# 【Android 音视频开发打怪升级：音视频硬解码篇】一、音视频基础知识 - 简书

简 jianshu.com/p/1749d2d43ecb

## [Android audio and video development and upgrade: audio and video hard decoding] 1. Basic knowledge of audio and video

Today, short video apps are booming and prosperous. With the rise of short videos, audio and video development has received more and more attention. However, because audio and video development involves a wide range of knowledge and the entry threshold is relatively high, many developers are daunted.

Although there are many blog posts on the Internet that summarize the route of audio and video fighting monsters, the knowledge related to audio and video development is relatively independent, some talk about "audio and video decoding related", some talk about "OpenGL related", and some talk about "FFmpeg related" "Yes", but it is very difficult for a novice to connect all the knowledge and understand it well.

In the process of learning audio and video development, I deeply realized the confusion and pain caused by the dispersion of knowledge and transitional faults. Therefore, I hope that through my own understanding, I can summarize the knowledge related to audio and video development and form a series. The article, step by step, analyzes each link, one summarizes and consolidates what I have learned, and the other hopes to help developers who want to get started with audio and video development.

**Tutorial code: [ Github Portal ]**

### Table of contents

### In this article you can learn

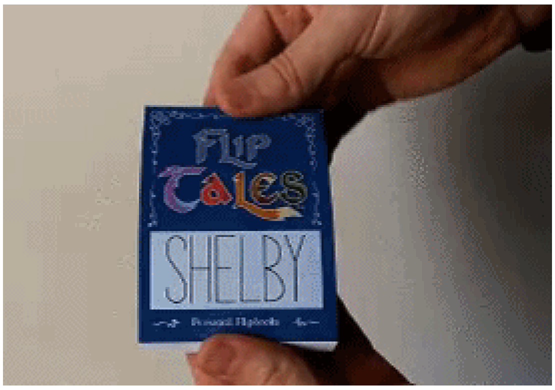
> As the opening article, let's take a look at what audio and video are made of, as well as some common terms and concepts.

### 1. What is video?

Flipbook

I don't know if you have played an animated little book when you were young. When you flip it continuously, the picture of the little book will become an animation, similar to the current gif format picture.

Flipbook: Source Network



Originally a static little book, after flipping it, it will become an interesting little animation. If there are enough pictures and the flipping speed is fast enough, this is actually a small video.

The principle of video is exactly the same. Due to the special structure of the human eye, when the picture is switched quickly, the picture will remain, and it feels like a coherent action. So, **a video is made up of a series of pictures** .

video frame

A frame is a basic concept of video, which means a picture, such as a page in the flip-flop book above, is a frame. A video is made up of many frames.

frame rate

Frame rate, that is, the number of frames per unit of time, in units of frames per second or fps (frames per second). For example, in a flipbook, how many pictures are included in one second, the more pictures, the smoother the picture and the more natural the transition.

The frame rate is generally the following typical values:

24/25 fps: 24/25 frames per second, normal movie frame rate.

30/60 fps: 30/60 frames per second, the frame rate of the game, 30 frames is acceptable, 60 frames will feel more smooth and realistic.

Above 85 fps is basically invisible to the human eye, so higher frame rates don't make much sense in video.

color space

Here we only talk about two commonly used color spaces.

RGB

The RGB color mode should be the one we are most familiar with, and it is widely used in today's electronic devices. Through the three basic colors of RGB, all colors can be mixed.

YUV

Here we will focus on YUV, this color space is not familiar to us. This is a color format that separates luminance and chrominance.

Early TVs were all black and white, i.e. only had the luminance value, ie Y. With the advent of color TV, two chromaticities of UV were added to form the current YUV, also called YCbCr.

Y: Brightness, which is the gray value. In addition to representing the luminance signal, it also contains a larger amount of green channel.

U: The difference between the blue channel and the luminance.

V: The difference between the red channel and the luminance.

**What are the advantages of using YUV?**

> Human eyes are sensitive to brightness but not to chroma. Therefore, the amount of UV data is reduced, but the human eye cannot perceive it. In this way, by compressing the resolution of UV, the volume of the video can be reduced without affecting the look and feel.

RGB and YUV conversion

```
Y = 0.299R + 0.587G + 0.114B
U = -0.147R - 0.289G + 0.436B
V = 0.615R - 0.515G - 0.100B
——————————————————
R = Y + 1.14V
G = Y - 0.39U - 0.58V
B = Y + 2.03U
```

## 2. What is audio?

The most commonly used way of carrying audio data is **pulse code modulation** , or **PCM** .

In nature, sound is continuous and is an analog signal, so how can the sound be preserved? That is to digitize the sound, that is, convert it into a digital signal.

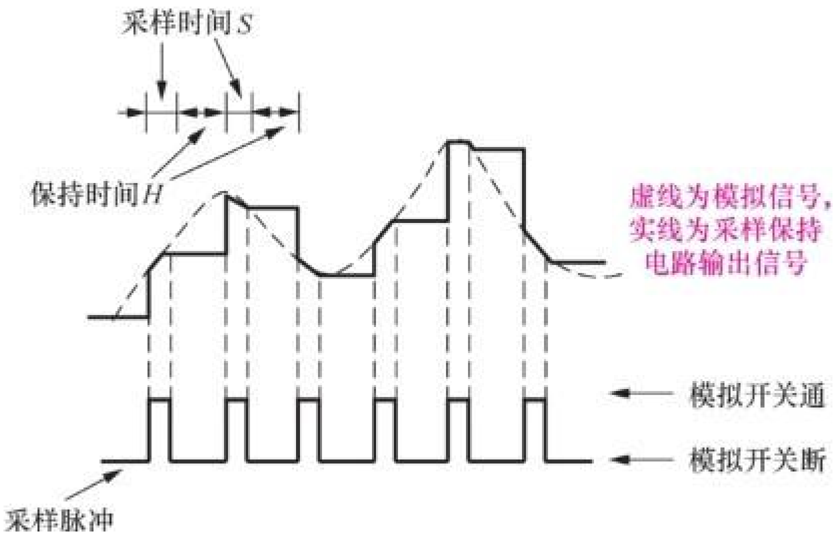
We know that sound is a kind of wave with its own amplitude and frequency, so to save the sound, we must save the amplitude of the sound at various points in time.

The digital signal cannot continuously save the amplitude of all time points. In fact, it is not necessary to save the continuous signal to restore the sound acceptable to the human ear.

According to Nyquist sampling theorem: In order to restore the analog signal without distortion, the sampling frequency should not be less than 2 times the highest frequency in the spectrum of the analog signal.

According to the above analysis, the acquisition steps of PCM are divided into the following steps:

> analog signal -> **sampling -> quantization -> encoding -** > digital signal



audio sample

### Sampling Rate and Number of Sampling Bits

Sampling rate, that is, the frequency of sampling.

As mentioned above, the sampling rate is greater than 2 times the frequency of the original sound wave, and the highest frequency that the human ear can hear is 20kHz, so in order to meet the hearing requirements of the human ear, the sampling rate is at least 40kHz, usually 44.1kHz, and higher usually is 48kHz.

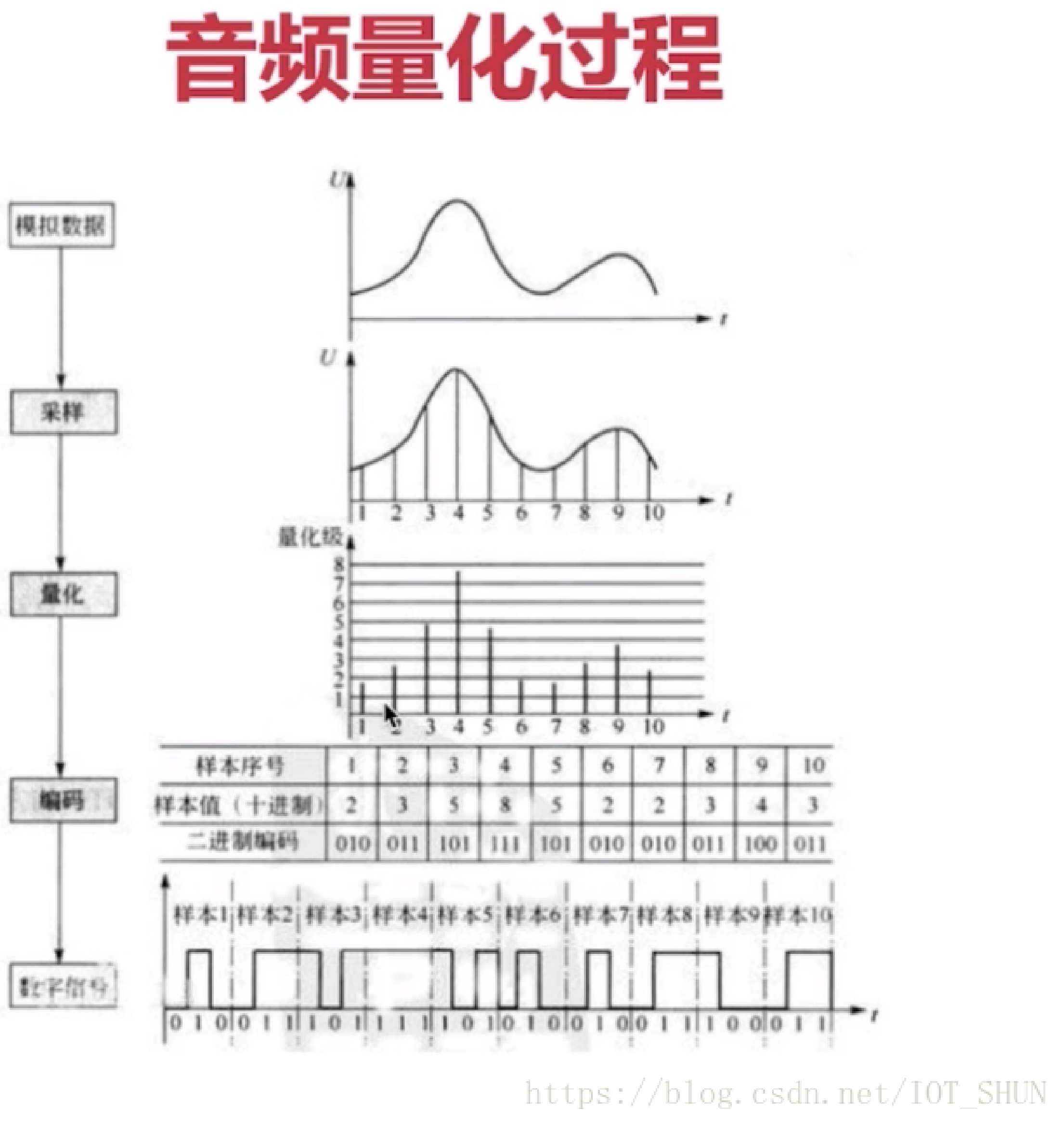
The number of sample bits, related to the amplitude quantization mentioned above. The waveform amplitude is also a continuous sample value on an analog signal, but in a digital signal, the signal is generally discontinuous, so after the analog signal is quantized, only an approximate integer value can be taken. In order to record these amplitude values, the sampler will use a A fixed number of bits is used to record these amplitude values, usually 8, 16, and 32 bits.

| number of digits | minimum | maximum value |
|---|---|---|
| 8 | 0 | 255 |
| 16 | -32768 | 32767 |
| 32 | -2147483648 | 2147483647 |

**The more digits, the more accurate the recorded value and the higher the degree of restoration.**

The last thing is coding. Since the digital signal is composed of 0 and 1, the amplitude value needs to be converted into a series of 0 and 1 for storage, that is, encoding, and the final data obtained is the digital signal: a series of 0 and 1 data.

The whole process is as follows:



| 样本序号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 样本值（十进制） | 2 | 3 | 5 | 8 | 5 | 2 | 2 | 3 | 4 | 3 |
| 二进制编码 | 010 | 011 | 101 | 111 | 101 | 010 | 010 | 011 | 100 | 011 |

number of channels

The number of channels refers to the number of speakers that support **different sounds** (note that they are different sounds).

Mono: 1 channel
Dual: 2 channels
Stereo: 2 channels by default
Stereo (4 channels): 4 channels

code rate

Bit rate refers to the amount of information that can pass through a data stream per second, in bps (bit per second)

Bit rate = sample rate * number of samples * number of channels

### 3. Why code

The encoding here is not the same concept as the encoding mentioned in the audio above, but refers to **compression encoding** .

We know that in the computer world, everything is made up of 0s and 1s, and audio and video data is no exception. Due to the huge amount of audio and video data, if it is stored as raw streaming data, it will consume a lot of storage space, which is not conducive to transmission. Audio and video actually contain a lot of repeated data of 0 and 1, so these 0 and 1 data can be compressed through a certain algorithm.

Especially in video, because the picture is gradually transitioned, the whole video contains a lot of picture/pixel repetition, which just provides a very large compression space.

Therefore, encoding can greatly reduce the size of audio and video data, making it easier to store and transmit audio and video.

### 4. Video coding

Video encoding format

There are many video encoding formats, such as H26x series and MPEG series encoding, these encoding formats appear to adapt to the development of the times.

Among them, H26x (1/2/3/4/5) series is led by ITU (International Telecommunication Union)

The MPEG (1/2/3/4) series is dominated by MPEG (Moving Picture Experts Group, an organization under ISO).

Of course, they also have jointly formulated coding standards, which is the current mainstream coding format H264, and of course the next-generation more advanced compression coding standard H265.

Introduction to H264 encoding

H264 is the most mainstream video encoding standard at present, so our follow-up articles mainly use this encoding format as the benchmark.

H264 is jointly customized by ITU and MPEG and belongs to the tenth part of MPEG-4.

> Since the H264 encoding algorithm is very complex, it cannot be explained clearly in a moment, and it is not within the scope of my current ability, so here is only a brief introduction to the concepts that need to be understood in daily development. In fact, the encoding and decoding part of the video is usually done by the framework (such as Android hard solution/FFmpeg), and the general developer will not touch it.

### video frame

We already know that the video is composed of one frame and one frame, but in the video data, it is not really saved according to the original data of one frame and one frame (if this is the case, the compression encoding is meaningless).

H264 will select a frame of picture as the complete encoding according to the changes of the picture within a period of time, and the next frame will only record the difference with the complete data of the previous frame, which is a dynamic compression process.

In H264, the three types of frame data are

**I frame** : Intra-coded frame. is a full frame.

**P-frame** : Forward predictive coded frame. is a non-complete frame, generated by referencing the preceding I-frame or P-frame.

**B frame** : Bidirectional predictive interpolation coded frame. Generated by reference to before and after image frame encoding. A B frame depends on the nearest I frame or P frame before it and the nearest P frame after it.

### Group of Images: GOP and Keyframe: IDR

Full name: Group of picture. Refers to a group of video frames that do not change much.

The first frame of the GOP becomes the keyframe: IDR

IDRs are all I-frames, which can prevent decoding errors of one frame and cause decoding errors of all subsequent frames. When the decoder decodes to the IDR, it will clear the previous reference frame and start a new sequence, so that even if there is a major error in the decoding of the previous frame, it will not spread to the following data.

> Note: Keyframes are all I-frames, but I-frames are not necessarily keyframes

### DTS and PTS

DTS full name: Decoding Time Stamp. Indicates when the data stream read into memory starts to be sent to the decoder for decoding. That is, the timestamp of the decoding order.

PTS full name: Presentation Time Stamp. Used to indicate when the decoded video frame is displayed.

> In the absence of B frames, the output order of DTS and PTS is the same, and once there are B frames, PTS and DTS will be different.

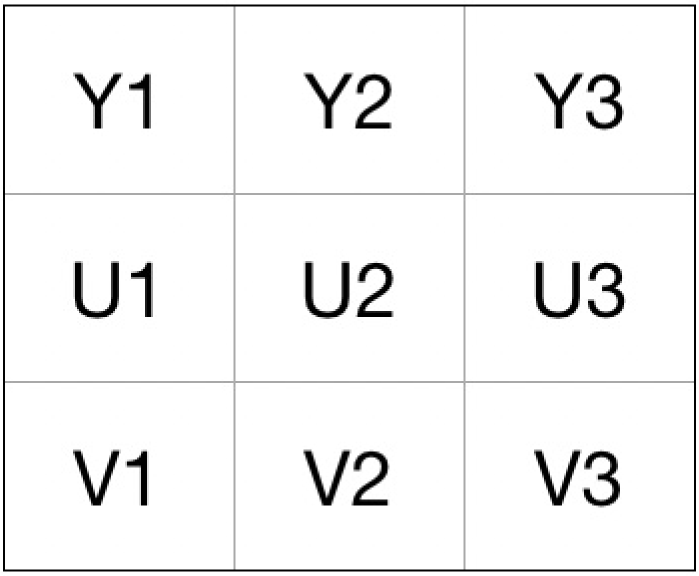
### frame color space

Earlier we introduced two image color spaces, RGB and YUV. H264 uses YUV.

YUV storage methods are divided into two categories: planar and packed.

> Planar: store all Y first, then store all U, and finally V;
> packed: Y, U, and V of each pixel are stored continuously and interleaved.

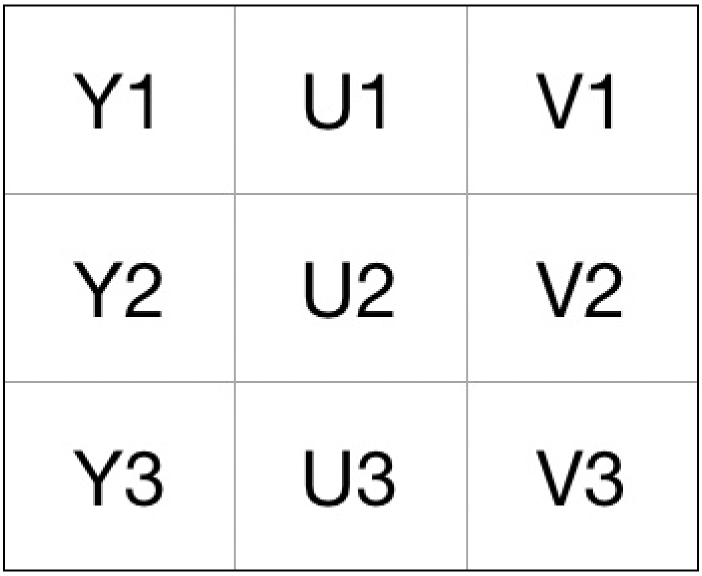
planar as follows:



| | | |
|---|---|---|
| Y1 | Y2 | Y3 |
| U1 | U2 | U3 |
| V1 | V2 | V3 |

YUV Planar

packed as follows:



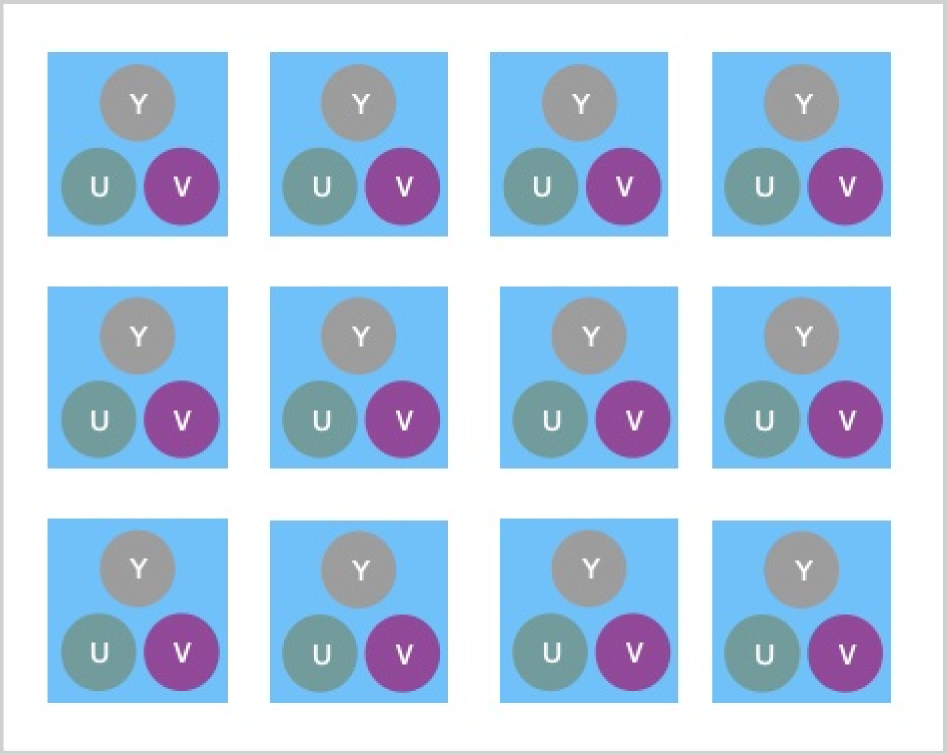
| | | |
|---|---|---|
| Y1 | U1 | V1 |
| Y2 | U2 | V2 |
| Y3 | U3 | V3 |

YUV Packed

> However, the pakced storage method is very rarely used, and most videos are stored in the planar storage method.

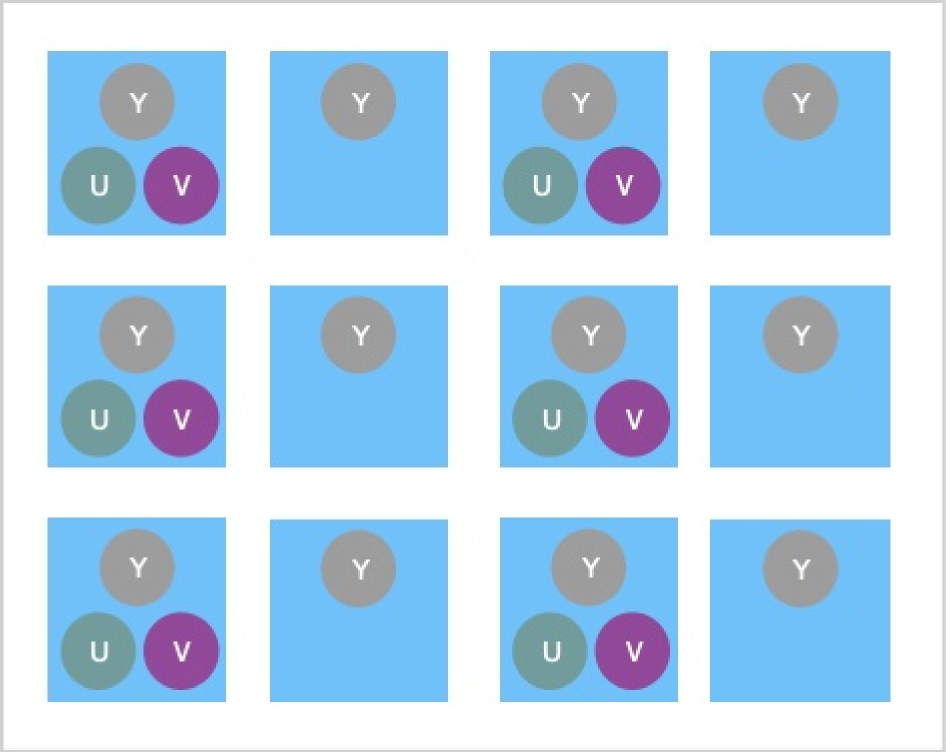
As mentioned above, due to the low sensitivity of human eyes to chrominance, storage space can be saved by omitting some chrominance information, that is, sharing some chrominance information with luminance. Therefore, planar distinguishes the following formats: YUV444, YUV422, YUV420.

YUV 4:4:4 sampling, each Y corresponds to a set of UV components.



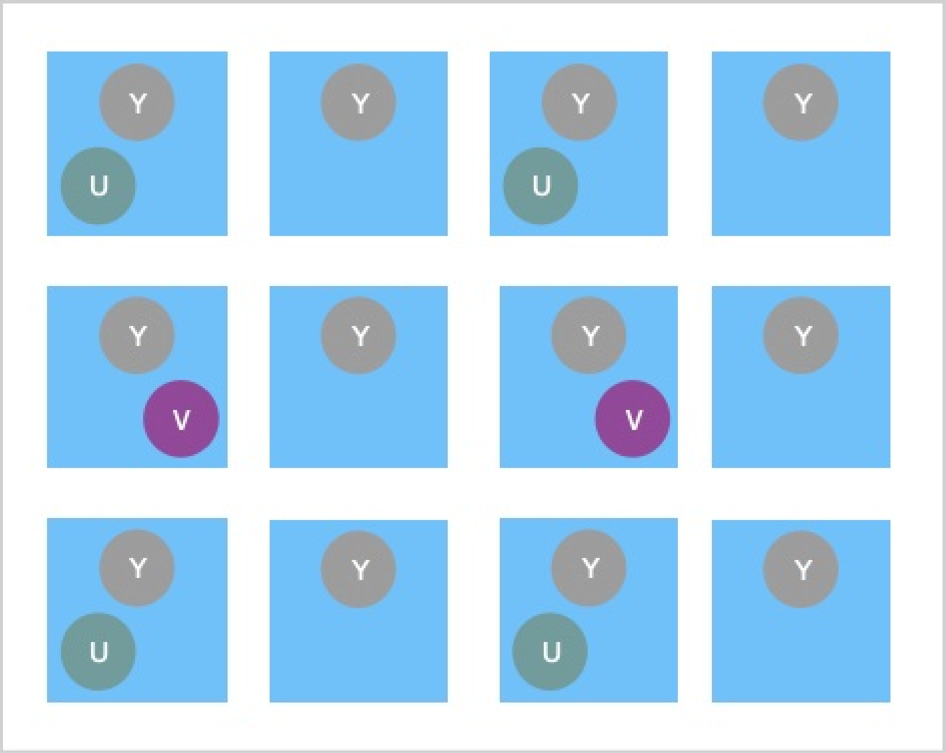
**YUV444**

YUV 4:2:2 sampling, each two Y shares a set of UV components.



# YUV422

YUV 4:2:0 sampling, each four Y shares a set of UV components.



YUV420

Among them, **the most commonly used is YUV420** .

### YUV420 format storage method

YUV420 belongs to the planar storage method, but it is divided into two types:

**YUV420P** : Three-plane storage. The data group is YYYYYYYYUUVV (eg I420) or YYYYYYYYVVUU (eg YV12).

**YUV420SP** : Two plane storage. Divided into two types YYYYYYYYUVUV (such as NV12) or YYYYYYYYVUVU (such as NV21)

> Regarding the encoding algorithm and data structure of H264, there is a lot of knowledge and space involved (such as the network abstraction layer NAL, SPS, PPS). This article will not go into details. There are also many tutorials on the Internet. If you are interested, you can learn it by yourself.

Getting started understanding H264 encoding

## 5. Audio coding

audio coding format

The raw PCM audio data is also a very large amount of data, so it also needs to be compressed and encoded.

Like video encoding, audio also has many encoding formats, such as: WAV, MP3, WMA, APE, FLAC, etc. Music enthusiasts should be very familiar with these formats, especially the latter two lossless compression formats.

However, our protagonist today is not them, but another compression format called AAC.

AAC is a new generation of audio lossy compression technology, a high compression ratio audio compression algorithm. Audio data in MP4 video is mostly in AAC compression format.

Introduction to AAC Coding

AAC format is mainly divided into two types: ADIF, ADTS.

**ADIF** : Audio Data Interchange Format. Audio data interchange format. The feature of this format is that the start of the audio data can be found deterministically, without decoding starting in the middle of the audio data stream, that is, its decoding must be performed at a well-defined start. This format is commonly used in disk files.

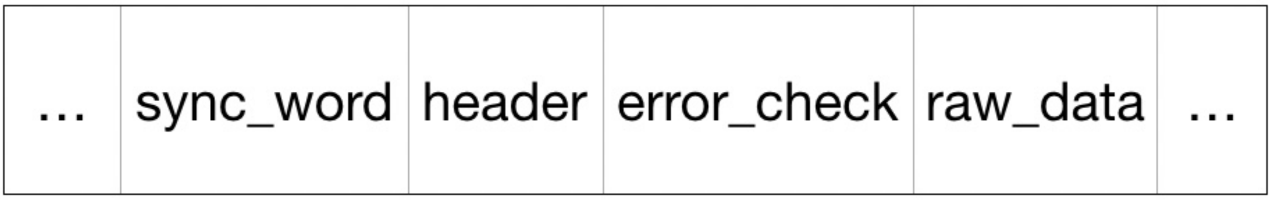
**ADTS** : Audio Data Transport Stream. Audio data transport stream. The characteristic of this format is that it is a bit stream with sync words, and decoding can start anywhere in the stream. Its characteristics are similar to the mp3 data stream format.

> ADTS can be decoded in any frame, and it has header information for each frame. ADIF has only one unified header, so it must get all the data and decode it. And the formats of the two headers are also different. Currently, the audio streams in ADTS format are generally encoded.

ADIF data format:

| header | raw_data |
|---|---|

ADTS ***one frame*** data format (the middle part, the left and right ellipsis is the data frame before and after):



| | | | | | |
|---|---|---|---|---|---|
| … | sync_word | header | error_check | raw_data | … |

ADTS

The internal structure of AAC will not be repeated. You can refer to the AAC file parsing and decoding process

## 6. Audio and video container

Careful readers may have found that none of the various audio and video encoding formats we introduced earlier are the video formats we usually use, such as: mp4, rmvb, avi, mkv, mov...

Yes, these familiar video formats are actually containers that wrap audio and video encoded data, and are used to mix video streams and audio streams encoded with a specific encoding standard into a file.

For example: mp4 supports video encoding such as H264 and H265 and audio encoding such as AAC and MP3.

> mp4 is the most popular video format at present. On the mobile terminal, the video is generally packaged in the mp4 format.

## Seven, hard decoding and soft decoding

The difference between hard and soft solutions

We will see in some players that there are two playback modes, hard decoding and soft decoding, for us to choose, but most of the time we cannot feel the difference between them. For ordinary users, as long as it can be played.

So what's the difference between them?

On a mobile phone or PC, there will be hardware such as CPU, GPU or decoder. Usually, our calculations are performed on the CPU, which is the execution chip of our software, and the GPU is mainly responsible for the display of the screen (it is a kind of hardware acceleration).

The so-called soft decoding refers to using the computing power of the CPU to decode. Usually, if the power of the CPU is not very strong, the decoding speed will be relatively slow, and the mobile phone may be overheated. However, compatibility will be good due to the use of a unified algorithm.

Hard decoding refers to the use of a special decoding chip on the mobile phone to speed up decoding. Usually the decoding speed of hard decoding will be much faster, but since hard decoding is implemented by various manufacturers, the quality is uneven, and compatibility problems are very likely to occur.

Hard decoding for Android platform

> Finally came to the part about Android, as the end of this article, it can be regarded as the beginning of the next article.

MediaCodec is the codec interface introduced in Android 4.1 (api 16) version, and it is a pit that all developers who want to develop audio and video on Android cannot avoid.