

A Study on the Application of Digital Cameras to Derive Audio Information from the TVs on Trains

Yiju Wu, Shigeru SHIMAMOTO, Member, IEEE

Abstract — *Digital camera can help people to take video easier for anytime. For this reason, application of Digital video camera's technology is the hottest topic right now. Due to the fact that the TV on the train is in a public place it can't give Audio information. We were trying to find a way to get the Audio information by second time (ST) detection video only. (ST) "Second time" means the user uses their digital camera to take video from original video. We propose a new way to use Digital watermarking by embedding some pointer symbols into the source video, so passengers can use this kind of video to detect high quality Audio in ST situations. In order to show other possible application for this process in the future, in this study we also use VB programming to design simulation software "DDAS" (Directly detection Audio System). The software was designed to handle uncompressed AVI files. We tested this software using ST video. In the Experiments of this software successfully detected audio information and audio file types covering AVI, MPEG, MIDI and WAV file type. According to the questionnaires from the users, DDAS system's output Audio file was given a 3.8 MOS level which shows the system has enough audio embedded ability. From the questionnaires, we also found ST video's screen size will affect the detected audio's MOS quality. We found the best MOS quality was located in 1:1~1:4/3 screen size. We also use this kind of technology into students study and we found using multimedia will improve the student's grades and the grades about 6 average grades.*

Index Terms — DV, AVI, Digital-watermarking.

I. INTRODUCTION

Most people have had some experience watching the TVs in the trains when taking the JR lines. From these TVs, we can get all kinds of information in a short time. We can see the news and study English among many other things. TVs in trains are a way to give passengers all kinds of information. However, TVs in trains are located in public places, so the TV system can not give passengers any audio information. For example, when you use JR line's TV to study English, you might have a question. Unfortunately, we can not hear the advice or advertisements. This is a big problem for the TVs in trains now because the TVs in trains can neither give passengers enough nor ultimately correct information without audio. We have tried to find a way to solve this problem [7,8] of how to give passengers the audio information that they want to know.

With the rapid and extensive growth of the digital video camera, many communication companies like NTT, KDDI and AT&T have provided some application technology in digital cameras. Presently, NTT [1] has started using digital images to handle Digital-watermarking. NTT uses the static photograph to get some embedded files and give users some embedded information like ID addresses or some audio files. From this case, we can see how to apply digital cameras with Digital-watermarking technology will be the hottest topic in the future.

We can also use this kind of technology to solve the problems that the TVs in trains have. There are some different file types for static images and the video file types found in the TVs in trains though. For this reason, we need to design new software to handle it. We tried to embed some audio information into the original video, so users can just use their DV (digital video camera) to take video (ST video) from the screen, and directly detect audio from the ST video. We have successfully created DDAS (Direct Detection Audio Software) system, and we will introduce the details of the technology in the following chapters.

II. REVIEWS OF PREVIOUS WORKS

In the past, there were two main theories of digital watermarking, one based on frequency measurements and the other based on spatial measurements

Jordan [1] has proposed using a moving spatial theory to determine whether data has embedded information or not. He claimed that the frames can divide into two kinds of areas: 1 and 0. He proposed to use a statistic method in order to improve the success rate of the detection process. Linnartz [2] has proposed using a method called Group of Pictures (GOP) to be the basis of the image coding. Depending on the data's size, the symbol would also change. Although the digital watermarking would more easily be detected in this way, the pixel quality of the frames might decrease. Hartung [3] has proposed using MC+DCT to do pseudorandom numbers (PN) image arrangements. In this method, the image will divide into 4 bits per area, while the property of the area is either 1 or -1. If the property changes into -1, this proposed method would perform the image re-image coding. Langelaar [4] also used DCT to be the basis of his theory, but he proposed to divide the image by 8*8 per unit, and use a high frequency coefficient to be the basis of the determinant. Sakazawa [5] proposed dividing 16*16 per image and defining b_0 ~ b_4 to define the space of the image.

Our study was based on the Linnartz [1] GOP theory and the Hartung [2] Discrete Cosine Transform (DCT) PN method. We also used the pointer ability in the programming to remember the relative place of the embedded data.

III. PROPOSITION OF ST VIDEO PROCESS

Before explaining the process of this system, we need to define the system input file type. Although there are all kinds of video file types, we define our digital video camera out file as AVI file. In this system, we need to have at least one frame with a pointer symbol.

In the DDAS system, AVI file is our ST video input file type. The AVI file's name is audio-video interleaved. The structure of the GOP (Group of Pictures) was made from JPEG file. The general AVI file's compression rate was 5:1. There are two types of frame rates, one is 30 frames per second and the other is 15 frames per second. Our system is defined as using uncompressed AVI. The uncompressed AVI can also be called Full screen AVI file. This means decompressing of the video was at a rate of 5:1 and the frame's rate was 30 frames per second. DDAS I/O file type we defined as the Input file Uncompressed AVI, and the output file type can be MPEG, AVI or WAV file.

How to apply the digital camera's optical ability to detect the animation's audio to make communication possible is the main topic of this thesis. We use the digital camera's optical ability to recognize various data and propose a scheme to detect audio file from second time (ST) video. This ST refers to the video taken from the screen. To design software to handle the detection process of ST video, we used programming methods. Our software has overcome the barrier that traditional watermarking has had in the detection of ST video. In our study, we used pointer symbols to memorize the embedded places of the watermark when using optical frames. In this study, we were able to successfully detect audio, creating a greater communication media on train TVs. The scheme in our study is shown in figure 1.

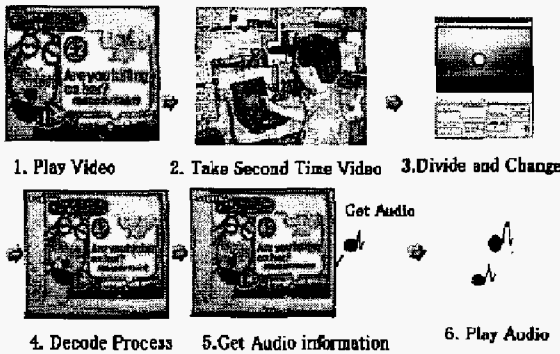


Figure 1, Schemes in Our Study

IV. WATERMARK EMBEDDING

In the embedding process, the traditional way of programming needs to use code to transform the design to the DCT (discrete cosines transform). This embedding process uses the following mathematical formula:

$$S_{uv} = \frac{1}{4} C_u C_v \sum_{x=0}^7 \sum_{y=0}^7 s_{xy} \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}$$

$$C_i = \begin{cases} \frac{1}{\sqrt{2}} & (i=0) \\ 1 & (i \neq 0) \end{cases}$$

$$S_{xy} = \frac{1}{4} \sum_{x=0}^7 \sum_{y=0}^7 C_u C_v S_{uv} \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}$$

00 means DC (Direct Current), the S_{xy} meanrepresents a value in the pel domain, and S_{uv} means represents a DCT coefficient.

In frequency domain watermarking techniques, the watermark is usually inserted into the mid-frequency sub-bands because they are relatively robust and have little impact on the image quality. We also choose to embed the digital watermarking in those sub-bands. Ching-Ting Chen [6] proposed a watermark embedding function by quantizing the coefficients. We define the quantization function Q as follows:

$$q(f, g) = \begin{cases} 1, & \text{if } : kq \leq f < (k+1)q, \text{ for } : k = \pm 0, \pm 2, \pm 4, \dots \\ 0, & \text{if } : kq \leq f < (k+1)q, \text{ for } : k = \pm 1, \pm 3, \pm 5, \dots \end{cases}$$

Where f is the wavelet coefficient and q denotes the size of the quantization interval. A different quantization interval should be chosen for different frequency bands. The quantity q is used as a parameter to control the image quality as well as the robustness of the embedded watermark. Letting

$$f = \left(\frac{f}{q}\right)^2 q, \quad r = f - f \text{ (quantization noise), and } b \text{ be the watermark bit, } f \text{ is updated by}$$

$$f = \Delta f + 0.5v, \text{ if } : Q(f, q) = b$$

$$\Delta f + 1.5v, \text{ if } : Q(f, q) \neq b, \text{ and } r = 0.5v$$

$$\Delta f - 0.5v, \text{ if } : Q(f, q) \neq b, \text{ and } r \leq 0.5v$$

One of the drawbacks of their method is that the result of the update is always located at exactly the middle of the quantization step. One can easily identify the updated coefficients just by examining their values. According to Ching-Ting Chen [6], we also modify their quantization method as follows:

$$Q(f, g) = \begin{cases} 1, & \text{if } : R+kV \leq d < R+(k+1)V, \text{ for } : k = \pm 0, \pm 2, \pm 4, \dots \\ 0, & \text{if } : R+kV \leq d < R+(K+1)V, \text{ for } : k = \pm 1, \pm 3, \pm 5, \dots \end{cases}$$

Where d is the distance between the two corresponding coefficients in sub-bands I_1^1 and I_3^1 , i.e., $d = w_1^1(p,q) - w_1^3(p,q)$, R is a random number (different for each watermark bit), and v is the quantization interval. R serves the purpose of determining the starting point of quantization; hence, with the variation of R , the result of the coefficient update will become random. Letting $d = [(d - R)/V] V + R$, $R = d - \lfloor d \rfloor$ (quantization noise), and w be the watermark bit, d is updated by

$$d' = \begin{cases} \Delta d + 0.5v, & \text{if } Q(d, v) = w \\ \Delta d + 1.5v, & \text{if } Q(d, v) \neq w, \text{ and } r = 0.5v \\ \Delta d - 0.5v, & \text{if } Q(d, v) \neq w, \text{ and } r \leq 0.5v \end{cases}$$

The amount of the distance update is equally distributed to the two coefficients to minimize the impact on the image quality. Letting $e = \lfloor d - d' \rfloor / 2$, the coefficients are updated by

$$w_1^1(p, q) = \begin{cases} w_1^1(p, q) - e, & \text{if } d > d' \\ w_1^1(p, q) + e, & \text{otherwise} \end{cases}$$

$$w_1^3(p, q) = \begin{cases} w_1^3(p, q) + e, & \text{if } d > d' \\ w_1^3(p, q) - e, & \text{otherwise} \end{cases}$$

The image of the proposed methods is shown in figure 2.

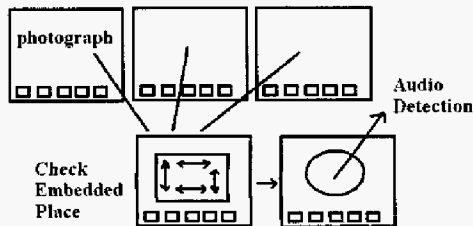


Figure 2, Embedded Information and Its Memory of Its Relative Positions

V. SYSTEM ORGANIZATION

There are many formats for organizing this system. For example, the user could click the link on the desktop, and the system will show the login format. When the user passes the login check, the system will show the main control form. From the main control, a user could edit the video and frames by using the "process" button.

As Figure 4.1 shows, there are four kinds of control tools: AVI video divide tool, decode tool, media control tool and control tool. By using these kinds of tools, the system will be able to handle the second uncompressed AVI detection process. The capabilities for every tool and format will be explained in the following.

1. AVI Divide Tool (Format):

AVI divide tool was designed on the left side of the main form. When the user clicks the button on this area, a user can edit the Uncompressed AVI file and divide it or convert the frames back to AVI video.

In this tool, because we only want to maintain the good quality of the divided frames, the system will change the RGB (Red, Blue Green) color. The mathematical formula for the RGB's change, as well as luminance component, is as follows. where, C_b & C_r mean the color difference components

$$Y = 0.2990R + 0.5820G + 0.1140B$$

$$C_b = -0.1687R - 0.3313G + 0.5000B$$

$$C_r = 0.5000R - 0.4187G - 0.0813B$$

When the divided frames return to the original format, the following mathematical formula will be used:

$$R = Y + 1.40200C_r$$

$$G = Y - 0.34414C_b - 0.71414C_r$$

$$B = Y + 1.77200C_b$$

In the dividing process, we define the AVI file based on the rate of 30 frames per second. When the user would like to return the frames back to the video though, he/she can define the frames by himself or herself.

The divided frame will automatically be saved in temp directory, and be given a sequence name by the system. Because the divided frame's file type was JPEG file, the user who checks the frames and selects the frames that have the pointer symbol embedded within need to change the frame into BMP file. Before to decode, system will check the file type as the following code:

```
Declare Function DecodePictureFile Lib "ddas" ( _
    ByVal strFilePath As String _
) As Long
```

2. The Tool to Change Frame File Type (Format):

After the video frames finish dividing, the user needs to use this tool to change the embedded frame. In this form, the system could handle JPEG or GIF file, change the file type into BMP file, and then return it to the original file type again. In the decoding process, the system was designed to handle the BMP file type.

The user needs to change the frame to BMP file so the decoding process will be able to function. In order to return the frames into original file type, the tools also have been designed with the ability to convert return file types. In this tool, when the user wants to change the BMP file back to the JPEG file, the system also allows the user to define the compression rate.

When the process will scan frame, we need to define the frame as the following:

```
Declare Function DecodePictureFileRect Lib "ddas" ( _
    ByVal strFilePath As String, _
    ByVal lngLeftX As Long, _
    ByVal lngTopY As Long, _
    ByVal lngRightX As Long, _
    ByVal lngBottomY As Long _
) As Long
```

After the frame scanned, system need to remember the

color of the frame also need to define and the define way as the following :

```
Declare Function DecodeGrayScaleArray Lib "ddas" ( _  
ByRef byData As Byte, ByVal lngWidth As Long, ByVal  
lngHeight As Long  
) As Long
```

The brighten single also control by the following code:

```
Declare Function DecodeBinaryArray Lib "Dcd" ( _  
ByRef byData As Byte, ByVal lngWidth As Long,  
ByVal lngHeight As Long _  
) As Long
```

3. Decode Tool (Format):

Decoding is the way for the system to find embedded Audio information. When users have finished changing the frame's file type into BMP file and opened the file in this DDAS, the system will start the decoding process. Decoding is the first step of the detection process. If the decoding process is successful, the audio information detection process will be able to start. In the decoding process, when the BMP file tries to find a pointer symbol in the frame, the system will scan the frame and check the resulting value and the value in the user's defined place. Depending on this information, the decoder will either find the pointer or print the error information for the user. The detail detection process is in 6 (detection process).

4. Media control Tool:

Because the embedded information is in audio file and we needed to run the music file in real time, the DDAS system was also designed to connect with Microsoft media player. This system's media control tool is depends on the Microsoft Media Player, so the input file type is the same as that of media player. The detection ability in this system can handle WAV, MPEG and three kinds of audio AVI files. Due to the audio file's quality being subject to change by different audio sound cards, we recommend users to use a speaker to keep the output file at high quality.

5. Software title format and Login format:

Because the software is designed for education's research and not intended for business, we wrote some copyright information in the title form and set a password on the login form.

6 Detection process

The detection process in DDAS system is different from traditional Digital-watermarking process (Figure 4.5). When this system reads an input file, the system will check uncompressed AVI file or compressed AVI file. If the file type is compressed AVI file, the system will show the error information for the user and the user will need to check their file type again. The system will automatically stop all of the process if the system can't find any uncompressed AVI file. If the input file is uncompressed AVI file, the system will start to divide the entire frame in this file.

Before dividing all of the frames from the ST uncompressed AVI file, the system will try to get the AVI

file's data information first. The System will find out the total Video time and data head of the video. Because the Video information's data type is non- integer file, we will need to exchange it into an integer one. After changing the time information in this video, the system will start the dividing process.

The frame's dividing process uses a rate of 30 frames per second. When the system finishes the dividing process, the it will automatically save all of the frames in temp directory. Every frame in the temp directory will be given a file name according to their dividing sequence, so the user can more easily find out the frames that they want by their file name. In our defined system, the frame will embed the audio information and the system will need to use the last frame to find out the audio. ST video's frame needs to include a pointer symbol. When the system detects the information form the frame, the frame's file type needs to be BMP file. In the traditional Digital- watermarking process, static frames need to change to BMP file because the BMP file can more easily be edited and has a smaller size than the JPEG file. For this reason, in our system we also need to top change the file type in the frame that we want to detect some information from.

After changing the file type of the embedded frame, the system will check if the frame includes a pointer symbol or not. If the frame doesn't include an embedded pointer symbol, the system will show the error information for the user and stop the process. How to find a pointer symbol in the frame is the main technology of the system. At first the system will scan the frame that was chosen by the user. In pointer symbol programming, if the pointer symbol is embedded in the frame, the defined integer "Recliner" will change their value and the value will be larger than 0. If the embedded place was defined by the user, the user's defined place's value will also be set larger than 1. By this theory, the system will check two of the numbers in this frame and decide if the frame has a pointer symbol or not.

If the embedded frame finds the pointer symbol, the system will call the module "ModDcd". This is the decoder module in our system. First it will find out and define the value "Position detection Pattern" to correct the place of the pointer symbol. After the area check has been finished by the system, "ModDcd" module will start to read the embedded data in the data area.

The embedded data is Audio information, and it is included in the audio file size, the place and the link. The data source could have two places where it is allowed in our system. These kinds of links could be in the system or Internet. When the DDAS system finds these kinds of information, the system will use the information to run the audio in Microsoft Media Player. The embedded link was control by the following code, and the system scheme as the figure 3.

```
Declare Function GetConcatenationInfo Lib "ddas" ( _  
ByVal lngIndex As Long, _
```

ByRef lngSeqNo As Long, _
 ByRef byCheckDigit As Byte _
) As Long

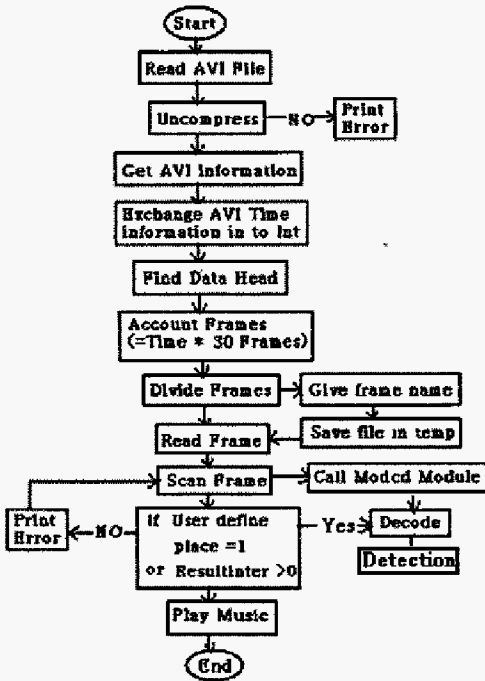


Figure 3, DDAS system programming process

VI. ANALYSES

Our proposed scheme has been designed into software, and we found that using pointer symbols can successfully solve the current bottleneck in ST video. Our software has been tested and questionnaires from users collected. According to the results of these questionnaires, our system's output audio file has been rated 3.8 MOS (MOS:ITUT.800). We also found that depending on the difference in screen size, optical detection ability will have a different MOS value. The recommended screen size is between original screen size and 4/3 screen size in order to obtain the best MOS value. (Figure 4)

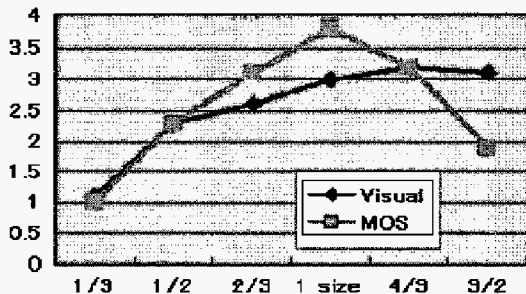


Figure 4, MOS and Different Screen Sizes

We using the software into Train's TV and we collected 100 people's examination papers. The 50 people were use traditional train's TV, and this kind of TV have no sound

and the another group were using our proposed (multimedia) way. From the figure 5 shows the sex of two groups are almost the same, and 60% of the user want to use our proposed scheme on TVs on train. Due to the software want to help the passenger to know the mean of the AD from TV's on train, so we try to use questionnaires again to be our analysis tool. We collect 100 questionnaires from the two group of the passenger (Multimedia group is the group that the passenger use our proposed idea.) We account the grades and the static data was show as the figure 6.

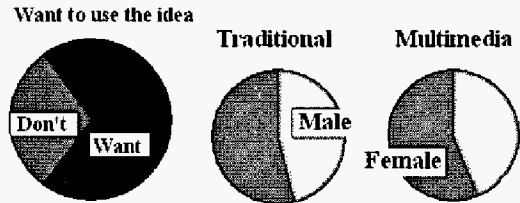


Figure5, Compare of difference ways in study

The red point on the figure 10 is the grades of the people who use traditional way to study and the blue point is the grades of the multimedia user. From the average of the two group, the average grade of traditional way's user have got 70.76 average grade but when we account the multimedia group's grade, we found the grades is 76.72. So from the average grades we could know using our proposed multimedia way on TVs on train will help the user to know the knowledge more.

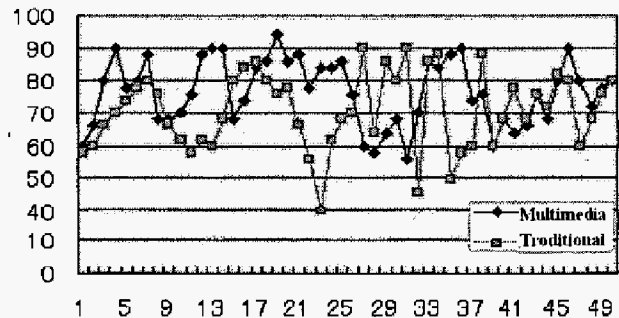


Figure6, Compare of difference ways in study

VII. EXPERIENCE AND RESULTS

When directly detecting audio information, the pointer symbol for embedded data has the same result as the traditional watermarking. Traditional video Digital-watermarking is not user-friendly and the software is quite expensive. Consequently, our aim is to find a cheaper and more user-friendly alternative. This will provide a solution to the problem associated with TVs in trains.

There are many ways to directly detect audio information, and in our thesis we propose that using a pointer symbol to remember the embedded data location will produce the same results as the traditional video watermarking. In order to show the viability of this idea, we created software DDAS to simulate the proposed process.

In the experiment, the Digital-watermarking embedded digital software could embed some audio in the source video, and add an arrow pointer symbol into the frame. It will help the video be able to detect the embedded file. The file was ST file. In our experiment, this process worked.

DDAS Software is a kind of system that can use detected music information and play it in real time. In this study, we can successfully read and decode the ST Digital-watermarking by the pointer symbol in video. We made our proposed idea come true and found a new application for Digital-watermarking. This study was successful in finding a way of detecting Digital watermarking from ST video. According to the graph of the effect factors between visual and Sound, we recommend keeping the ST video screen size between 1:1 to 1:4/3 screen sizes. By doing so, the output audio quality and visual quality will be good. In this way, we can get the balance of MOS level between the exchanges of the sound and visual. Depending on the different embedding and detection processes, the results might be different. Users need to find out the rules on ST uncompressed AVI files. It shows MOS levels in ST video also have been affected by different visual and sound factors. Application the software on TVs in Train will help the user to know the knowledge more and from the examination paper's average grades, we know the fact that using our proposed way will success to improve the user's grades about 6 average grades.

VIII. CONCLUSION

In this thesis, we have proposed a way to detect Audio information from the ST Video. In order to directly detect these files, we designed simulation software DDAS and it was based on the Digital-watermarking theory. This kind of way will also help the user easier to know the knowledge from TVs on trains.

In DDAS software, we could directly detect WAV, MPEG and AVI audio music information and play it in Microsoft Media Player. Using DDAS software, we could put the ideas into practice for the TVs on trains. This study's process is defined by using uncompressed AVI to help ST video detect some audio information. We finished this kind of simulation and created the system DDAS.

The ST video detection process that we are proposing is a little different from the Digital-watermarking system that NTT is using in mobile phones now. NTT's digital watermarking system can embed data links in the static image and it can make an application anywhere. In our system's process, we still can't use it in the mobile phone, and the detection ability will also be changed by the distance that digital camera took the ST AVI video. DDAS is just used in uncompressed AVI file, and for this reason, the system will not so easily handle other types of data.

In the near future, we think similar software will be created for other kinds of the tools, and using all kinds of new processes to handle different kinds of Video file types.

How to make detection processes faster and how to improve the detected Audio's quality will be the main topic in this kind of software's study in the future.

ACKNOWLEDGMENT

I would like to express my thanks to Prof. Shigeru Shimamoto for giving me permission to commence this research in the first place, to do the necessary research work and to use the laboratory resources. His kind ness and profound knowledge has given me a great deal of help at every step of my study. I need to give my special thanks to "Randy Birch"&" Ray Mercer" for sharing me a part of their source code. They were so kind to teach me how to design the detection process and gave me all kinds of new ideas, without their help maybe I wouldn't have been able to finish the software so quickly.

REFERENCES

- [1]F.Jordan,M.Kutter,T.Ebrahimi,"Proposal of a watermarking technique for hiding/retrieving data in compressed
- [2]J.Talstra,J.P.M.G Linnartz,"MPEG PTY-Marks' cheap detection of embedded copyright data in DVD video',Proceedings of computer Security-ESORICE 98,5TH European Symposium on Research in computer Security,pp221-240,1998
- [3]F.Hartung,B.Girod,"Digital Watermarking of Uncompressed and Compressed Video ",Signal Processing,Vol.66,No3,pp.283-301,1998
- [4]G.Langelaar,R.Lagendijk,J.Biemoed,"Real time Labeling Methods for Compressed Video "Proceedings on 18th Symposium on Information Theory in the Benelux,pp.25-32 ,1997
- [5]S.Sakazawa,Y.Takishima,M.Wada,"A watermarking Method Retrievable from MPEG Compressed Stream " IEICE Trans.Vol.E85-A NO.11,pp.2538-2543,2002.
- [6] Ching-Ting Chen," Wavelet-Based Image Authentication Techniques", Chaoyang University of Technology, 2004
- [7]Yiju Wu, Shigeru Shimamot," A study on direct audio detection from second take Video", IEICE, HCS, 2004
- [8]Yiju Wu, Shigeru Shimamot," A study on audio information detection scheme from second take video", Waseda University, 2005
- [9] Yiju Wu," Using multimedia technology in museum", Hufan University, IEB, 2005



Yiju,Wu

He graduated from Graduate School of Global Information and Telecommunication Studies and got master degree from Waseda University. Current, he work in CPT(CHUNGHWA PICTURE TUBES,LTD.) and research in multimedia communication and human

factors. He is a member of IEIEC, IEEE, ITE and ETC



Shigeru SHIMAMOTO

He graduated and got the degree of PH.D degree from Tohoku University. He joined NEC in 1987.His research filed is wireless communication and Optical communications. He is currently, a professor at Waseda University. He is also a member of IEEE and

IEICE.