

MythTV videófeldolgozás felsőfokon

Kifinomult módszerek a lejátszás közbeni deinterlacing megvalósítására, és a videó hordozhatóvá tételére.

Stabilitása és kiterjeszhetősége miatt a *Linux* gyakran megtalálható a számítástechnika határterületein. Ígéretes platformnak mutatkozik a házimozzi audiovizuális alkalmazásai esetében is. Saját tévéhasználatomat jelenleg egy speciális célú *Linux PC* vezérli, amin *MythTV* fut. A *MythTV* telepítésekor mindkét fő probléma, amivel találkoztam, a videófeldolgozáshoz kapcsolódott. Az első kihívás a sima videólejátszás konfigurálása, a második a felvett műsorok hordozhatóvá tétele volt.

Deinterlacing videólejátszás közben

Napjaink elektronikájának korlátai miatt a televíziós képkockák két külön félkép formájában kerülnek adásra. Egy félkép a képek vagy a páros, vagy páratlan számú vízszintes sorából áll. Lejátszásnál a félképek páros és páratlan számú sorai összefűlésre kerülnek, amit a képernyőtől kellően távol levő néző folyamatos mozgásként érzékel.

Két egymást követő félkép összefüggő, de nem azonos. A kamera gyors oldalirányú mozgása közben egy adott félkép kicsivel előtte lesz az öt megelőzőnek, és egyetlen kontúrok jelenhetnek meg a képernyőn gyorsan váltakozó képeken. Az 1. ábra egy 1080i videómodot használó, nagy felbontású adás képét mutatja. A jelenetben a kamera balról jobbra pásztáz, ezért a kép objektumai gyorsan mozognak a képernyőn. Minden egyes félkép kicsit eltérő pozícióban van, ami fűrészfog-szerű torzulást okoz, amit fűsödöttségnek, csipkézetttségnek, illetve fogazottságnak is hívnak. Azoknál a jeleneteknél, ahol nagy az oldalirányú mozgás, a torzulás miatt

hihetetlenül nehezen követhetővé válhat a műsor tartalma. Ahhoz, hogy a 1. ábrán láthatóhoz hasonló videót nézhetővé tegyük, sima képpé konvertálható egy deinterlacing néven ismert eljárással. A *MythTV* felhasználók a deinterlacing számos módszere közül választhatnak:

- **Egy félkép** – Ahelyett, hogy két félképből állítana össze egy képkockát, ez a rendkívül egyszerű módszer csak az egyiket használja fel. Minden második félkép állóképként jelenik meg, a nem használtakat pedig figyelmen kívül hagyjuk.
- **Lineáris** – Ez a módszer összekeveri a közvetlenül szomszédos sorokat, amelyek váltakozó félképekből származnak. Enyhe szellemkép tűnhet fel, de a fűrészfog-szerű torzulás megszűnik.
- **Kernel** – Az előzővel ellentétben nem csak a szomszédos sorokat, hanem számos más sort is összekever. Nincs szellemkép, bár az egyik félképből visszamaradhatnak halvány részletek.
- **Bob** – Ez a legbonyolultabb módszer. Minden egyes félkép sorai megduplázódnak, majd az így létrehozott képkockákat dupla sebességgel játsszuk le.

A deinterlacing jelentős számítási teljesítményt követel, de a legtöbb modern *CPU* rendelkezik multimédiás utasításkészlettel, ami csökkenti a processzorra jutó terhelést. Ha *Intel*

processzort használunk *MMX* vagy *SSE* utasításokkal, vagy *AMD* processzort *3Dnow!* támogatással, a deinterlacing nem jelenthet gondot. A *Bob* a szinkronizált TV kimenet előállítására alkalmas legjobb deinterlacing eljárás, bár egy kevésbé fejlett gépen nehézségeket okozhat. A saját *MythTV* rendszerem egy 2 GHz-es *AMD Athlon64* processzort rejt, ami bőven elég *Bob* módszerrel deinterlace-elt nagy felbontású videó megjelenítésére egy analóg TV-n. Bár a *CPU* terhelése nagyobb, mint más deinterlacing módszerekénél, még mindig bőven a rendszerem teljesítőképességén belül van. A lineáris és a kernel deinterlacing alkalmazásával hasonló képi hatás érhető el, míg az utóbbi kissé jobban igénybe veszi a *CPU*-t. Kevésbé erőforrás-igényesek, mint a *Bob*, ami jól jöhet gyengébb *CPU* esetén. A kettő közül személy szerint a kernel deinterlacinget kedvelem jobban, mert több szomszédos sort kever és kiküszöböli a szellemképet, így a kapott kép élesebb lesz.

Videó exportálása más rendszerekbe

Az egyik alapvető oka annak, hogy felállítottam egy *MythTV* rendszert, az volt, hogy a tévéműsoraimat szerettem volna utazáskor magammal vinni. Most a „televízió” számomra fájlokat jelent a merevlemezben, így sokkal könnyebb tárolni, szállítani, és ott nézni, ahol éppen kényelmes. Mint a sokat utazó szakembereknél általában, a laptop számomra is egy extra tartozék lett, ami utazás közben ideális eszköz a videó lejátszására, főleg mostanában, amióta sok repülőgépen



■ 1. ábra Lejátszás közbeni fésűsödés

megjelentek a hordozható elektronikus eszközök energiaellátásához szükséges csatlakozók.

A *MythTV* videó exportálásakor alapvető kérdés, hogy megtaláljuk az egyensúlyt a méret és a feldolgozási idő között. A digitális TV adás szabványai leírják, hogyan továbbítsunk *MPEG-2* videó adatfolyamot egy TV csatornára, így a *MythTV* a digitális tévéműsorokat „természetes” *MPEG-2* formátumukban tárolja. Az *MPEG-2* digitális TV adatfolyam konvertálása egy másik *MPEG-2* alapú videóformátumra viszonylag könnyű, és nem igényel hosszú feldolgozási időt. A digitális tévéadás *MPEG-4* formátumra konvertálása már sokkal nagyobb számítási teljesítményt követel, de az eredményként kapott videó fájl jóval kisebb lesz.

A *MythTV* felvételek feldolgozása

Bár a digitális TV felvételek *MPEG-2* videó adatfolyamok, a *MythTV* által használt *NuppelVideo* konténer formátum speciális *MythTV* sajátosság, és a legtöbb videólejátszó szoftver nem ismeri. Amennyiben a videókat nemcsak a *MythTV* kezelőfelületén keresztül szeretnénk megtekinteni, olyan formátumra kell őket konvertálni, amit több lejátszó is támogat.

A videó exportálását tovább bonyolítja a *MythTV* felvételek tárolásánál használt fájlnevei. A fájlnev első része

a *MythTV* által használt csatornaszám, a két hosszú szám pedig a kezdés illetve a befejezés időpontja:

```
myth:/video$ ls -lh *.nuv
-rw-r--r-- 1 mythtv users 2.8G
➔ Sep 3 08:00 1000_2005090307
➔ 3000_20050903080000.nuv
-rw-r--r-- 1 mythtv users 9.1G
➔ Aug 31 23:30 1007_200508312
➔ 20000_20050831233000.nuv
-rw-r--r-- 1 mythtv users 808M
➔ Sep 3 01:30 1008_200509030
➔ 10000_20050903013000.nuv
-rw-r--r-- 1 mythtv users 1.8G
➔ Sep 1 09:00 1009_200509010
➔ 80000_20050901090000.nuv
-rw-r--r-- 1 mythtv users 3.7G
➔ Aug 28 22:00 1019_200508282
➔ 10000_20050828220000.nuv
```

A *MythTV* felvételek szélesebb körben támogatott formátumra konvertálásához szükséges eszközök készen elérhetők. A *MythTV* disztribúció része a *mythtranscode*, egy program, ami képes a *MythTV NuppelVideo* fájlokat más alkalmazások által is támogatott szabványos videó adatfolyammá alakítani. A *mythtranscode* kimenetét tetszőleges kódoló program bemenetére irányítva bármilyen típusú videó fájl létrehozhatunk. A gyakran használt kódoló programok egyike az *FFmpeg*, ami számos elterjedt videóformátumot támogat. A két program összekapcsolása a megfelelő

kimeneti fájl létrehozásához elméletileg lehetséges, de az összes parancssori opción, és a rendszerben tárolt felvételeken átvergődni nem egyszerű feladat.

Használjuk inkább a *nuvexport* nevű *Perl* szkriptet, ami igazán felhasználóbarát módon kezeli a folyamatot. A *nuvexport* összegyűjti azokat a parancssori opciókat, amelyek a *mythtranscode* és az *FFmpeg* futtatásához szükségesek. Egy szövegalapú kezelőfelületen keresztül kiválaszthatjuk az exportálni kívánt műsort, majd beállíthatjuk a konverziós programok számára szükséges paramétereit.

A *nuvexport* telepítése előtt szükségünk lesz néhány további eszközre. Az *FFmpeg* az alapértelmezett program, amivel a videó a választott célformátumra kódolható. (A *nuvexport* támogatja az átkódolást is, de az sokkal lassabb.) A *MythTV* fájlokat az *MPlayerrel* dekódoljuk a konverzióhoz. A konverzióval gyakran együtt jár a zaj megjelenése is, amit ki kell szűrniük. A *nuvexport* a *yvdenoise* programot használja, ami az *MJPEG* eszközök egyike. Mindhárom programot széles körben használják, és valószínű, hogy az általunk használt *Linux* disztribúcióhoz is elérhetők az őket tartalmazó csomagok. A *nuvexport* a *DateManip* modult is használja, úgyhogy szerezzük be ezt is a disztribúciónk weboldaláról, vagy a kedvenc *CPAN* tüköroldalunkról.

A *nuvexport* elindításakor első dolgunk az exportált videófájl formátumának kiválasztása. Az alapvető kérdés itt az, hogy minimális fájl mérettel rendelkező, ám hosszabb feldolgozási időt igénylő *MPEG-4* alapú formátumot válasszunk, vagy használjunk nagyobb méretű, de könnyebben létrehozható formátumot. A főbb lehetőségek az alábbiak:

- **Video CD (VCD):** *MPEG-1* videó adatfolyamból áll (1.150kbps), egy különálló *MPEG layer 2 (MP2)* formátumú hangsávval (224 kbps).
- **Super Video (SVCD):** változó bitsűrűségű *MPEG-2* videó, *MP2* audiosáv mellett. A *VCD*-vel ellentétben a hang sáv többszoros lehet, így 5.1-es vagy 7.1-es hang is tárolható ebben a formátumban.

- **DVD:** az **MPEG-2** videón alapul, a hangsávra többféle opcionális lehetőség van. Magasabb felbontású bármely **Video CD** formátumnál.
- **DivX:** **MPEG-4** alapú formátum, ami kis fájlokat eredményez; a fájl méretet a minőség jelentős romlása nélkül minimalizálja. A **DivX** állandó illetve változó bitsűrűségű videó létrehozására is képes.
- **XviD:** **MPEG-4** alapú formátum, a **DivX** fejlesztésének egyik oldal-hajtása. Egy 2001-ben kiadott nyílt forráskódú **DivX** kodek fejlesztésén alapul. Alapértelmezésben a **nuvexport** változó bitsűrűségű videó kódolást használ az **XviD**-nél, és felkínálja az egy- illetve többmenetes kódolás lehetőségét. A többmenetes kódolás jobb minőségű videót biztosít, a számítási idő növekedése árán.
- **Advanced Streaming Format (ASF):** a **Microsoft** által kifejlesztett általános médiaformátum, rendszerint a **Windows Media Audio (WMA)** és **Windows Media Video (WMV)** fájloknál használják. A **WMV** a **Microsoft MPEG-4** bővítményein alapul.

Tapasztalataim szerint a **VCD** és az **SVCD** kodekek jó minőséget nyújtanak gyors feldolgozási idővel, míg a **DivX** és az **XviD** kodekek adják a legkisebb fájl méretet, de létrehozásuk hosszabb ideig tart. Miután kiválasztottuk a videóformátumot, a **nuvexport** szövegalapú menürendszerét használva kiválaszthatjuk az exportálni kívánt filmrészleteket, és beállíthatjuk a kodek paramétereit. Miután kijelöltük az exportálásra váró részleteket, a **nuvexport** általános kérdéseket tesz fel, mint például hogy hová kerüljön az exportált fájl. Felajánlja a **MythTV** egyik szolgáltatását, amivel az exportált videóból kivághatjuk a reklámokat. A zajcsökkentés és a deinterlacing opcionálisak. Bár alapértelmezés szerint mindkét opció be van kapcsolva, általában ki-kapcsolom őket a megnövelt számítási idő miatt. Sok videólejátszó képes a deinterlacing elvégzésére lejátszás közben, és nem találtam a konverzióból származó rendkívüli zajt.

Néhány formátumnál további kodekspecifikus kérdések is felmerülnek. Mind a **DivX**, mind az **XviD** lehetővé teszi a bitsűrűség és a felbontás állítását. Az alapértelmezett bitsűrűség 128 kbps a hang illetve 960 kbps a videó esetében, ami elegendő arra, hogy a legtöbb számítógépes megjelenítőn jó minőségű videót eredményezzen. A videó méretének beállításakor a **nuvexport** először megkérdezi a kívánt szélességet, majd javasol egy magasságot, ami a felvétel képméretarányán alapul. Figyeljünk arra, hogy a szélesség a felvétel szélességén belül maradjon. Az alapértelmezett szélesség 624, ami általában jó videót eredményez, de 1920x1080 méretű felvételeknél nagyobb is lehet. A **VCD** és az **SVCD** nem kérdezik rá a felbontásra, mert ezek a formátumok fix felbontásúak. Az 1. táblázat a különböző formátumokra jellemző feldolgozási időket és tárhelyszükségletet hasonlítja össze, köztük a **MythTV** saját formátumát és az én kereskedelmi **DVR**-emet is. Forrásként egy órányi hosszúságú felvételt használtam a **PBS Nova** sorozatából, amit 704x480 felbontásban sugároznak. A „kereskedelmi **DVR**” bejegyzés a **DVR**-ről a laptopomra másolt videóra utal. Hozzávetőleges számítás szerint az **MPEG-2** körülbelül 1 GB-ot igényel óránként, míg az **MPEG-4** csak 350-450 MB-ot. A táblázatban feltüntetett feldolgozási időbe nem értendő bele a deinterlacing illetve a zajcsökkentés; mindkét feladatot a videólejátszóval oldom meg, lejátszás közben.

Bár az **ASF** mérete a legkisebb, ez is néz ki messze a legrosszabbul. Az **ASF** fájlban gyakoriak a tömörítésből adódó hibák, ami nagyon idegesítővé teszi a nézését. Bár a kis fájl méret vonzó, a gyenge képminőség miatt kizárhatjuk ezt a formátumot. A **DVD** videó rendelkezik a legjobb felbontással, de ez foglalja a legtöbb helyet is. A két véglet közötti kompromisszumként a **VCD**-t és a **DivX**-et használom; egy fájl gyors létrehozására az előbbi alkalmas, az utóbbival pedig a lehető legkisebb fájl méret érhető el. A videó átkódolása erősen igénybe veszi a **CPU**-t. Alapértelmezés szerint a **nuvexport** alacsony prioritási értékkel futtatja a segéd folyamatokat, hogy megelőzze a rendszeren futó egyéb műveletekkel (például videólejátszás vagy felvétel) való ütközéseket. Az utóbbi időben kiadott összes **Linux** disztribúcióban található olyan szoftver, ami lehetővé teszi a **CPU** órajeletnek megváltoztatását, eleget téve a feldolgozási teljesítményt érintő igényeknek. A **CPU** sebességvezérlőt használva az órajelet a lehető legalacsonyabb értéken tartom, ami még elegendő a szükséges munka elvégzéséhez. Sok sebességvezérlő program nem veszi figyelembe a csökkentett prioritású folyamatokat, de beállíthatók az ilyen folyamatok figyelésére. Az én **Linux** disztribúcióm a **CPUfreq** kernelmodult használja, amit be kell konfigurálni a csökkentett prioritású folyamatok felügyeléséhez. Egy kisméretű indító szkript az alábbi két parancsot futtatja:

```
echo "ondemand" >
  /sys/devices/system/cpu/
```

1. táblázat **Az exportálás eredménye (jellemző értékek)**

Formátum	Felbontás	Fájl méret	Kódolási idő
Myth formátum	704x480	1,756MB	N/A
Kereskedelmi DVR	N/A	1,236MB	N/A
VCD	352x240	596MB	15 perc
SVCD	480x480	601MB	25 perc
DVD	720x480	899MB	34 perc
DivX	624x464	432MB	35 perc
XviD (egymenetes)	624x464	451MB	1 óra, 39 perc
XviD (kétmenetes)	624x464	472MB	2 óra, 35 perc
ASF	320x240	143MB	18 perc


```

→ cpu0/cpufreq/scaling_governor
echo "1" > /sys/devices/
→ system/cpu/cpu0/cpufreq/
→ ondemand/ignore_nice

```

A második parancs arra utasítja a **CPUFreq** vezérlőt, hogy számolja a csökkentett prioritású folyamatoktól érkező feldolgozási kérelmeket. Az 1. Táblázatban felsorolt idők egy csúcsteljesítményen működő **2.0 GHz-es AMD Athlon64**-ra érvényesek. Az 1.0 GHz-es minimum sebességen a feldolgozási idő ennek körülbelül három-négyszerese. A **MythTV** videó exportálása azonban csak az egyik fele a dolgoknak. Amint kész a videó, a lejátszáshoz átvihető egy másik gépre. A visszajátszásra használt platformon jelenlevő alkalmazásokon kívül létezik két figyelemre méltó nyílt forráskódú lejátszó: az **MPlayer** és a **Video LAN Client (VLC)**. Én az **MPlayer**t használom, mert a beépített deinterlacing képességek simább képet eredményeznek, mint a **VLC**. Mindkét alkalmazás elérhető **Linux** és **Windows** alatt is. Az **MPlayer** parancsora a különböző operációs rendszereken megegyezik.

A cél az éles, teljes képernyős lejátszás elérése. Az **-fs** opció a videót teljes képernyőn játssza le, nincs ablak a kép körül.

A videósűrítők a lejátszás megváltoztatására használhatók, és a **-vf** paraméterrel aktiválhatók. Két **-vf** opciót használok. Az egyik egy kis fekete keretet hoz létre a képernyő körül az **expand** szűrővel. Az **expand** szűrő több paramétert is kaphat, egy negatív számmal határozzuk meg a keretet. A **-vf expand=0:-50** szűrő utasítás egy 50 pixeles szegélyt helyez a képernyő aljára, miközben a videó a kereten belül középre igazítva marad. Az éles videóhoz szükséges a **deinterlacing** is: ezt az **MPlayer** az utófeldolgozó szűrővel (**postprocessing filter**, röviden **pp**) aktiválja. Általános szabály szerint négy utófeldolgozó szűrőt kapcsolok be: a vízszintes blokktorlódás (**hb**), a függőleges blokktorlódást (**vb**), a **de-ringing-et (dr)**, és a fényerő/kontraszt korrekciót (**a1**). Az eredményként kapott szűrő a **-vf pp=hb/vb/dr/a1** paranccsal aktiválható. Az egészet összerakva az alábbi parancssort kapjuk:

```

mplayer -fs -vf pp=hb/vb/dr/
→ a1 -vf expand=0:-50
→ (filename)

```

Lejátszás alatt az **MPlayer** kiterjedt billentyűparancsai lehetővé teszik a pillanatnyi szünetet, a gyors előretekerést és a képbéállítást, csakúgy, mint a kivetített megjelenítést.

Linux Journal 2006., 140. szám



Matthew Gast

a vezeték nélküli hálózatról szóló kiváló szakönyv (802.11 Wireless Networks: The Definitive Guide

(O'Reilly Media) szerzője. Jelenleg nagyon sok időt tölt a MythTV-vel. Elérhető a matthew.gast@gmail.com email címen, de csak ha éppen a tengerszint közelében tartózkodik.

FORRÁSOK

→ www.linuxjournal.com/article/8585

A jól informáltak klikkje!

PRIM

ONLINE

www.prim.hu