Northeastern University

Computer Science and Engineering School

Telecommunication Engineering

Undergraduate Thesis Project

Real-Time Multi-channel TV Processing with Embedded TNT (TV Tuner) and FFmpeg Library

Author:
Yongda LIN

Student Number:
20144458

June 4, 2018

Abstract

Most of TV channels nowadays is diffused by internet network but majorally via the TNT (Turner Network Television). In this work, we firstly study in how a TNT signal generates and broadcasts and compare different TV tuner cards' performances. Then, we move on its related multi-channel processing that consists of streams reading, postprocessing as well as duplicate detection and prospect its future functionalities - synchronization of advertisements, demonstration of film detail. In this part, we use multi-thread to improve the quality of channels reading and provide the ability of multiple processing. As for our recognition algorithm, we search in the part of postprocessing and detection, we focus on its real time processing ability which means each frame can't be processed over the limit time. We take thousands of tests to get the response time table and make a comparison for our two algorithms.

Index terms : TV Tuner; FFmpeg; Multi-channel; Real Time; Video Processing; Detection

Contents

1 Introduction

Nowadays most of countries in the world is in the process of digital TV at different stages. The familial way to watch TV is changing. TV card is an external device that allows clients to watch programs on a computer or other display terminal. Mostly, a proprietary software is used to operate the audio/video/data content, which makes it difficult for users to add extra features.

This thesis presents our work for tv multi-channel processing where our video streams are captured by TV cards. There are 2 objectives in this work.

(i) Comparison the performance (resolution, price, channels number, etc) of different TV tuners cards (multiple channels PCI in particular for the fixed machine)

(ii) Multi-channel reading and processing (recording, scaling, converting, etc.) in real time by FFmpeg libraries and TV tuners

1.1 Background

1.1.1 Digitization of TV services

As we know that some TV signal capture devices are analog signals, or both analog and digital signals. In order to determine which device we should focus on, we investigated the digitalization of television signals at first.

According to our investigation, digitalization is the trend for all communications and information systems. It can use advanced source compression coding techniques to reduce the amount of source data while preserving the original information. The channel occupied resources during transmission are greatly reduced, and it is also convenient for the storage. Besides the digitalization enables advanced digital transmission technology, error correction coding technology, encryption technology, which make the transmission efficient and reliable, it can also make information processing easier, more flexible and inexpensive.

The switch-over process of digitization is being accomplished on different schedules in different countries. China has ended analog broadcasts on 14 May 2016 [1].

As for the transmission methods, digital television can be divided into three types: digital cable, satellite television and digital terrestrial television.

1.1.2 Digital terrestrial television

We investigated the television broadcasting technology and protocols used in France and Europe. Currently they are TNT and DVB-T respectively.

Digital terrestrial television (DTTV or DTT, in French "TNT-La television numerique terrestre") is a broadcast television technology in which terrestrial television broadcasts broadcast television content in digital format to televisions in consumer homes via radio waves. DTTV is a major technological advancement in the field of analog television and is replacing analog television as the new standard television broadcast [2].

Compared to digital cable, it has ability to cover not only cities, but also the remote areas, which has huge potential market. Comparing to satellite television, it has less noise and transmission delay.

Different countries have adopted different digital broadcasting standards. DVB-T (Digital Video Broadcasting — Terrestrial) is one of the most widely used formats worldwide.

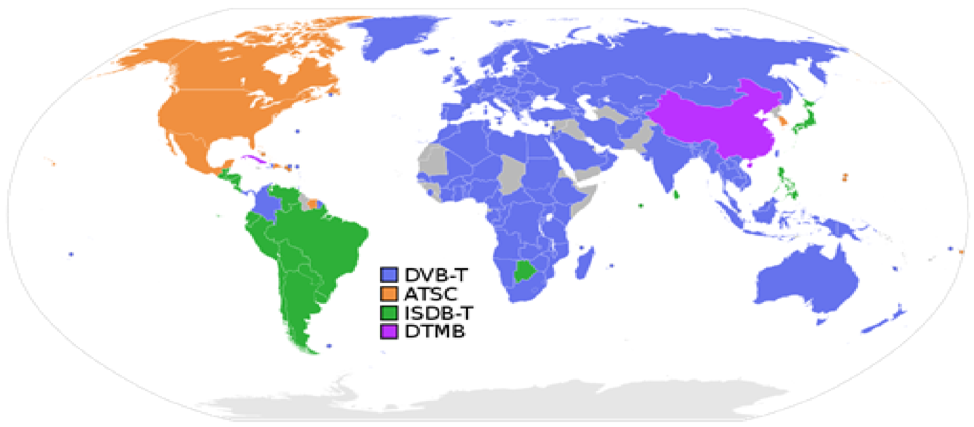

Figure 1: Worldwide Digital Terrestrial Television Systems

The core of DVB-T standard is MPEG-2 digital video and audio compression coding. It provides a payload rate of more than 5Mbps for standard-definition television (SDTV) signal transmission and supports mobile reception under conditions of vehicle speed movement; Single-frequency networking capability.

1.1.3 Mobile digital television

In addition to using TV cards to watch TV programs on PC, there are other similar methods often used. Our work can also be applied to these terminals with some modifications.

Mobile digital television is based on DTTV. It enables people to use different terminals to watch TV programs anytime anywhere. It is considered a revolution in digital multimedia technology and refresh the concept of television media.

Digital TV terminals can be car displays, computers, ordinary TV, pads or mobile phones.

There are some common terrestrial digital television terminals :

• Set-top box-For terrestrial digital television fixed reception. The set-top box receives terrestrial digital television signals, and its audio and video output is connected with a traditional analog television.

• Digital TV integrated - For terrestrial digital television fixed reception. The TV itself has the ability to receive digital terrestrial television.

• Car TV - Car TV terminal mobile receiving terrestrial digital TV. Used for bus, subway and private cars.

• Computer TV - A laptop or desktop computer receives digital terrestrial television through a small USB digital TV receiver.

• Portable/Handheld TV - Use small personal multimedia terminals (such as PMPs, PDAs, etc.) to watch digital terrestrial television.

• Mobile TV - Mobile phone with terrestrial digital TV function.

• Stick PC - a single-board computer. Barely two times larger than a USB key, these tiny ”Stick” format computers connect directly to the HDMI interface and turn any TV or monitor into a real PC [5].

1.2 Application

There are so much rebroadcast such as movies and advertisements in our worldwide programs. By the duplicate detection, comparing the contents of the current channel and the template which has been defined, we have the ability to recognize what the current content is. By dong so, there will be many interesting applications we could develop.

(a) Demonstration of Film Detail (b) Synchronization of Ads

Figure 2: Application of Duplicate Detection

Here we demonstrate two examples as our future application direction : demonstration of film detail and synchronization of advertisement. For the first one, we can obtain automatically the related film information like cast, outline and release date while we are watching the movie in the television. Also, for the ads companies, this also provides a function to help clients with their commands. Without doubt, the terminal of this application we are talking about can be your phones, Pad and so on.

According our research [6], the top five activities conducted on a PC while watching TV are: browsing the web in general, reading personal email, online shopping, browsing the web for content tied to what they're watching (over 39% said they did this), and reading the news.

Nearly 40% of respondents have already tried to manually bind the content they watch on the big screen with the small mobile screen. Imagining if there were some mechanisms which can automatically connect their devices, the percentage and the benefits it would bring would be higher. Also it can increase greatly the experience of the TV viewer.

1.3 Objectives

Before our development, we firstly investigate the current status of French television channels and related background knowledge. Including protocols, compressed formats, means for receiving television signals, etc.

In order to capture the TNT stream, we need to choose a TV card. So, we looked for the working principle of the TV card and its main performances. We have compared most of the details of the TV cards that have different number of input streams. After understanding the TV card, we chose a suitable one and captured the stream using it for our own processing.

After we've got frame-by-frame TNT streams in our computer. We use the FFmpeg library to perform real-time analysis on it. In the end, to display better the multi-channels processing, we choose the detection of duplicate video streams as our experiment example.

Additionally, we use multi-thread to handle multiple TNT channels in real time and process them in parallel.

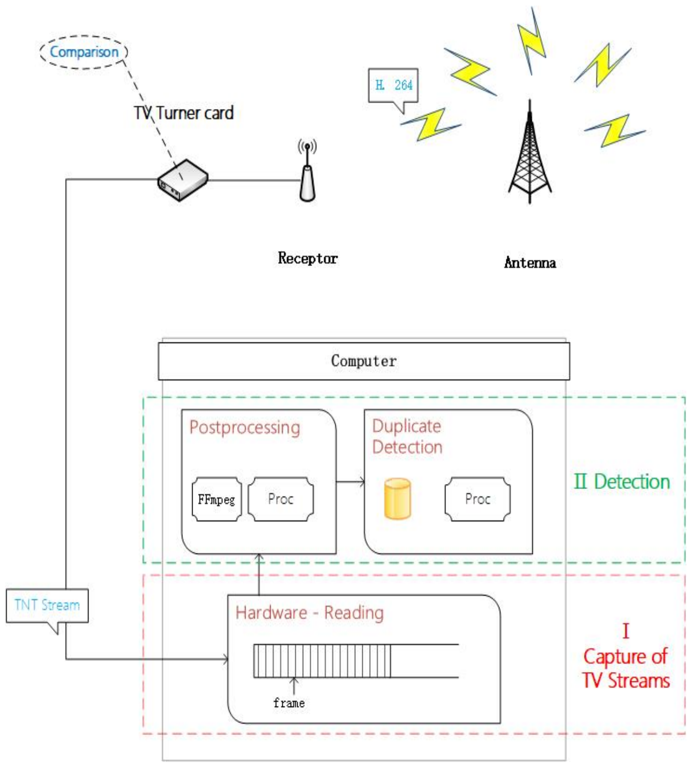

Figure 3: Objectives

2 TV Turner

This part is one of our work focuses. We found most of key preferences for TV tuner card and compared its details, in particular for those who have multiple input streams.

A TV tuner card [7] is a kind of television tuner that allows television signals to be received by a computer. Most TV tuners also function as video capture cards, allowing them to record television programs onto a hard disk. It uses an antenna to capture the TV streams which it supports and on the other hand it forwards them through different types of interfaces.

Usually it is equipped with a remote control, allowing it to be manipulated like a normal TV. Also, its supply company provides an app that is packaged on the PC and sometimes provides other useful features or SDKs for the developers.

2.1 Classification

• Analog television cards: It outputs a raw video stream, suitable for real-time viewing but ideally requiring some sort of video compression if it is to be recorded.

• Hybrid tuners: A hybrid tuner has one tuner that can be configured to act as an analog tuner or a digital tuner. Switching between the systems is fairly easy but cannot be done immediately.

• Combo tuners: This is similar to a hybrid tuner, except there are two separate tuners on the card. One can watch analog while recording digital, or vice versa. The card operates as an analog tuner and a digital tuner simultaneously.

• Mobile TV Adapter: External TV tuner card attachments are available for mobile phone handsets like the iPhone, for watching mobile TV.

2.2 Interfaces Types

The interfaces are most commonly either PCI bus expansion card or the newer PCI Express (PCIe) bus for many modern cards, but PCMCIA, ExpressCard, or USB devices also exist.

PCIe (Peripheral Component Interconnect Express) is a high-speed serial computer expansion bus standard. PCI Express 4.0 [4] provides a 16 GT/s bit rate. In the same time USB 3.0 [8] in the mode of SuperSpeed is reasonable to achieve only around 3.2 Gbit/s. There are almost 5 times from the USB one. In order to meet the needs of high-definition, PCIe is obviously more suitable.

2.3 Comparison of TV Tuners

The cards below typically include one or more software drivers to expose the cards' features, via various operating systems or let software applications that can further process the TV streams.

There are many applications for video capture cards like EasyCap which can convert a live analog resolution into some type of analog or digital media, (such as a VHS tape to a DVD), archiving, video editing, scheduled recording (such as a DVR), television tuning and video surveillance.

For better comprehension of our comparison (Table 1~3), we firstly explain the performances we're going to compare.

• Brand - Product brand, common are AVerMedia, Hauppauge, etc

• Model – Model of each cards

• Input - Number of possible input channels which is related to the parallel processing of multiple TNT streams

• P/I - The average price of one input - cost performance

• Dtt – Whether it can receive Dtt or not

• Sat Tv – Whether it can receive satellite TV or not

• Resolution - TV channel highest quality

• Store - Differ from the diffusing format, it's the compression format used to store data

• Preprocessing - Additional processing provided by the server

• Time-shifting - Recording a program to a storage medium to watch or listen to after the live broadcast.

• Development Ability - Secondary development capabilities : whether it is possible to export the stream in real time to do the other processing

In these tables, we pay more attention to its number of input streams, cost performance, resolution capacity, preprocessing and whether it can be developed. After our comparison, we give two recommend products as our result - TerrTac CINERGY DUAL and AVerMedia VE511-MN. They all support for multi-channel processing, high resolution and cost-effective.

Table 1: My caption

Brand	**Model**	**Operating System**	**Interface**	**Input**
AVerMedia	H727	WIN	PCI Express1x	1
AVerMedia	Nova T2	WIN, iOS, Android	PCI Express1x	1
Hauppauge	1213	WIN	PCI Express1x	2
TerraTec	CINERGY DUAL	WIN	PCI Express1x	2
DVBSky	T9580 V2	WIN, Linux	PCI Express1x	2
AVerMedia	CL311-M1	WIN	PCIe Gen2 x4	4
AVerMedia	CE511-MN	WIN	PCIe Gen2 x4	4
AVerMedia	CE314-SN	WIN	PCIE Gen1 x 4	4
Hauppauge	1609	WIN	PCI Express1x	4

Table 2: My caption

P/I	**Dtt**	**Sat Tv**	**Resolution**	**Store**	**Preprocessing**
99.95	Yes	No	1080P	MPEG-2	Picture in Picture
83.58	Yes	No	HD	MPEG-2	Color Space Conversion; Down Scaling; De-interlacing
Discontinued	Yes	No	HD	No	3D comb; Dual hardware MPEG-2 encoders
32.50	Yes	Yes	HD	No	Recoder
42.5	Yes	Yes	1080P	MPEG / TS	On-screen display
New	Yes	Yes	2160P	YUY2; YVYU; UYVY	Color Space Conversion; Down Scaling; De-interlacing
New	Yes	Yes	2160P	PCM	Color Space Conversion; Down Scaling; De-interlacing
371.00	Yes	Yes	1080P	YUV 4:2:2	Color Space Conversion; Down Scaling; De-interlacing
24.75	Yes	No	HD	MPEG-2	No

Table 3: Comparison of TV Tuners-3

Time-shifting	**FM Tuner**	**Remote Control**	**Development Ability**
Yes	Yes	Yes	No
Yes	No	Yes	Yes
No	Yes	Yes	No
Yes	Yes	Yes	Yes
Yes	No	Yes	Yes
No	No	Yes	Yes
No	No	Yes	Yes
No	No	Yes	Yes
No	No	Yes	No

3 FFmpeg

FFmpeg [3] is a leading multimedia framework, able to decode, encode, transcode, mux, demux, stream, filter and play pretty much anything that humans and machines have created.

With a simple command line, user can realize most of video processing. Such convenience is based on the core - FFmpeg program itself. It's designed for user to employ it easier. As for developer, FFmpeg provides a series of powerful libraries to design their own video processing programs. They aren't only able to implement the command line's functions, but also can go further such as reading and processing the live video stream.

On this project, we've used several libraries of FFmpeg to read and postprocess a digital TV live stream. For better understanding of our program, we will introduce the used libraries as examples to demonstrate FFmpeg.

Libavcodec : This library is about the coding of multimedia (audio, video and so on). It provides a number of coding functions with which the developers can encode and decode their multimedia. Also, it offers a common framework like H.264 for multimedia coding. We've used one of its header - "avcodec.h" to register all possible codecs and decode the live video stream.

Libavformat : "Libavformat" is mainly about multiplexer (muxer) and demultiplexer (demuxer) of multimedia. It contains most of multiplexing and demultiplexing functions for users to develop their multimedia format. By using its "avformat.h", we register all possible video formats and open the input video stream. Then, we retrieve stream information to print on terminal. In especial, it helps us to access the resource with several protocols at the same time.

Libswscale : It provides two pivotal process of multimedia - "Scaling" and "Conversion". a. The scaling indicates the change of image (frame in a video stream) size. This library provides a few scaling algorithms which have already been improved a lot and preferences to developers. b. The conversion here means the conversion of image color place like from Color to Gray and pixel format such as RGB to CMYK. According to our demand, we convert the stream from its original format to our target format - "AV_PIX_FMT_GRAY8" and down-scale from native size to "9x8". By "swscale.h", we also achieve converting it to RGB24 packed or any format we want. Furthermore, we've succeeded in extract any part of our live stream.

Libavutil : This is a genetic library which gives major programming utility tool function such as allocation and mathematic method. To uniform our programming standards and avoid memory leak, we use it to initialize and release our variables.

4 Conception

A video can be considered as a stream containing continuous frames. There are approximately 20 - 28 frames in one second. Thus, we need to calculate how long our processing will take for each frame and to ensure that it won't disturb its normal transmission. In order to clarify the processing capacity of our program, we firstly measure the response time of different video revolutions. Then, we seek the maximum channel quantity for each revolution. At last, to display better the multi-channels processing, we choose the detection of duplicate video streams as our example.

4.1 Class diagram

Each channel diffused by TNT is independent. Hence, we use multi-thread to handle their own parallel processing. To manage these threads better, we write a class "CMultiVideoProc" to parse the pass parameter in its constructors and generate respective thread. After that, we use its public method to invoke all threads.

There is only one task in each thread. It's to read its video stream and do its post processing. So we make a class "CVideo" to deal with all the mission about video stream. In this way, it's enough for each thread to declare one "CVideo" and call its task method. This is a big class. We give a lot of functionalities such as frame conversion, behavior record, hash template generation, video detection, etc to this class. To realize its various tasks, we add a list of classes components and their functions are as follows :

• CTask : Handles video processing's task information

• CCodec : Handles video processing's codec information like video format, decoder and encoder

• CStream : Catches video stream and decomposes its packets to get the context

• CResponseTime : Calculates the response time of each frame processing such as conversion, down-scaling and detection

As we said in last paragraph, the processing of video stream in fact is a processing of continuous frames. Therefore, we create a class "CFrame" to process the frame. For each frame in our video stream, beyond the basic colour conversion and down-scaling, we will use it to do our own processing. In other word, this class will be declared and used 20 - 28 times per second in video stream. Besides the template generation and duplicate detection, we add some popular simple functions (frame rotation, frame inversion, etc) in this class.

The diagram of our class in the end is designed as follows :

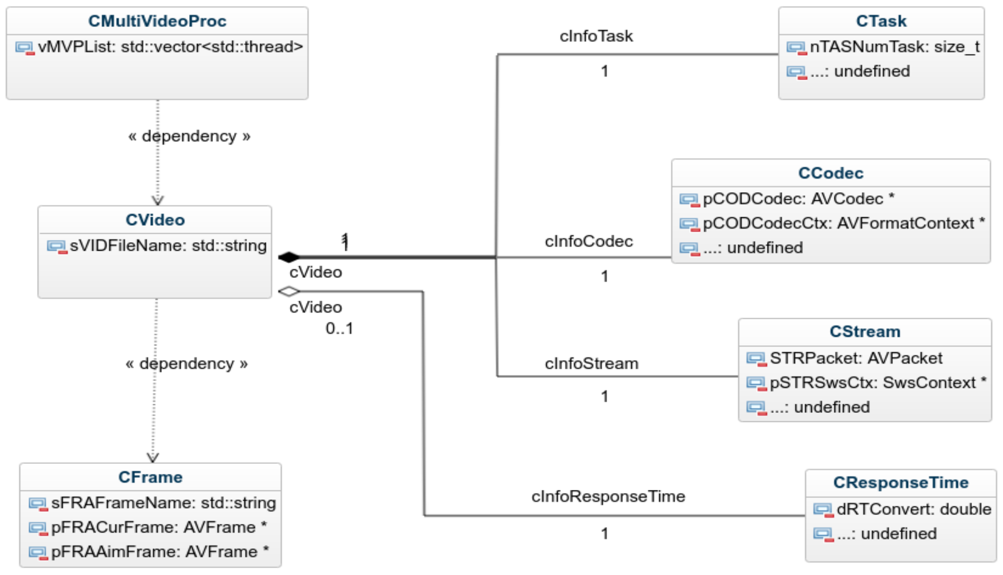

Figure 4: Class Diagram

Class "CMultiVideoProc" depends on class "CVideo" because it has several "CVideo" to start its respective input stream (the number of "CVideo" is same as the number of threads). And the association between "CVideo" and "CFrame" is also considered as dependency. Each time we process a new received frame we declare a new "CFrame" (the number of "CFrame" is same as the number of frames in the video stream). The classes including "CTask", "CCodec" and "CStream" are the composition of "CVideo". Nevertheless, class "CResponseTime" is the aggregation. We use it only when we want to collect the response time.

4.2 Program Flow

Our multi-video streams detection can be composed of two part. The first is to analyse template video by which every video stream can establish a pattern vector. This part is an unique and initial phase. Next we create a corresponding number of threads to read the input video streams. In the end, we invoke all the video processing threads.

These two part have only one difference that is the content of processing which will be talked in next subsection. We explain first how we use "FFmpeg" to read the video stream. Then we present the detail of processing.

As we discuss in introduction, this multi-channel processing is finished by TV tuner and computer program. We consider that the TV tuner part will be designed for better second development. Thus, we separate it into three parts (Stream Reading, Postprocessing and Duplicate Detection). The first part and the second will be processed in a developable TV tuner. And the last part is considered to be finished in the computer.

4.2.1 Stream Reading

Library of "libavformat" lets us have the ability to initialize video streams. We firstly use it to register all formats and codec. Then it open our video streams. At the mean time, it retrieve our streams information. To handle the exception, we dump the information onto standard error. Next it find the first video stream and get a pointer to this codec context so we can use the corresponding decoder. Copying codec context is the next step. At last, it allocate a dynamic pointer to the current video frame. After the initialization, it use an infinite cycle to read our frames stream to "AVPacket". In this cycle, we need to distinct the video stream because in a live TNT TV stream, there is also the audio stream. Then it decode the video frame and when this is finished, we begin to do our processing.

4.2.2 Postprocessing

Before we start the duplicate detection, we do a postprocessing for our raw frames. Each frame can be any format. This processing transforms it from its original format to GRAY8 frame (step 1). The response time here we consider it as T1. Then it down-scales the GRAY8 frame from its native size to ”9x8” (step 2). The response time here we consider it as T2. At last we return the target frame.

According to our demand, we need to process all of it in real time. We create a limit of total response time for this postprocessing - 1 / 30 (s) because there are 20 - 28 frames per second. The concept is as follows :

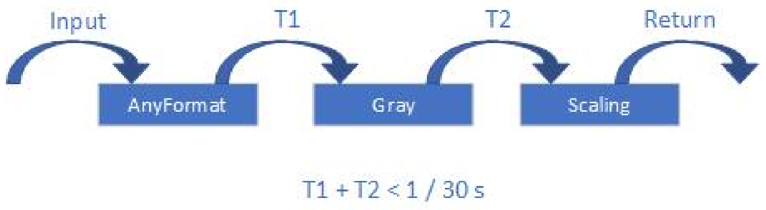

Figure 5: Postprocessing

4.2.3 Duplicate Detection

As we consider this part will be finished in the computer, the whole detection response time also need to be within 1 / 30 (s). And it consists of two parts (Template Generation and Recognition). The detail will be showed in next subsection. The final program flow is described in the following picture.

5 Experiment

Considering that our objectives are double, we compare the performance of different TV tuners cards at first (multiple channels PCI in particular for the fixed machine). Then we read multiple video streams at the same time, simulating multiple channels, and do a sequence of post processing in real time. For the first part, we've given the comparison in section 2.3. In this section, we mainly show how we make it come true and its related results of implementation will be demonstrated in the next section.

[11] [10] [12]

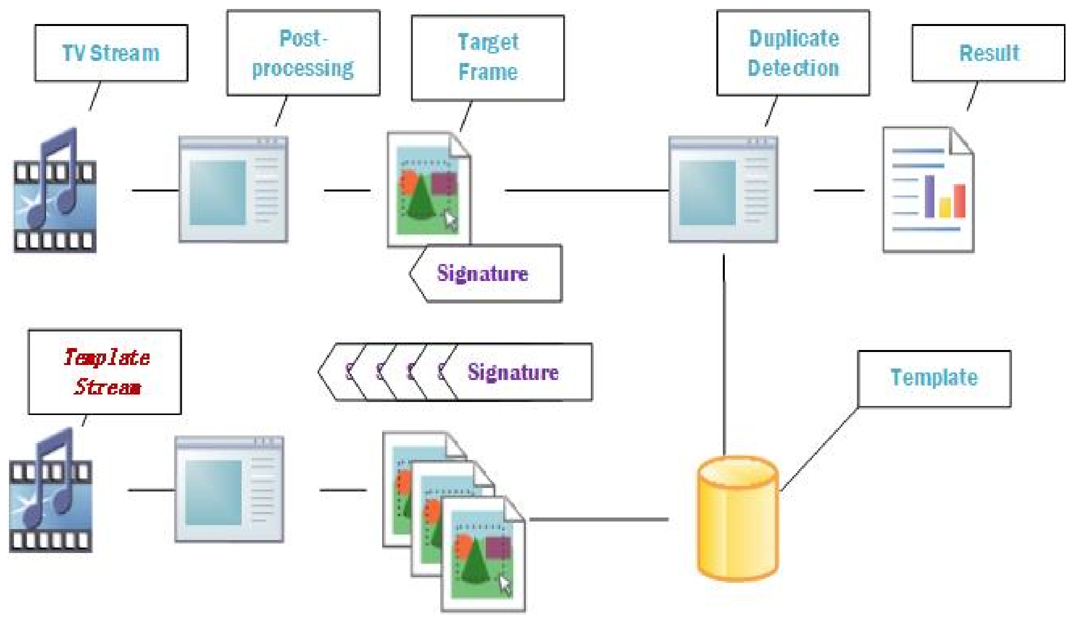

Figure 6: Duplicate Detection

5.1 Template Generation

Our templates for detection is a set of signatures. Each signature consists of 16 hexadecimal characters. This string set represents the feature of the template video that we want to find out its duplicates. We will generate this string set before our multi-video streams reading. And each video stream will be compared with the template while its reading. The reason why we use this to represent the template feature has been talked in the report [9]. Here we base on their work to do our algorithm and make a little optimization on it.

To simplify our detection example, we extract one frame from the template video stream per 5 second. We do a down-scaling and colour conversion for the frame. The size destined is 9x8 and it will be black and white. After that, we generate its corresponding character string. For that, we assign a bit to each pixel except those of the last column. So that is 8x8 or 64 bits. We give the value 1 to pixels lighter than their neighbor on the right and 0 otherwise. We write the bits from left to right, then from top to bottom. Once they are converted to hexadecimal, the feature string is 16 characters long. At last, we store these template string in a specific file.

5.2 Recognition

The post processing for duplicate detection is very similar to its template generation. Before that, each stream reading thread needs to load the template file with its feature set. Then we read the frame one by one in video stream and do the same post processing (black and white in size 9x8). Also, we generate its corresponding 16 hexadecimal characters string. The difference to the last operation is that we compare this string with the template set. If there are more than certain times that it already exists in the set, we consider this video stream as a duplicate.

5.3 Multi-threads

Our processing can not only deal with one TV channel but also multiple channels in the same time. The limit of response time stands still but the quality of processing has been improved. The final detection example is as follows :

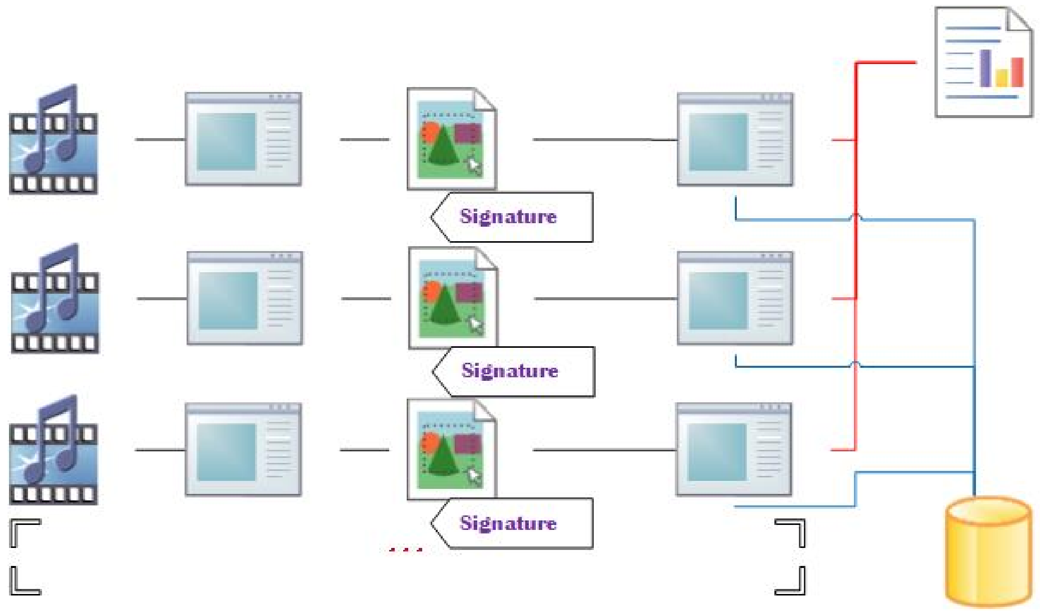

Figure 7: Multi-channel Duplicate Detection

5.4 Optimization

Differ from the algorithm we've referred in [9], we use the frames each certain seconds and recognize the duplicate only when all template frames are corresponding. In this way, we avoid some mis-predication cases like two streams are in the same proprietor.

The popular way to process a frame is to operate a self-generated corresponding 2D pixel table. In this case, the time complexity is O($3N^2$) - one N^2 for generation, one N$\hat{2}$ for operation and one N^2 for refreshing frame. This way is easy and simple for developers to process the frame. But when it comes to a big frame size, it takes a long time. So we do some amelioration to improve the time complexity as O(N^2), in which we skip the generation of pixel table. We directly allocate a new frame data pointer to store the final pixel data. Then we free the old raw frame data pointer and cover its position.

We are aware that the template strings are only used for comparison. So we make a little optimization on it. We directly use its raw pixel comparison result (64 bits) as a unit of feature set. It will avoid the convert calculation part. Also, while our last detection, it makes the judgement after certain times positive comparison. To maximize the use of template string, we can calculate the hash distance to the current characters string. In this way, we can not only detect the duplicate but also the similarity.

6 Result

As we've already discussed with the limit time of response, we here demonstrate our tests on the relation between number of channels (postprocessing and duplicate detection) and its corresponding response time (RT).

- Line Fps : the maximum response time that the processing could spend
- Line Avg : the average response time for all the channels
- Line Max : the maximum response time for all the channels

6.1 Postprocessing

With this table (Fig 8), we can easily observe the maximum number of channels our postprocessing can deal with for each resolution. 25 channels maximum for 4K TV streams processing and 100 maximum for 1080P (HD). In the wake of stream quality, the maximum number brings down. However, in the case of France where we write this thesis (TNT TV signal), there are approximately 60 channels. And there are usually only about 10 channels have high quality which means this postprocessing well past its demand.

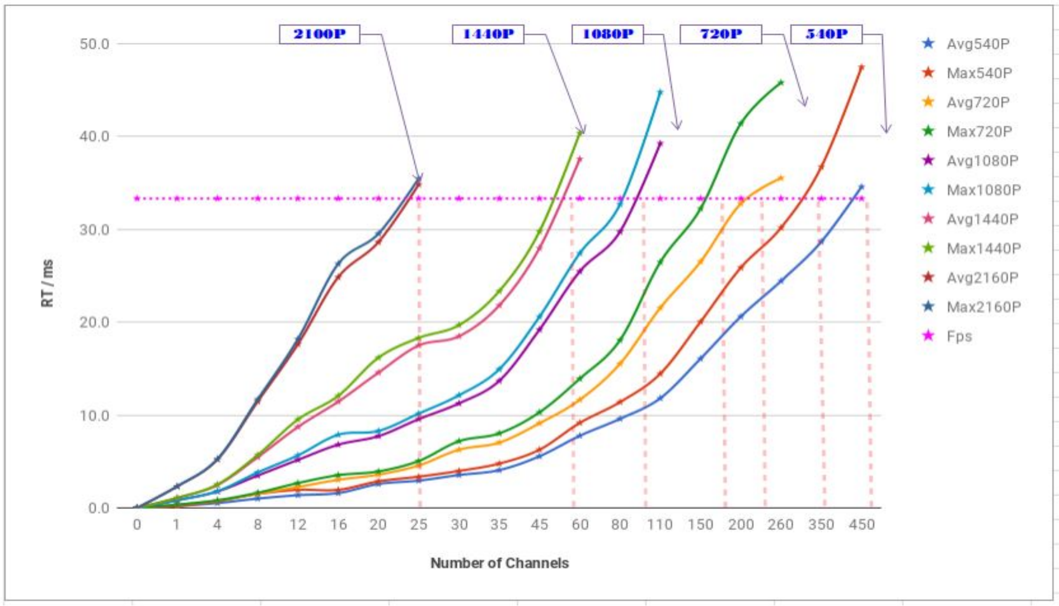

Figure 8: Response Time - Postprocessing

6.2 Duplicate Detection

With this table (Fig 9), we can easily observe the maximum number of channels our duplicate detection can deal with for each resolution. And this result show us that its capability is far surpass the reality. Even for the whole Europe, there are only approximately 600 channels. And our detection can easily finish its job (≈15 ms).

The line ”Before” indicates the algorithm that is before our improvement and the line ”After” is the after. We can observe that with the increment of channels, its distance also augment which means the percentage we've optimized is increasing.

7 Conclusion and Future Work

In this work, we study in multi-channel processing that consists of comparison of different TV tuner, stream reading, postprocessing and duplicate detection and prospect its future functionality - synchronization of ads, demonstration of film detail. As for caparison, we search in the part of postprocessing and detection, we focus on its real time processing ability which means each frame can't be processed over the limit time. We take thousands of test to get the response time table and make a comparison for our two algorithms.

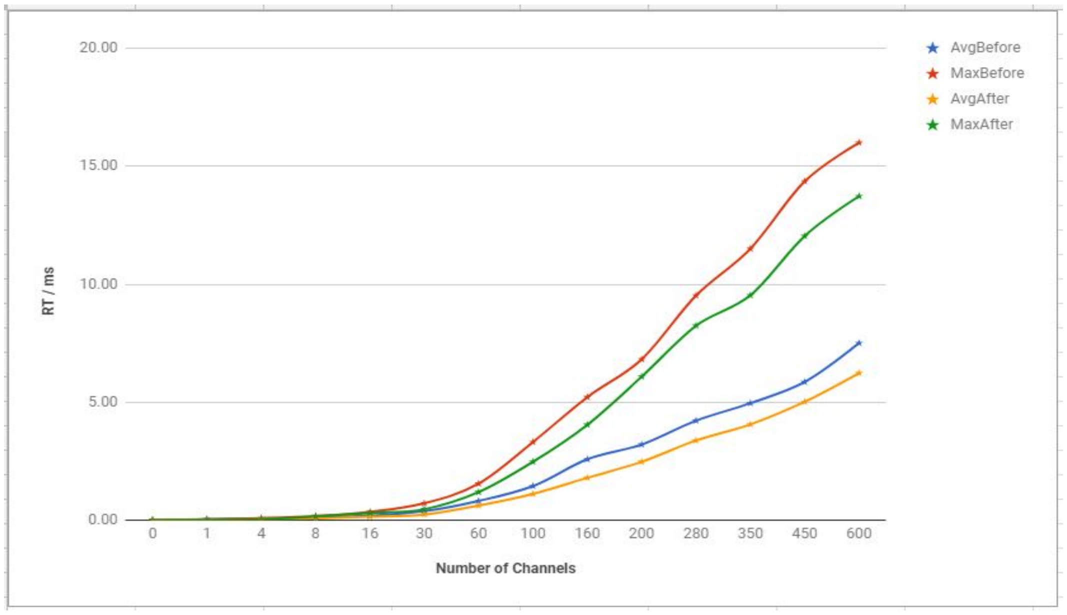

Figure 9: Response Time - Duplicate Detection

There are many places still can be improved in this work which makes us more impetus to study - the extraction of signature for template video, the recognition of current stream and so on. After that, we could make our own application / software to realize the future functionality we've talked.

References

[1] Analog broadcasts investigation. `https://en.wikipedia.org/wiki/Digital_television_transition#Asia`. Accessed: 2018-06-03.

[2] Digital terrestrial television. `https://en.wikipedia.org/wiki/Digital_terrestrial_television`. Accessed: 2018-06-03.

[3] Ffmpeg. `http://ffmpeg.org/about.html`. Accessed: 2018-06-03.

[4] Pci express. `https://en.wikipedia.org/wiki/PCI_Express`. Accessed: 2018-06-03.

[5] Stick pc. `http://www.tomshardware.fr/articles/media-center-mini-pc-windows,2-2459.html`. Accessed: 2018-06-03.

[6] Syncnization of two screens. `https://www.techspot.com/news/63067-battle-second-screen.html`. Accessed: 2018-06-03.

[7] Tv tuner card. `https://en.wikipedia.org/wiki/TV_tuner_card`. Accessed: 2018-06-03.

[8] Usb 3.x. `https://en.wikipedia.org/wiki/USB#USB_3.X`. Accessed: 2018-06-03.

[9] L. Lefaucheux et T. Fournier. Algorithme de detection de duplicata sur des videos en flux live. Parcours des ecoles d'ingenieurs Polytech, 2018.

[10] G. Kaur et all S.S. Saini. High gain reduced ground terahertz microstrip patch antenna design for the detection of trinitrotoluene (tnt) explosives material. 2017 Progress In Electromagnetics Research Symposium - Spring (PIERS), 2017.

[11] L. Beloica et all T. Mastilovic. Spherical video player implementation using ffmpeg and sdl program support. 2017 25th Telecommunication Forum (TELFOR), 2017.

[12] T. Zobel. The digital pc tv card. IEE Seminar on HomeNet.TV Plus, 1999.