Video Encoding with Open Source Tools

Steffen Bauer, 26/1/2010 LUG Linux User Group Frankfurt am Main

Overview

Basic concepts of video compression

Video formats and codecs

How to do it with Open Source and Linux

1. Basic concepts of video compression

Characteristics of video streams

Framerate

Number of still pictures per unit of time of video; up to 120 frames/s for professional equipment. PAL video specifies 25 frames/s.

Interlacing / Progressive Video

Interlaced: Lines of one frame are drawn alternatively in two half-frames
Progressive: All lines of one frame are drawn in sequence

Resolution

Size of a video image (measured in pixels for digital video)
768/720×576 for PAL resolution
Up to 1920×1080p for HDTV resolution

Aspect Ratio

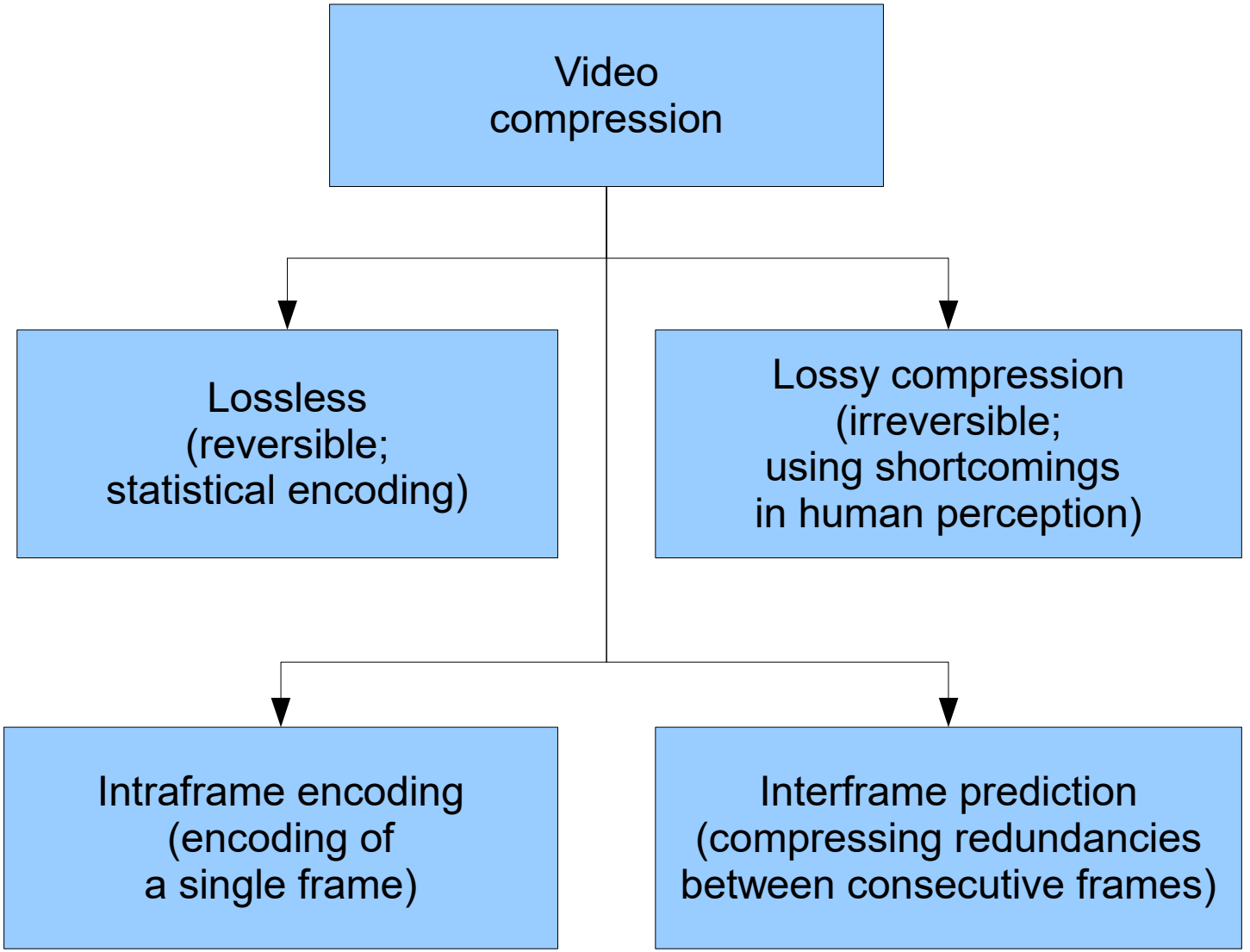
Dimensions of the video screen; ratio between width and height.
Pixels used in digital video can have non-square aspect ratios!
Usual ratios are 4:3 (traditional TV) and 16:9 (anamorphic widescreen)

Why video encoding?

Example: 52 seconds of DVD PAL movie
(16:9, 720x576p, 25 fps, progressive scan)

Video codec	Raw Size	Compression factor	Comment
Raw frames	1.1 GB	-	1300 single frames, MotionTarga, uncompressed
HUFFYUV	459 MB	2.2 / 55%	Lossless compression
MJPEG	60 MB	20 / 95%	Motion JPEG; lossy; intraframe only
lavc MPEG-2	24 MB	50 / 98%	Standard DVD quality
X.264 MPEG-4 AVC	5.3 MB	200 / 99.5%	High efficient video codec

Basic principles of multimedia encoding



YUV color space

RGB Colorspace

Red

Green

Blue

YUV Colorspace

Luminance

Blueness

Redness

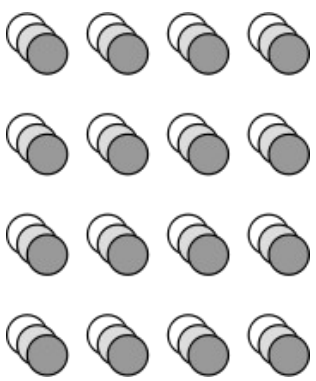
Source: Wikimedia Commons

YUV chroma subsampling

Reduce signal rate by subsampling the chroma channels

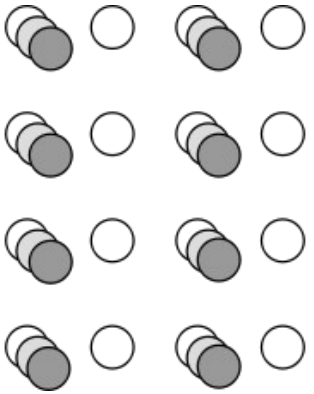
Ratio of subsampling usually expressed in J:a:b format

J: Horizontal sampling reference
a: Number of chrominance samples in first row
b: Number of chrominance samples in second row



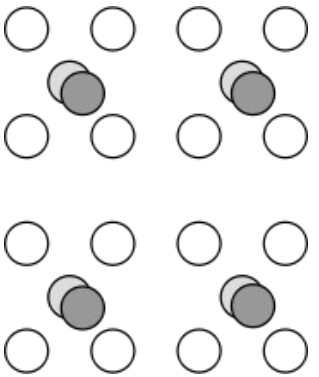
YUV 4:4:4

(lossless compression)



YUV 4:2:2

(used in high-end digital video)



YUV 4:2:0

(all MPEG codecs, JPEG, TV, ...)

Source: Wikimedia Commons

Frequency transformation and Quantization

Most important technique of lossy intraframe compression

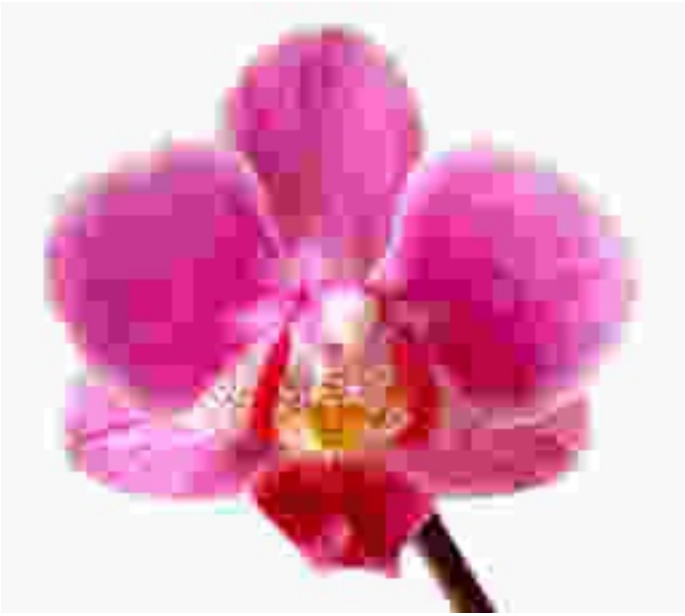
Parting the picture into *Macroblocks* (usually 8x8)

Transforming into *frequency space* (mostly DCT, also Wavelets etc.)

Reducing local detail by *Quantization*

100%
60 kB

25%
5 kB



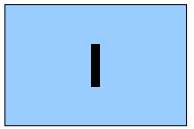
5%
1,5 kB

Source: Wikimedia Commons

Interframe encoding: Types of frames

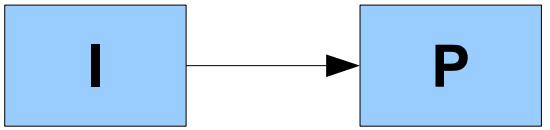
I-Frame (Intra Frame)

Least compressed frame type, intraframe only
Also called *keyframe*
Needed for regular stream refresh (*keyframe interval*)
Obligatory at scene cuts



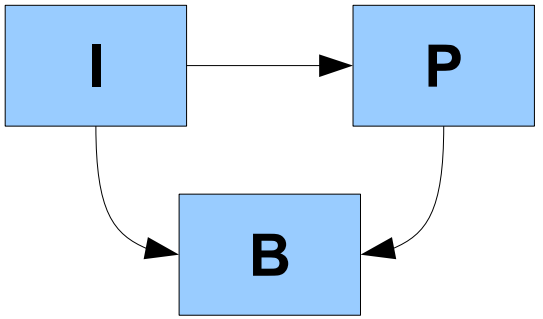
P-Frame (Predicted Frame)

Higher compressed frame type
Stores only difference to previous frames
High influence of codec effectiveness on compression (motion vector estimation etc.)



B-Frame (Bi-predicted Frame)

Highest compressed frame type
Calculated from previous and successive frames
Does not degrade video stream quality



Video encoding modes

1-Pass, - bitrate

Quality: Worst quality/bitrate ratio
Encoding time: Fast
Video size: Variable within certain boundaries
Recommended: Generally not, only when bitrate is fixed (videoconferencing etc.)

1-Pass, - quantizer

Quality: Anything from very low to very high (depending on quantizer)
Encoding time: Usually fast
Video size: Variable; cannot be predicted easily
Recommended: When exact file size is of minor importance

2-Pass, - bitrate

Quality: Best quality/bitrate ratio
Encoding time: Slow (two passes needed)
Video size: Fixed within tight boundary (usually)
Recommended: For given target file size (storage on physical media)

2. Video formats (and more...)

(Choosing the right format for your video can be very difficult ... there are so many to choose from!)

aac
aasc
ac3
adpcm_4xm
adpcm_adx
adpcm_ct
adpcm_ea
adpcm_ea_maxis_xa
adpcm_ea_r1
adpcm_ea_r2
adpcm_ea_r3
adpcm_ea_xas
adpcm_g726
adpcm_ima_amv
adpcm_ima_dk3
adpcm_ima_dk4
adpcm_ima_ea_eacs
adpcm_ima_ea_sead
adpcm_ima_iss
adpcm_ima_qt
adpcm_ima_smjpeg
adpcm_ima_wav
adpcm_ima_ws
adpcm_ms
adpcm_sbpro_2
adpcm_sbpro_3
adpcm_sbpro_4
adpcm_swf
adpcm_thp
adpcm_xa
adpcm_yamaha
alac
als
amv
ape
asv1
asv2
atrac1
eightsvx_exp
eightsvx_fib
escape124
ffv1
ffvhuff
flac
flashsv
flic
flv
fourxm
fraps
frwu
gif
h261
h263
h263i
h264
huffyuv
idcin
imc
indeo2
indeo3
interplay_dpcm
interplay_video
jpegls
kmvc
loco
mace3
mace6
mdec
mimic
mjpeg
mjpegb
mlp
mmvideo
motionpixels
mp1
mp2
pcm_s16le
pcm_s16le_planar
pcm_s24be
pcm_s24daud
pcm_s24le
pcm_s32be
pcm_s32le
pcm_s8
pcm_u16be
pcm_u16le
pcm_u24be
pcm_u24le
pcm_u32be
pcm_u32le
pcm_u8
pcm_zork
pcx
pgm
pgmyuv
pgssub
png
ppm
ptx
qcelp
qdm2
qdraw
qpeg
qtrle
ra_144
ra_288
rawvideo
rl2
roq
roq_dpcm
rpza
rv10
rv20
rv30
atrac3
avs
bethsoftvid
bfi
bmp
c93
cavs
cinepak
cljr
cook
cscd
cyuv
dca
dnxhd
dpx
dsicinaudio
dsicinvideo
dvbsub
dvdsub
dvvideo
dxa
eac3
eacmv
eamad
eatgq
eatgv
eatqi
eightbps
v210
v210x
vb
vc1
vcr1
vmdaudio
vmdvideo
vmnc
vorbis
vp3
mp3
mp3adu
mp3on4
mpc7
mpc8
mpeg1video
mpeg2video
mpeg4
mpegvideo
msmpeg4v1
msmpeg4v2
msmpeg4v3
msrle
msvideo1
mszh
nellymoser
nuv
pam
pbm
pcm_alaw
pcm_bluray
pcm_dvd
pcm_f32be
pcm_f32le
pcm_f64be
pcm_f64le
pcm_mulaw
pcm_s16be
vp5
vp6
vp6a
vp6f
vqa
wavpack
wmapro
wmav1
wmav2
wmv1
rv40
sgi
shorten
smackaud
smacker
smc
snow
sol_dpcm
sonic
sp5x
sunrast
svq1
svq3
targa
theora
thp
tiertexseqvideo
tiff
tmv
truehd
truemotion1
truemotion2
truespeech
tscc
tta
twinvq
txd
ulti
wmv2
wmv3
wnv1
ws_snd1
xan_dpcm
xan_wc3
xl
xsub
zlib
zmbv

Common misconceptions: Containers, Formats, Codecs

Confusing „Container“ and „Format“

A **container** stores the different streams (video, audio, subtitles) of a video file, together with metadata

A **format standard** defines how a media stream looks like

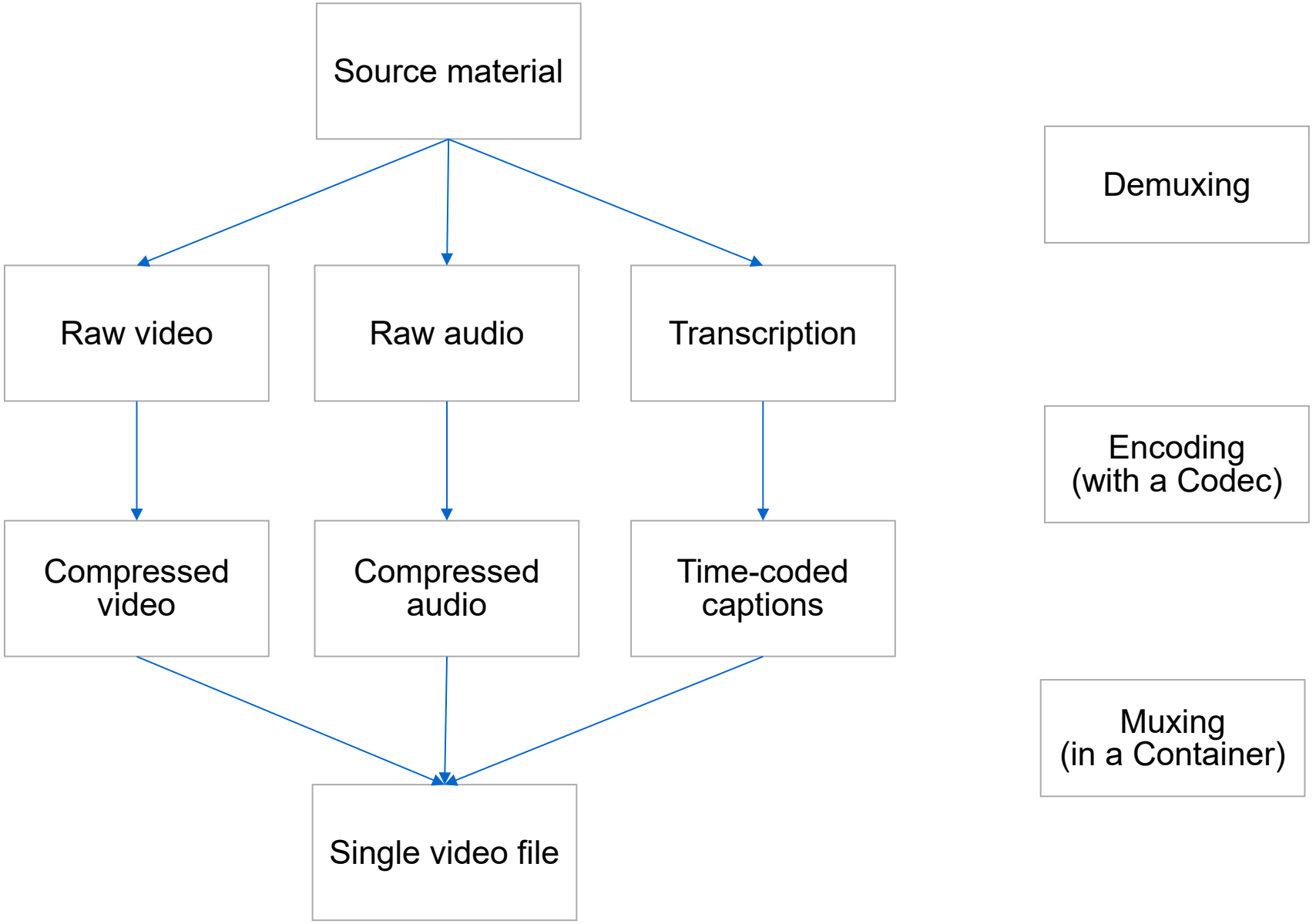
Confusing „Format“ and „Codec“

A **codec** (x264, Xvid etc.) is encoding into a format standard (MPEG-2, H.264 etc.)

A **format standard** is only an abstract definition how an encoded stream has to look like; different codecs can encode with different efficiency into the same format!

Example: „DivX“ is not a video format. It is the name of a corporation, which provided a codec of the same name for encoding into the MPEG-4 ASP format

Video encoding process



Source: Marc Pilgrim „A Gentle Introduction to Video Encoding“

Constraints on choosing a format/codec

CPU capacity for decoding and playing in real time

Codec and Container compatibility

CPU capacity for encoding

Acceptable delay between recording and delivery

Acceptable quality degradation

Screen dimensions

Bandwidth

Hard limits on storage size (for physical media)

Patent licensing costs

Source: Marc Pilgrim „A Gentle Introduction to Video Encoding“

Common container formats

Format name	Ext	Open Source Tools	Comment
Audio Video Interlaced	.AVI	libavformat	„The outdated format which will just not die“
Advanced System Format	.ASF	libavformat	Windows video (WMV)
Flash Video	.FLV	libavformat	Made popular by YouTube; embedded web video
Matroska	.MKV	mkvtoolnix	Very flexible container format, free format; gained some popularity in recent years
Ogg Video	.OGV	libogg	Open standard not encumbered by any known patents; standard container for theora/vorbis.
MPEG-4	.MP4 .M4V	MP4Box	Container defined by MPEG-4 standard, based on .mov format
Quicktime	.MOV	libavformat	Proprietary, developed by Apple
Video Object	.VOB	DVDauthor	DVD stream container

Sources: Marc Pilgrim „A Gentle Introduction to Video Encoding“, Wikipedia

Common video formats (with relevance for OSS)

Format name	Since	Open Source Codecs&Tools	Popular usage	Patent-free?
MJPEG	?	mjpegtools, libavcodec	(older) Digicams, Webcams	Unclear*
MPEG-1	1991	mjpegtools, libavcodec	VCD	No
MPEG-2	1994	mjpegtools, libavcodec	DVD, digital TV	No
MPEG-4 Part 2 ASP	1999	Xvid	Movie enthusiasts („DivX“)	No
MPEG-4 Part 10 AVC H.264	2003	x264	YouTube HD, iTunes, Blu-Ray, DVD backups, future digital TV...	No
Theora	2002	libtheora, ffmpeg2theora	Free software enthusiasts, Wikipedia	Yes*
Dirac	2004	dirac-research, schroedinger	Possible alternative to H.264	Yes*

(*) As far as I know

Sources: Marc Pilgrim „A Gentle Introduction to Video Encoding“, Wikipedia

Common audio formats (with more or less relevance to OSS)

Format name	Since	Open Source Codecs&Tools	Popular usage	Patent-free?
PCM	Stone-age	Sox	.wav files, Audio CD, some DVDs (high quality concert DVDs)	Yes*
MPEG-1 Audio Layer 2 MP2	1991	TwoLAME	VCD, DVD, digital TV/Radio	No
MPEG-1 Audio Layer 3 MP3	1991	LAME	Music	No
MPEG-4 Part 3 AAC	1999	FAAC	iPod, iTunes, YouTube (competitor to MP3)	No
AC3	1992	liba52	DVD, Blu-ray („Dolby Digital“)	No
Vorbis	2000	libvorbis	Free software enthusiasts; Wikipedia	Yes*

(*) As far as I know

Sources: Marc Pilgrim „A Gentle Introduction to Video Encoding“, Wikipedia

Common subtitle/captions formats

Caption format	Encoding	Popular usage
OGG Writ	UTF-8	Subtitles for OGG; now unmaintained
Kate	multiple	Subtitles for OGG files
SubRip (.SRT)	windows-1252	AVI files, “fansubbing”
Advanced SubStation Alpha	UTF-8	AVI files, “fansubbing”
Synchronized Accessible Media Interchange	(multiple)	Windows Media Player
Synchronized Multimedia Integration Language	(multiple)	QuickTime (SMIL 1.0 only) / W3C recommended
DVD	Stored as images	DVD
MPEG-4 Timed Text	UTF-8	MP4 files, iTunes Store

Sources: Marc Pilgrim „A Gentle Introduction to Video Encoding“, Wikipedia

Common combinations

Container	Video	Audio	Captions	Usage/Audience
VOB	MPEG-2	MP2 AC3	VOBsub	DVD Video discs
AVI MKV	H.264	MP3 AC3	SRT	DVD/BluRay backup
FLV MP4	H.264	AAC	Google sub (modified SRT)	YouTube
OGV	Theora	Vorbis	Kate	No software patent restrictions / Linux
MP4	H.264	AAC	MPEG-4 Timed Text	Iphone, iPod
MOV	H.264 Sorenson	AAC	QTTextTrack	Mac
EVO	H.264	AC3	SUP BluRay Subtitles	Blu-Ray Discs

Sources: Marc Pilgrim „A Gentle Introduction to Video Encoding“, Wikipedia

The future ... ?

HVC (High performance Video Coding)

- Currently developed as successor to H.264/MPEG-4 AVC (H.264+)
- Aims at further substantial compression improve, *especially at high resolutions*
- Call for Proposals to be submitted by Feb. 2010

MVC (Multiview Video Coding)

- Amendment to H.264/MPEG-4 AVC video compression standard; forward compatible.
- Intended for encoding stereoscopic (two-view) video, as well as free viewpoint television and multi-view 3D television.
- Stereo High profile has been standardized in June 2009.

Source: Wikipedia

3. Video Open Source tools and projects

xiph.org

*„The **Xiph.Org Foundation** is a non-profit corporation dedicated to protecting the foundations of Internet multimedia from control by private interests. Our purpose is to support and develop free, open protocols and software to serve the public, developer and business markets.“*

Main projects:

OGG media container format
Theora video codec
Vorbis audio codec
FLAC lossless audio codec
Speex speech audio codec
Icecast Streaming Server
cdparanoia physical media ripper
... and a lot more

In 2008 the FSF listed the Xiph.Org projects as *High Priority Software Projects. („There is a vital need to draw the free software community's attention to the ongoing development work on these particular projects“)*

FFmpeg is a free software / open source project for handling multimedia data, released under LGPL V2.1 and GPL V2 (certain optional parts and optimizations).

ffmpeg	Command line tool for video conversion
ffserver	Multimedia streaming server
ffplay	Media player
libavcodec	Library containing all FFmpeg codecs (audio&video)
libavformat	Library containing muxers/demuxers for container formats
libavfilter	Video modification
libavdevice	I/O devices for grabbing/rendering multimedia

libavcodec and/or **libavformat** are integral parts of numerous Open Source video tools including ffmpeg2theora, VLC, MPlayer, Handbrake, AviDemux, Blender, Xine, transcode, etc...

VideoLAN is a project started at the Ecole Centrale Paris to produce free and open source software for video and multimedia purpose, released under the GNU General Public License. Their most important projects:

VLC media player a highly portable multimedia player supporting most audio and video formats from files, physical media (DVDs, VCD, Audio-CD), TV capture cards and many network streaming protocols.

x264 a free library for encoding H264/AVC video streams, *one of the best codecs currently available.*

libdvdcss for decoding CSS-encrypted DVDs (Note: Legality unclear!)

ffmpeg2theora

ffmpeg2theora is currently the most functional Theora encoder; based on FFmpeg and libtheora/libvorbis/libkate

It is designed specifically to create multimedia files using:

Theora codec for video content
Vorbis codec for audio content
Kate for captions/subtitles
OGV (OGG Video) as container

Open Source Encoding Tools

Libraries

- libavcodec
- libavformat
- libtheora
- x264
- xvid
- libvorbis
- libkate
- libogg

Coding/Decoding

- FFmpeg
- mencoder
- transcode
- mjpegtools
- ffmpeg2theora
- Avidemux
- Handbrake

Support Tools

- mkvtoolnix
- MP4Box
- DVDauthor
- DVDstyler

... and a lot more

Encoding with ffmpeg2theora

> ffmpeg2theora *inputvideo* -o *outputvideo.ogv*

--videoquality *[0-10]* --audioquality *[0-10]*
--videobitrate <*vbitrate*> --audiobitrate <*abitrate*>
--two-pass --soft-target

--width <*W*> --height <*H*>
--aspect *ratio* --framerate <*N*> --keyint <*N*>
--subtitles <*subtitles.srt*>
--title "*Title*" --artist "*artist*" --date "*date*"
--croptop <*N*> --cropbottom <*N*> --cropleft <*N*> --cropright <*N*>
--contrast <*N*> --brightness <*N*> --saturation <*N*> –gamma <*N*>
--sync --deinterlace

Source: ffmpeg2theora manpage

Encoding with mencoder

> mencoder *inputvideo*
> mencoder *mf://[filemask|@listfile] [-mf options]*
> mencoder *dvd://[title]*

-ovc <videocodec> -oac <audiocodec> -of <format> -o *outputvideo*
-<codec/format>opts <option1[=value1]:option2[=value2]:...>

-vf <filter1[=parameter1:parameter2:...],filter2,...>
-sub <subtitlefile>

-ovc copy
- raw
- lavc
- xvid
- x264

-oac copy
- pcm
- lavc
- mp3lame

Source: mencoder manpage

Example: Creating DVD-compliant video with mencoder

> mencoder sourcevideo.avi	Source file
-oac lavc -ovc lavc	Use libavcodec for audio&video
-of mpeg	Output format mpeg
-mpegopts format=dvd:tsaf	DVD video w/timestamps
-vf scale=720:576,harddup	Scale to DVD w/h, no frameskip
-srate 48000	Audio rate 48 kHz
-af lavcresample=48000	
-lavcopts	
vcodec=mpeg2video:	Video format MPEG-2
vrc_buf_size=1835:	Bitrates as of DVD standard
vrc_maxrate=9800:	
vbitrate=5000:	
keyint=15:vstrict=0:	for standalone player compatibility
acodec=ac3:abitrate=192:	Audio format AC3
aspect=16/9	Video aspect „Widescreen“
-ofps 25	Framerate 25 fps
-o dvdvideo.mpg	Output file

Source: mencoder documentation

Example: High quality H.264 two-pass encoding with mencoder

Two-Pass encoding of all .png files in the current directory:

1. pass, dumping the raw video (store only the bitrate characteristic)

```
> mencoder mf://*.png -mf fps=25
        -ovc x264 -x264encopts
        bitrate=2000:threads=auto:pass=1:turbo=2:
      <more x264 encoding options>
        -nosound -ofps 25 -noskip
        -of rawvideo -o /dev/null
```

2. pass, storing the encoded video stream in .264

```
> mencoder mf://*.png -mf fps=25
        -ovc x264 -x264encopts
        bitrate=2000:threads=auto:pass=2:turbo=0:
      <more x264 encoding options>
        -nosound -ofps 25 -noskip
        -of rawvideo -o output.264
```

Muxing the raw video into .mp4 container:

```
> MP4Box -fps 25 -add output.264 output.mp4
```

Source: mencoder manpage

Useful links

Tutorials, forums etc.

http://diveintomark.org/tag/give	'A Gentle Introduction to Video Encoding'
http://diveintohtml5.org/video.html	Video on the Web (Good intro to web video)
http://forum.doom9.org/	Doom 9 forum (Ask any question here!)
http://www.videohelp.com/	Videohelp (Lots of guides)
http://encodingwissen.de/	Good tutorial (in german)
http://en.flossmanuals.net/theoracookbook	The Theora Cookbook

Open Source projects

http://www.mplayerhq.hu/	Mencoder / Mplayer
http://ffmpeg.org/	FFmpeg
http://v2v.cc/~j/ffmpeg2theora/	FFmpeg2theora
http://www.xiph.org/	Xiph.org foundation (theora, vorbis, ogg etc.)
http://www.videolan.org/	VideoLAN (x264 and other great video projects)
http://www.xvid.org/	Xvid codec
http://www.matroska.org/	Matroska container

Thank you for your attention!

Enjoy your video!

... any Questions or comments?