Command cheetsheet

ImageMagic

Scale to a dimension:

```
convert i.png -resize 100x100 o.png
```

Extend image to another dimension

```
convert original.png -resize 100x100^ -gravity center
-extent 100x100 new.png
```

Resizing files in place

```
mogrify -resize 100x100 original.png
```

```
convert i.jpg -resize 75% o.jpg
```

Batch convertion to jpg

```
mogrify -format jpg *.png
```

Append images

veritcal: `convert -append in-*.jpg out.jpg`

horizontal: `convert +append in-*.jpg out.jpg`

Subtract A from B

```
convert B.jpg A.jpg -compose minus -composite  o.png
```

Rotate image by a°

```
convert i.jpg -rotate a o.jpg
```

Youtube-dl

Extract audio only

```
youtube-dl -x --audio-format mp3 link
```

Instaloader

Download a post

```
instaloader -- -shcode
```

FFmpeg

Convert to stable mp3

```
ffmpeg -i i.mp3 -vn -ar 44100 -ac 2 -b:a 192k o.mp3
```

Concat multiple audios

```
ffmpeg -f concat -i list.txt -c copy o.mp3
```

list.txt: ‘ file 1.mp3 file 2.mp3 ....

Trim audio

```
ffmpeg -ss start -i i.mp3 -t len e o.mp3
```

-ss : start from (in seconds or std frmt)

-t : length (end = ss+t)

-to : end time (in std frmt)

Fade in/out 'n's of audio:

```
ffmpeg -i i.mp3 -af "afade=t=in/out:st=L-n:d=n" o.mp3
```

L = length of audio

Increase or decrease volume by 'x' times

```
ffmpeg -i i.wav -filter:a "volume=x" o.wav
```

or by ±ndb:

```
ffmpeg -i i.wav -filter:a "volume=ndb" o.wav
```

scale/resize a video/img

resize :`ffmpeg -i in -vf scale=w:h out`

scale : `ffmpeg -i in -vf scale=w:-1 out`

half the size: `ffmpeg -i in -vf scale=w=iw/2:h=ih/2 out`

Rotate video

```
ffmpeg -i i.mp4 -vf "transpose=1" o.mp4
```

where

0 = 90CounterCLockwise and Vertical Flip (default)

1 = 90Clockwise

2 = 90CounterClockwise

3 = 90Clockwise and Vertical Flip

Speed up/down video by 'x' times

```
ffmpeg -i i.mp4 -filter:v "setpts=PTS/x" o.mp4
```

Stabilize video

```
ffmpeg -i i.mp4 -vf vidstabdetect -f null -
ffmpeg -i i.mp4 -vf vidstabtransform o.mp4
```

Export single frame

```
ffmpeg -i i.mp4 -ss ... -frames:v 1 o.jpg
```

Adjust quality of video

```
ffmpeg -i .. -c:v libx265 -crf ... o.mp4
```

Add metadata

```
ffmpeg -i . -metadata k=v -c copy ..
```

MP4: title, author, album, artist, composer, year, track, comment, genre, copyright, description, synopsis, show, episode_id, lyrics

MP3: album, composer, genre, copyright, title, language, artist, album_artist, disc, publisher, track, encoder, lyrics, date, creation_time, performer

Add cover art

```
ffmpeg -i i.mp3 -i cov.png -c copy -map 0 -map 1 o.mp3
```

Pandoc

Markdown to PDF

```
pandoc i.txt --pdf-engine=xelatex -o o.pdf
```

Set margin to output pdf

```
pandoc i.txt -V geometry:margin=1in --pdf-engine=xelatex
-o o.pdf
```

Miscellaneous functions

Compress pdf:

```
pdfcompress i.pdf o.pdf level
```

level:

- screen: 72dpi
- ebook: 150dpi
- prepress: 300dpi

SoX

Create silent audio of 't's

```
sox -n -r 16000 -c 1 o.mp3 trim 0.0 t
```

GhostScript

Combine pdfs

```
gs -dBATCH -dNOPAUSE -q -sDEVICE=pdfwrite -sOutputFile=out.pdf
ins.pdf
```

Multiline scripts

Average of images

```
i=0
for file in img*jpg; do
    echo -n "$file.. "
    if [ $i -eq 0 ]; then
        cp $file avg.jpg
    else
        convert $file avg.jpg -fx "(u+$i*v)/$[$i+1]" avg.jpg
    fi
    i=$[$i+1]
done
```