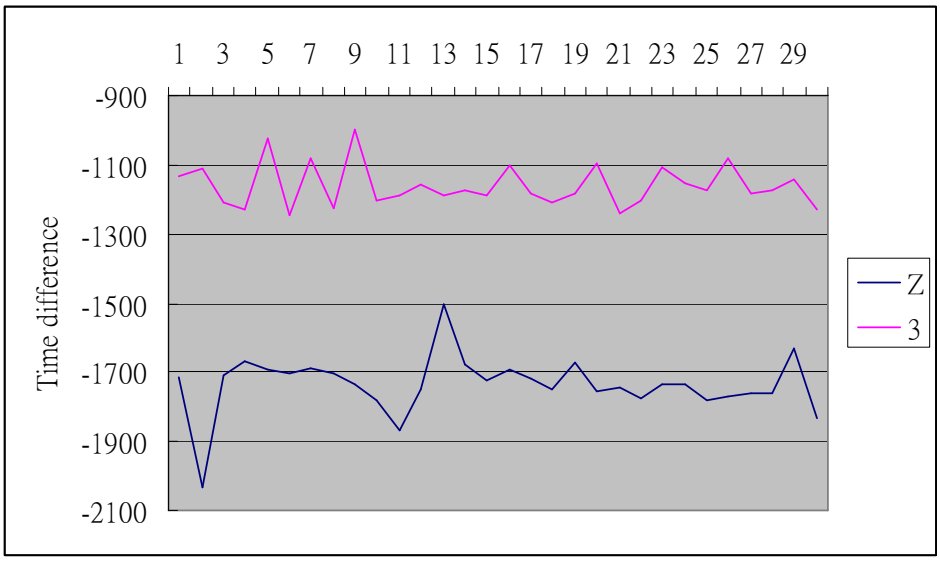

Figure 3.16 A graph showing the time difference of keys 'Z' and '3' using the maximum peak position approach

When we apply the correlation approach, there is still no overlap between the keystrokes. (Figure 3.17) The recognition rates of ‘Z’ and ‘3’ are 100% and 95% respectively.

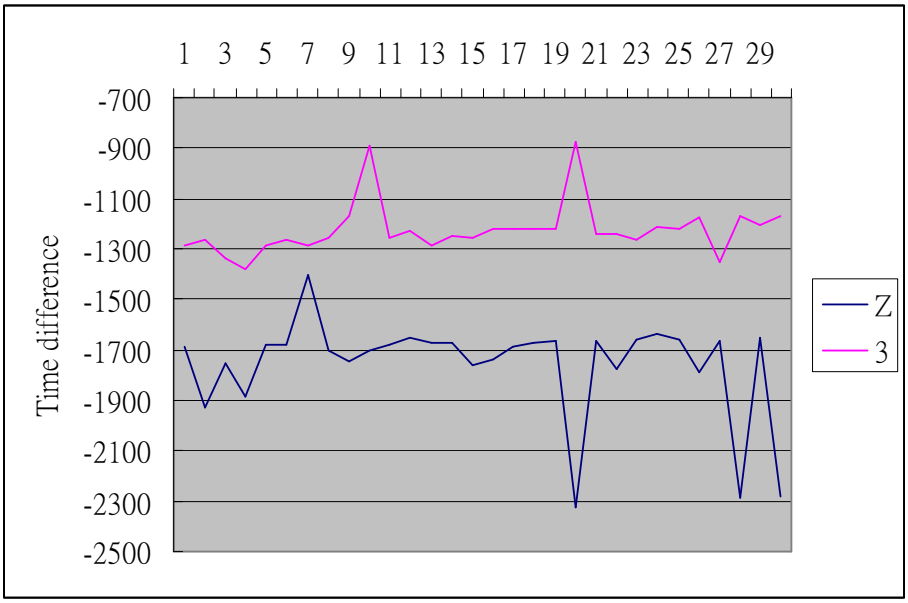

Figure 3.17 A graph showing the time difference of keys ‘Z’ and ‘3’ using the correlation approach

Therefore, we can conclude that we can distinguish two keys uniquely. Moreover, the recognition rate is very good when there are only two keys.

3.4.3 Comparing Keys ‘Z’, ‘B’, ‘/’, ‘↑’(up) and ‘3’ (keypad) in a Keyboard

Next, we try to prove the uniqueness between more keys. In this experiment, we compare five keys with similar distance in between: ‘Z’, ‘B’, ‘/’, ‘↑’(up) and ‘3’ (keypad) in a keyboard.

The experiment begins with pressing each key 30 times for training. Then each key is pressed for 10 extra times for recognition.

When the maximum peak position approach is applied, we find out that there are some overlapped regions between keys. (Figure 3.18) However, the mean time differences are quite distinguishable. (Figure 3.19) The reason for the variation is mainly due to random noise.

Using the mean time differences as reference, the recognition rates of keys ‘Z’, ‘B’, ‘/’, ‘ ↑ ’ (up) and ‘3’(keypad) are 80%, 55%, 50%, 65% and 100% respectively. In this case, the recognition rate is quite good.

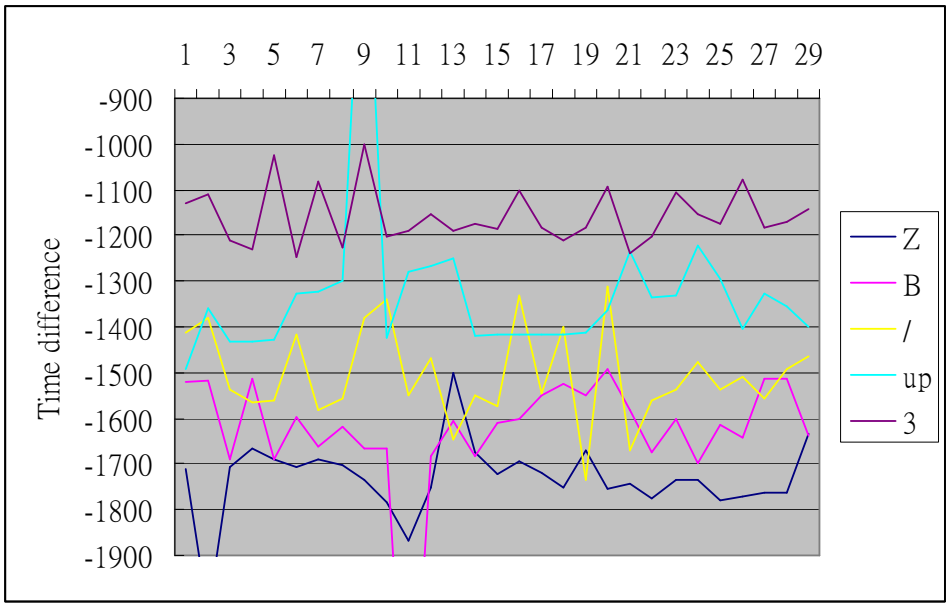

Figure 3.18 A graph showing the time difference of keys 'Z', 'B', '/', '↑' (up) and '3' using the maximum peak position approach

Time difference	Key				
	Z	B	/	Up	3
Average	-1734.97	-1629.13	-1504.97	-1323.17	-1160.40
Median	-1685.50	-1627.00	-1394.00	-1258.00	-1240.00
Maximum	-1400.00	-1574.00	-1282.00	-1135.00	-877.00
Minimum	-2328.00	-1703.00	-1569.00	-1388.00	-1377.00
Variance	41003.31	693.77	3264.62	5170.66	11074.44

Figure 3.19 The time difference of keys 'Z', 'B', '/', '↑' (up)and '3' using the maximum peak position approach

Next, we compute the set of data with the correlation approach. The result is not as good as the one using the maximum peak position approach. There are slightly more overlapped regions. (Figure 3.20) Luckily, the mean time differences of the keys are still distinguishable. (Figure 3.21) Using the mean time differences as reference, the recognition rates of keys 'Z', 'B', '/', ' ↑ ' (up) and '3'(keypad) are 32%, 100%, 80%, 45% and 75% respectively.

The recognition rate is satisfactory. This means that a key can be uniquely identified among a set of horizontal keys.

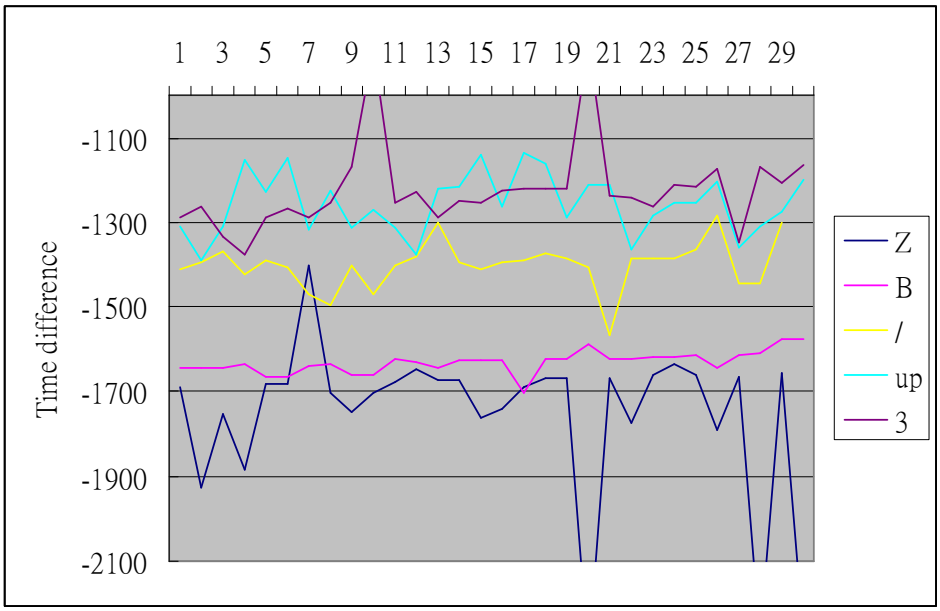

Figure 3.20 A graph showing the time difference of keys 'Z', 'B', '/', '↑' (up) and '3' using the correlation approach

Time difference	Key				
	Z	B	/	Up	3
Average	-1758.93	-1630.23	-1400.76	-1256.40	-1222.80
Median	-1733.00	-1607.00	-1537.00	-1357.50	-1178.50
Maximum	-1501.00	-1493.00	-1313.00	-395.00	-1000.00
Minimum	-2031.00	-2465.00	-1734.00	-1494.00	-1247.00
Variance	7253.21	29392.95	10402.46	36030.49	3874.73

Figure 3.21 The mean time difference of keys 'Z', 'B', '/', '↑' (up)and '3' using the correlation approach

3.4.4 Comparing Keys 'F5' and 'B' in a Keyboard

In previous experiments, we are only comparing keys in horizontal. Now, we want to prove that sound emitted by vertical keys can also be uniquely distinguished by two microphones. So keys 'F5' (function key) and 'B' are tested.

This time, we press each key 50 times. Then we plot the time difference of each key after processed by the two methods.

By using the maximum peak position approach, we can see that the ranges of time difference for each key are quite distinguishable. (Figure 3.22) There is only very little overlapped regions. Yet, when the correlation approach is applied, the result is not as good as the first one. (Figure 3.23) We cannot tell the range of time differences for a specific key.

But still, with the maximum peak position approach, we can conclude that two vertical keys can be uniquely classified using two microphones.

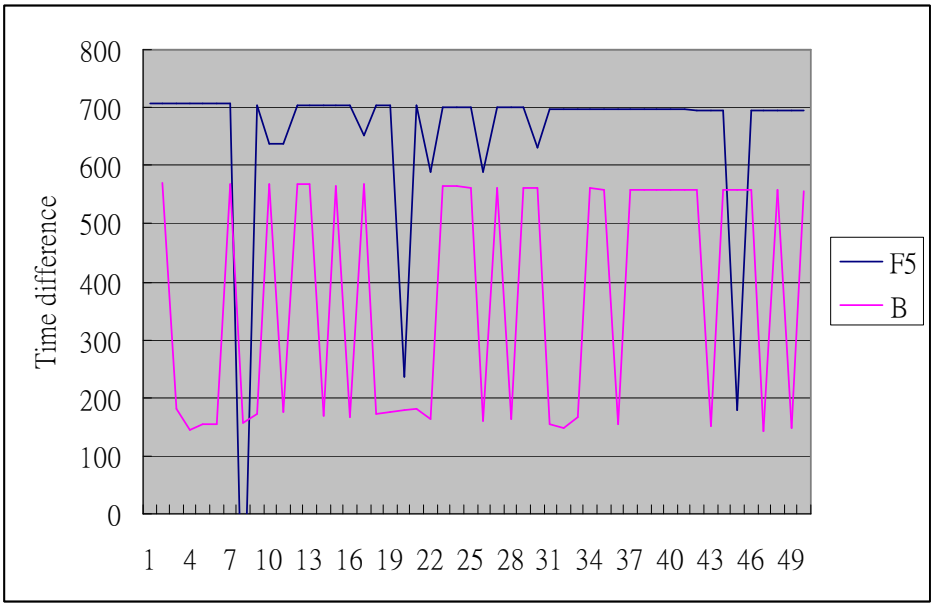

Figure 3.22 A graph showing the time difference of keys 'F5' and 'B' using the maximum peak position approach

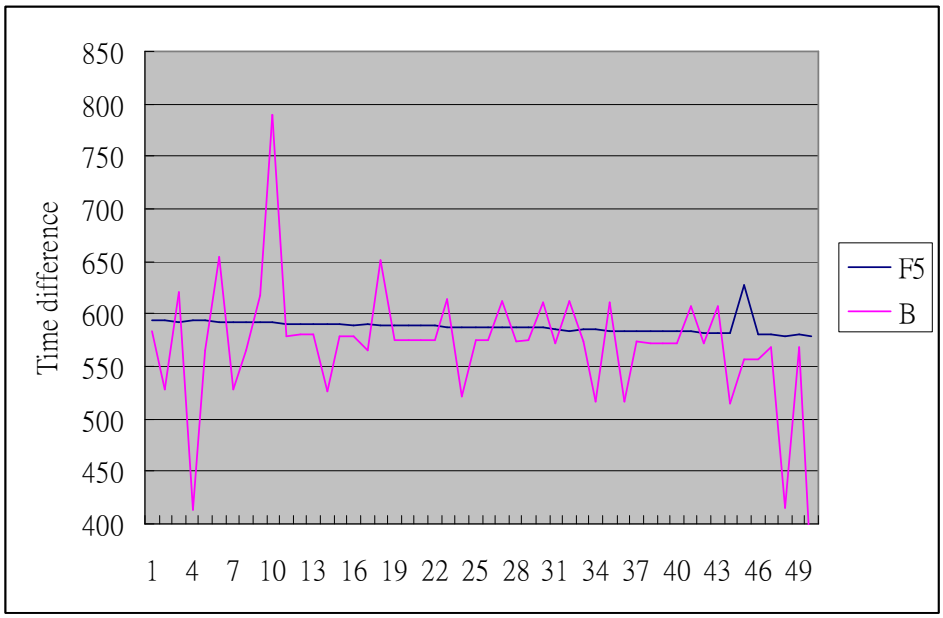

Figure 3.23 A graph showing the time difference of keys 'F5' and 'B' using the correlation approach

3.4.5 Comparison between the Two Methods

Based on the above experiments, we find out that the maximum peak position approach generally performs better than the correlation approach.

The correlation approach often leads to a larger range in time differences. It is mainly due to the more complex mathematical computation in cross correlation than in simple peak and peak difference. However, we would still keep the correlation approach in our recognition process because it still gives valuable information for classifying a key.

Consider the experiment with five keys. (Section 3.4.4.3) Some keys have a better recognition rate with the maximum peak position approach, while the others with the correlation approach. Therefore, to get a more accurate recognition rate, we could combine the results from both approaches. According to a simple experiment, the accuracy rate of the combined system can be up to 85%.

3.4.6 Limitation of the Experiment

The experimental setup is only desirable for checking keys with a quite large distance apart, e.g. the distance between keys ‘Z’ and ‘B’ is 7cm.

Since this is a preliminary research, we only use simple tools and algorithms to prove our hypothesis. Therefore, we cannot make precise judgment on the positions of close keys.

The accuracy rate also decreases if this experiment is performed under a very noisy environment. Because keystroke sounds are relatively soft.

3.4.7 Improvements

The following improvements can be made to increase the recognition rate:

1. Noise Reduction

Random noise leads to variation in time differences. It is not desirable since it will cause error in recognizing a key. To minimize the effect of random noise on the reference time difference, we can use a larger set of data as training to obtain a more accurate mean. But we can do nothing to improve the recognition of a single key.

2. Combining Two Approaches

From the experimental results, we can see that both the maximum peak position approach and the correlation approach give valuable information on the classification of a keystroke sound. Therefore, the recognition is improved when we combine the results of the two methods and make the final judgment. (Refer to Section 3.4.5)

3. Use More Microphones

The time difference of a key is based on the difference in arrival times of a keystroke sound wave at the microphones

In our experiment, we only use two microphones. This means there are only two factors per key for reference. However, if we add more microphones, we can get more factors, i.e. more information, about a keystroke. Thus, a key can be identified with a higher probability. And hence, the recognition rate can be improved.

4. Advanced Equipments and Algorithms

Simple equipments and algorithms do not allow us to classify keys with a small distance in between. If we use more delicate microphones and powerful sound card, we can extract more features from a keystroke. With an advanced algorithm in classification, we can combine more parameters to judge a key better. Thus, the recognition rate can be improved. A detail description of the advanced algorithm will be discussed in Chapter 4.

5. Dictionary

It is often found that, in a pool of input data, there exist a number of repeated sequences. This is true especially for some specialized keyboards, e.g. password for entering a security system.

In the computer world, a dictionary is a list of entries which are likely to appear in a pool of data. A dictionary can be built to store frequently inputted sequences. By comparing the input with sequences, the possibility of getting the next key correct increases. [2] Hence, the recognition rate can be increased.

3.4.8 Summary

According to our experiments, we can conclude that a key can be uniquely distinguished using two microphones. For each key, the difference in arrival times of a keystroke sound wave at two microphones is unique. It is unique no matter the two keys are horizontally or vertically lay on the keyboard.

Moreover, the recognition rate is quite good for both the maximum peak position approach and the correlation approach. By choosing 3 to 5 keys from a set of 104 keys, the recognition rate can be up to 80% with a 5-minute computation. The rate can be improved when the results from both methods are considered in classification.

The main source of error in this experiment relates to noise. It causes variation in the time differences, which then decreases the recognition rate. Delicate tools and advance algorithms can be used to solve the problem.

To conclude, it is a sparkling finding that a keyboard can be eavesdropped using cheap tools and simple algorithms. Only basic knowledge in sound mechanism is required. Therefore, this attack is more threatening than the previous attacks.

3.5 Defensive Methods

Previous keyboard acoustic attacks are performed by examining the frequency domain of a keystroke. It can defend by making keys sound similar.

However, in our new approach, keyboard acoustic triangulation attack, keys can be detected when they just make a sound! It is more difficult to defend than before.

From our experiments, we find out that the recognition rate reduces when the surrounding noise level is high. Therefore, the acoustic triangulation attack may be defended by typing in a very noisy environment.

Another suggested method is to use virtual keyboards. (Figure 3.24) A projector projects the image of a keyboard. An infrared sensor scans the plane of the images to detect the intrusion of a finder into the desired portion of those images. [25] Since no sound is produced during typing. The acoustic triangulation attack can be defended.

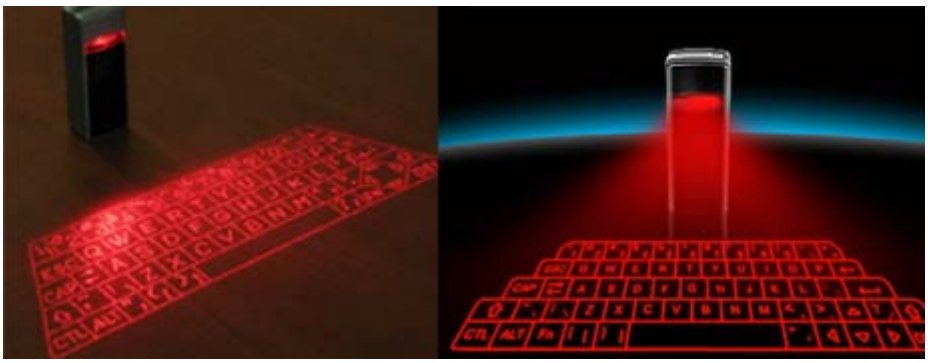

Figure 3.24 A virtual keyboard

Chapter 4 Further Development

We have successfully proven that a key can be identified by the difference in arrival times of a keystroke sound wave at two microphones. Even using simple tools and methods, the result recognition is quite high.

Since this is only a preliminary research, the whole system is not well developed. There are still rooms for improvements. Inspired by the research conducted by L. Zhuang, F. Zhou and J. D. Tygar, [2] we design an advanced algorithm, as follows, for better computation.

1. Data Collection and Preprocessing

Firstly, we take a recording of a user typing English text by three microphones. Parabolic microphones and a more powerful sound card should be used due to higher sensitivity.

Since noise is the main source of error, it is necessary for us to eliminate it before feeding into the system. Then a noise reducing filter should be customized. By applying advance noise reduction technique, most of the noise can be filtered away. Hence, the recognition rate is increased.

2. Feature extraction

Keystrokes are extracted in a method similar to the experiment. The two approaches, the maximum peak position approach and the correlation approach, are used to give a set of time differences. (Refer to Section 3.3.6.2 and 3.3.6.3)

3. Clustering

Before classification, a pool of data containing the time differences for keys is fed into the system for training. The computed time differences are divided into K classes, where K is slightly larger than the number of keys on the keyboard. Keystrokes of the same key are sometimes placed in different classes and conversely keystrokes of different keys could be classified in the same class. [2] During testing, the letter with the highest probability for each keystroke is picked to yield the best estimation.

The accuracy of clustering could be improved by matching alike English text sequences by means of Hidden Markov models. For example, consider the current key is probably either ‘g’ or ‘t’ and the previous key is known as ‘i’. Since it is more likely to have ‘it’, the system will choose ‘t’ instead of ‘g’.

4. Spelling and Grammar Checking [2]

After all the keys are identified, the system will perform a dictionary-based spelling check on the words. A simple statistical model of English grammar can be used to further correct the output text. Moreover, a feedback-based training can be applied over the intermediate result to improve the accuracy.

Since the experiment discussed is offline, it would be nice if we can put it online. That means a system can do all the stuff itself after the sound waves are recorded. A graphic user interface should also be developed to increase user-friendliness.

Chapter 5 Conclusion

Previous keyboard acoustic attacks require a deep knowledge in frequency domain characteristics of a keystroke sound. It also has a complex computation algorithm. In this project, we have proven that a key can be uniquely identified by applying ***Acoustic Triangulation Attack***. This attack computes the differences in arrival times of a sound wave at two microphones. The recognition rate of 3 to 5 keys from a set of 104 keys can be up to 80%.

This new finding is spectacular because ordinary people can also perform this attack! Experimental tools and softwares can be obtained easily from shops and the Internet. Hackers are able to know what you are typing just because your keyboard makes a sound!

We hope, through this project, to increase public's awareness in keyboard security. By knowing the attacking methods, people could protect themselves against hackers.

In the near future, hackers do not only thieve information from your computer or hijacking messages over the Internet. They do ***side channel attack***!

Reference

[1] D. Asonov, and R. Agrawal, “Keyboard Acoustic Emanations”, In ***Proceedings of the IEEE Symposium on Security and Privacy***, 2004.

[2] L. Zhuang, F. Zhou and J. D. Tygar, “Keyboard Acoustic Emanations Revisited”. In ***Proceedings of the 12th ACM Conference on Computer and Communications Security***, 2005.

[3] F. J. Owens, “Signal Processing of Speech”, New York: McGraw-Hill, 1993, pp.25.

[4] J. Harrington and S. Cassidy, “Techniques in Speech Acoustics”, The Netherlands: Kluwer Academic Publishers, 1999.

[5] H. Hermansky, “Analysis in Automatic Recognition of Speech”, in Speech Processing, Recognition and Artificial neural Networks, G. Chollet, M. G. Di Benedetto, A. Esposito and M. Marinaro Eds., London: Springer-Verlag, 1999, pp. 115-137.

[6] R. de Mori, “Statistical Methods For Automatic Speech Recognition” , in Speech Processing, Recognition and Artificial neural Networks, G. Chollet, M. G. Di Benedetto, A. Esposito and M. Marinaro Eds., London: Springer-Verlag, 1999, pp. 165-189.

[7] J. P. Haton, “Neural networks for automatic speech recognition: a review”, in Speech Processing, Recognition and Artificial neural Networks, G. Chollet, M. G. Di Benedetto, A. Esposito and M. Marinaro Eds., London: Springer-Verlag, 1999, pp. 259-280.

[8] S. Goronzy, “Robust Adaptation to Non-Native Accents in Automatic Speech Recognition”, Berlin Heidelberg: Springer-Verlag, 2002.

[9] Mathworld. http://mathworld.wolfram.com/.

[10] H. Bar-El, “Introduction to Side Channel Attacks”, Discretix Technology Ltd., http://www.discretix.com.

[11] D. Rocchesso, “Introduction to Sound Processing”, Universit`a di Verona, March 2003.

[12] R. McMillan, “Can Spies Decipher Keyboard Clicks?”, PC World, September 2005.

[13] B. Larkin, W7PUA, M. Reed, KD7TS, “On-Line User's Manual -- DSP10 2-Meter Transcevier”, http://members.ispwest.com/kd7ts/ver3/Man2.html, chapter 3 ***Windowing of Spectral Data***.

[14] A. Shamir, E. Tromer, “Acoustic cryptanalysis -- On nosy people and noisy machines”, http://www.wisdom.weizmann.ac.il/~tromer/acoustic/.

[15] G. White, “Cepstrum Analysis”, DLI Engineering, 1998.

[16] I. Potamitis, N. Fakotakis, G. Kokkinakis, “Speech Recognition Based on Feature Extraction with Variable Rate Frequency Sampling”, University of Patras, Greece.

[17] L. R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications on Speech Recognition”, in Proceedings of IEEE, Vol. 77, No.2, February 1989.

[18] C. Stergiou, D. Siganos, “Neural Networks”, Imperial College, London, http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html.

[19] P. Bourke, “Cross Correlation”, http://astronomy.swin.edu.au/~pbourke/other/correlate/, August 1996.

[20] Stuttgart Neural Network Simulator, developed at University of Stuttgart, http://www-ra.informatik.uni-tuebingen.de/SNNS/.

[21] P. Wright, “Sypcatcher: The Candid Autobiography of a Senior Intelligence Officer”, New York : Simon & Schuster Audioworks, 1987.

[22] H. Canistraroy, E. Jordan, “Projectile-impact-location determination: an acoustic triangulation method”, Meas. Sci. Technol. 7 , 1996, pp. 1755–1760.

[23] GoldWave, free trial version, http://www.goldwave.com/.

[24] Matlab 7.0.1, http://www.mathworks.com/products/matlab/.

[25] HoloTouch technology. http://www.holotouch.com.