Useful scripts and command line options

To convert an image sequence to an mp4:

convert2mpeg4

Usage: convert2mpeg4 dirname [scaleFactor]

Creates an MPEG-4 (dirname-scaleFactor) from image sequence found in dirname

Where:

dirname is the directory containing the image sequences

scaleFactor is the dimensions of the generated mpeg4. Default is 2048x1024

Example:

convert2mpeg4 /tmp/seq 2048x1024

Makes an mpeg4 from sequence images in /tmp/seq, scaled to 2048x1024

convert2mpeg4 /tmp/seq 1024x512

Makes an mpeg4 from sequence images in /tmp/seq, scaled to 1024x512

To convert one movie format to another:

ffmpeg –v 2 –i old_movie.wav –b 25000 –s size –qscale 1 new_movie.mp4

-v verbose	control amount of logging
-i input	the old movie that you want to convert
-b bitrate	set video bitrate (in kbit/s) (default is 200, but higher bitrates make higher quality movies that play smoothly)
-s size	widthxheight standard for SOS 2048x1024
-qscale	Use fixed video quantizer scale (VBR) The available qscale values range from 1 (highest quality) to 31 (lowest quality)

The very last thing you include is the name for the new video with the proper video extension that you want

To convert file format of a directory of images and/or resize them:

rtResizeDir.sh

Usage: prog size quality destType srcDir dstDir

This can be used to either resize a directory of images and/or to convert the file type.

Example

rtResizeDir.sh 2048x1024 85 png images_jpg images_png

The suggested quality to use is at least 85

Supported file formats include jpeg, png, tiff, gif, and ppm

In this example images_jpg is the directory of original images and images_png is the directory of where the new images will be put. If the new directory doesn't exist, it will automatically be generated

To resize a single image:

convert –resize 2048x1024 original.jpg resized.jpg

where 2048x1024 is the new size that the image will be (should only be used to reduce the size of an image, enlarging pictures reduces the quality).

If you use the same file name for the input and output, it will overwrite the input

To convert file format of a single image:

Convert image.jpg image_new.png:

Where the first file is the original image and the second is what you want to create
The you can use the same name with different extensions

To list the dimensions of an image:

identify image.jpg

The output looks like this:

image.jpg JPEG 8000x4000 DirectClass 8.4mb 0.000u 0:01

If you use identify –verbose image.jpg, you get much more information

To list information about a movie (can be used for audio files as well):

mplayer –identify video.mp4 –frames 0

The output is lengthy, but the helpful part looks like this:

```
VIDEO:  [avc1]  2048x1024  24bpp  15.000 fps    0.0 kbps ( 0.0 kbyte/s)
ID_FILENAME=video.mp4
ID_DEMUXER=mov
ID_VIDEO_FORMAT=avc1
ID_VIDEO_BITRATE=0
ID_VIDEO_WIDTH=2048
ID_VIDEO_HEIGHT=1024
ID_VIDEO_FPS=15.000
ID_VIDEO_ASPECT=0.0000
ID_LENGTH=80.00
```

To list information about an audio file:

sox file.wav –e stat

The output looks like this:

```
Samples read:          17904600
Length (seconds):    203.000000
Scaled by:         2147483647.0
Maximum amplitude:     0.734467
Minimum amplitude:    -0.507111
Midline amplitude:     0.113678
Mean    norm:          0.032595
Mean    amplitude:    -0.001058
RMS     amplitude:     0.052694
Maximum delta:         0.117004
Minimum delta:         0.000000
Mean    delta:         0.001281
RMS     delta:         0.003606
Rough   frequency:          480
Volume adjustment:        1.362
```

If sox doesn't work, the mplayer example above works with audio files as well