Mobile Video Editor

Sanket Tandulwadkar, Vivek Teke, Vaibhav Mahajan, Manish Motwani

sankettandulwadkar@gmail.com, vivekteke1991@gmail.com,

vaibhavmahajan5@gmail.com, manishmm4@gmail.com.

Abstract:

With the camera being an integral part of mobiles, these days capturing and exchanging videos has become banal. Hence video editing has become necessary and moreover famous since people usually want to trim the unwanted part of the video or edit the unwanted noise. MMS has changed the way video transfers took place. With the development of technology there have been many video editing platforms. Editing videos on PCs is fairly easy. However, there has been minimal research for video editing on mobile systems. With mobiles, there are a few limitations like decrease in the display screen, lesser storage area and RAM. In this paper, we discuss the basic idea behind editing a video. We discuss the movie atom container in detail and the method of locating a track in an mp4 file using functions. The process of editing a video by separating the frames and performing similar tasks with limitations of size and resolution is discussed in brief.

Keywords:

Video editing, mobiles, mpeg-4, H.264, metadata, movie atom.

Introduction:

Today, in the era of technology, mobiles and its different application are very vital in day to day life. Multimedia mobiles are the fundamental need of living. Cameras are embedded in mobiles and motion pictures can be captured and stored in devices. As we can store and capture video there is need to modify it. Videos consist of visual object and video frames and hence we should be able to remove a specific video frame from that video stream. If one wants to insert specific video clip, this also can be done.

Mobile:

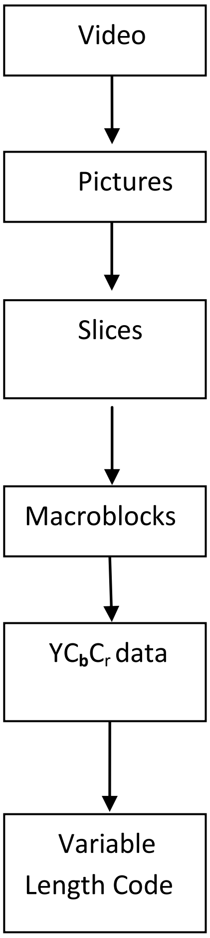
Video editing can be done much faster on the S60 phones. Symbian based mobile phones of the s60 v3 family are the suitors for this application.

Video:

MPEG-4 and H.264 are the most commonly used ISO file formats for mobile systems. MP4 videos are object based i.e. data is stored in boxes called objects. The container box contains the movie atom. The movie atom (moov) contains all the metadata such as video duration, size of the MP4 file in a sequential manner.

The color and luminance information is stored in this data structure. Manipulating this container is equal to manipulating the video content.

Flow of the information of a video:



With set theory, Let S be a system representing a video format for the mobile editor.

Then S = M

Where M = Set of movie atoms.

M = { mvhd, iods, trak }

Where

mvhd =movie header atom.

Iods = object descriptor.

Trak = container for a track or stream.

And trak = { tkhd, tref, edts, mdia }

where

Tkhd = track header.

Tref= track reference.

Edts = edit list container.

And edts = { elst }

where

Elst = edit list.

Mdia = media track information

And mdia = { mdhd, hdlr, minf }

where

Mdhd = media header.

Hdlr = media type handler.

Minf= media information container.

And Minf = { vmhd, smhd, hmhd, <mpeg>, dinf, stbl }

where

Vmhd = video media header.

Smhd = sound media header.

Hmhd = hint media header.

<mpeg> = mpeg stream headers.

Dinf = data information header.

And dinf= { dref }

where

Dref= data reference header.

Stbl = sample table atom.

And stbl = { stts, dtts, stss, stsd, stsz, stsc, stco, stsh, stdp }

Where

Stts = time to sample number map.

Dtts = decoding time to sample number map.

Stss = sync sample map.

Stsd = sample descriptions.

Stsz = sample size.

Stsc = sample to chunk offset information.

Stco = chunk offset.

Stsh = shadow sync.

Stdp = degradation priority.

Moov

- ➔ Audio chunk
  - audio sample 1
  - Audio sample 2

    :
  - Audio sample n
- ➔ Video chunk
  - Video sample 1
  - Video sample 2

    :
  - Video sample n

    Locating track to be modified:

Searching a track in an mp4 file is done with the help of data within the sample table atom.

i. $f(s,T) \rightarrow S$

   Where 'f' is the function to locate the track that contains the sample at time T and S is the set of samples at time T.

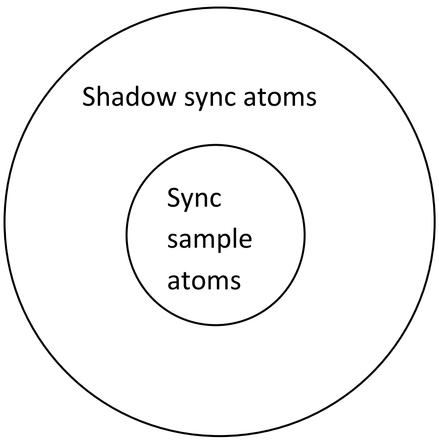
ii. $f(\sum SS_i, T) \rightarrow Sy_j$

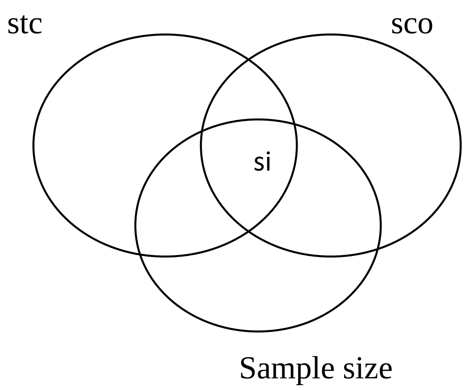
   where 'f' is the function to summate the shadow sync atoms(SS) which are nearly equal to the sync sample atoms(Sy).

iii. $Stc(Sy_i) \cup sco(c_i) \cup ssz(Sy_i) = S_i$

   Find the sample with the help of sample to chunk(stc), chunk offset(sco) and sample size(ssz).

Venn Diagrams:

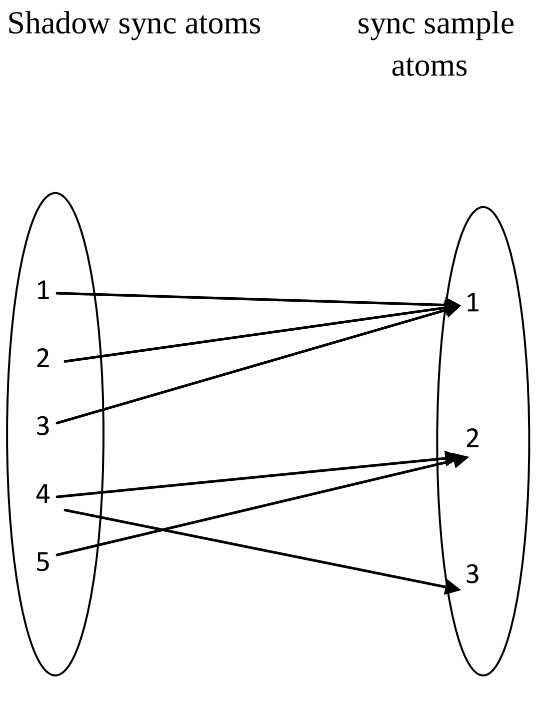


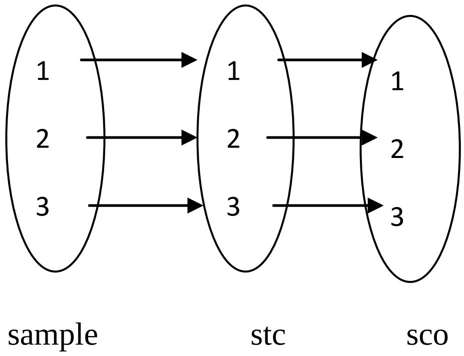


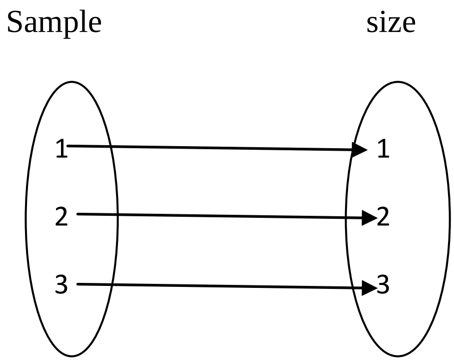
si = sample atom

stc = sample to chunk

sco = sample chunk offset

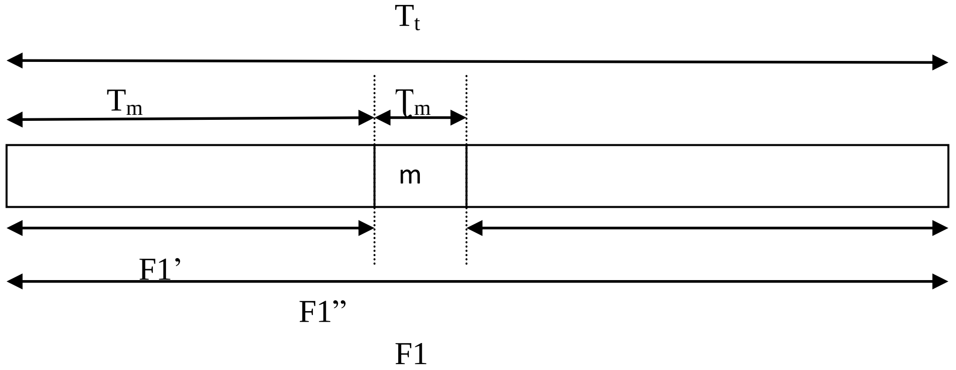






Algorithms:

1. <u>Cutting algorithm:</u>



Input: Frame of a video from which a part is to be removed.

Let

T_t = time duration of K frame.

τ_0 = time stamp for first frame.

T_m = time stamp for mth frame.

τ_m = duration of mth frame.

F1’ = Formed frame 1.

F1” =Formed frame 2

Translation factor:

$$T_m = \sum_{i=0}^{m-1} \tau_i$$

Output: Two frames are formed after the removal of the part to be cut.

Let

V1 = Video 1

V2 = video 2

T_m = time stamp of last frame in V1

T_n = timestamp for first frame in V2

Tv1= duration of V1

τ_m = duration of mth frame

Therefore,

$\tau_m = Tv1 - T_m$

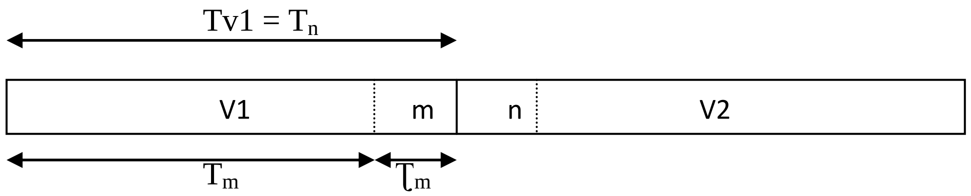
After splicing,

$T'_n = T_m + \tau_m$

$T'_n = Tv1$

Therefore timestamp of remaining part of video

2. Splicing algorithm:



Input: Two videos of same format and resolution.

$T'_k = \tau'_n + T_k$

Where T_k =timestamp of the rest k frames.$\tau'_n = T'_n - T_n$

Output: Spliced video.

Conclusion:

In this paper, we have mentioned the method of finding the sample atom. We also have stated the structure of movie atom, which is one of the main containers of a video. The basic aim of modifying the video as wished by the user is met by Mobile Video Editor. The video editing at lower resolution and with lesser storage and memory will usher in a new era of mobile based video editing.

References:

1] ISO/IEC JTC1/SC29/WG11 MPEG98/N2612p4 – Coding of Moving Pictures and Audio.

2] Video and Audio Editing for Mobile Applications. By - *Ari Hourunranta, Asad Islam, Fehmi Chebil.*

3] "An Optimized Method and Implementation for
Parsing MP4 Metadata." Lina Zhao, Lili Guan, Dept. of information management, Shanghai Finance University Shanghai, China.

4] "The MPEG4/AVC standard: description and basic tasks splitting."Isabelle Hurbain.