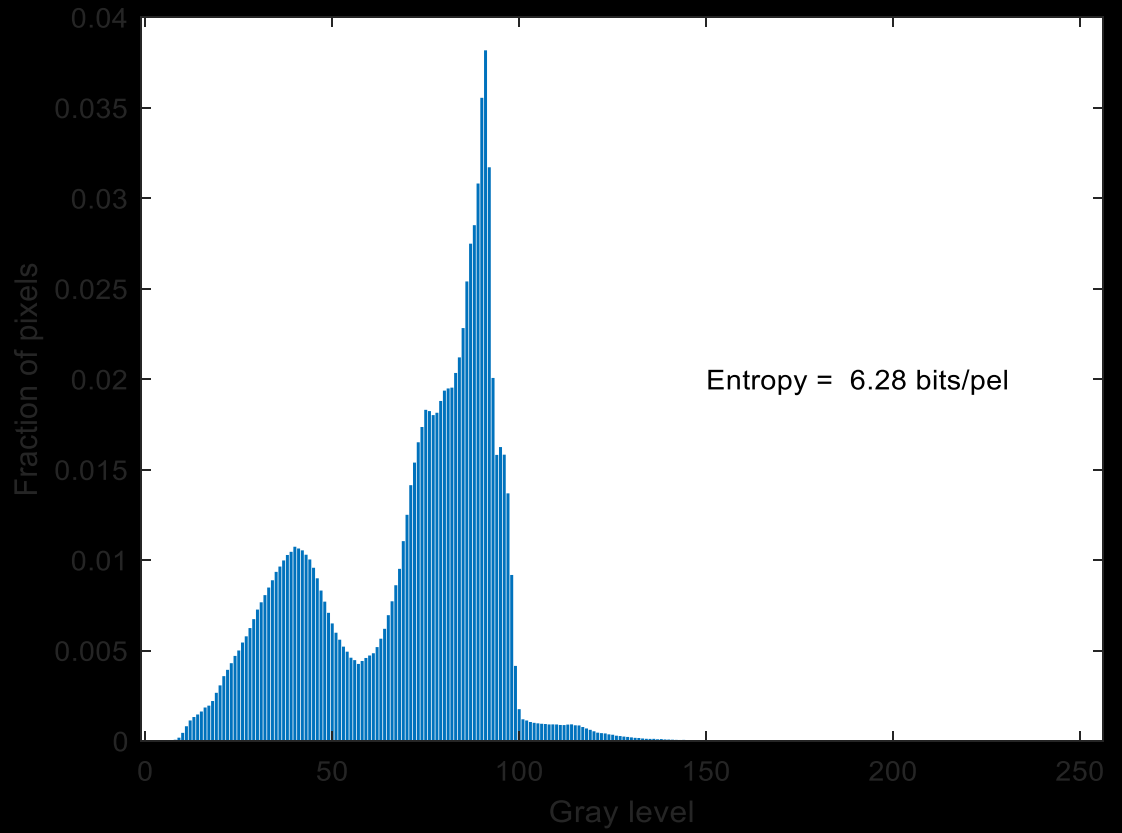
COMPRESSION LAB

due on Tuesday 29th March 2021 at 12 midday

INFORMATION CONTENT IN THE RAW 8BIT IMAGE

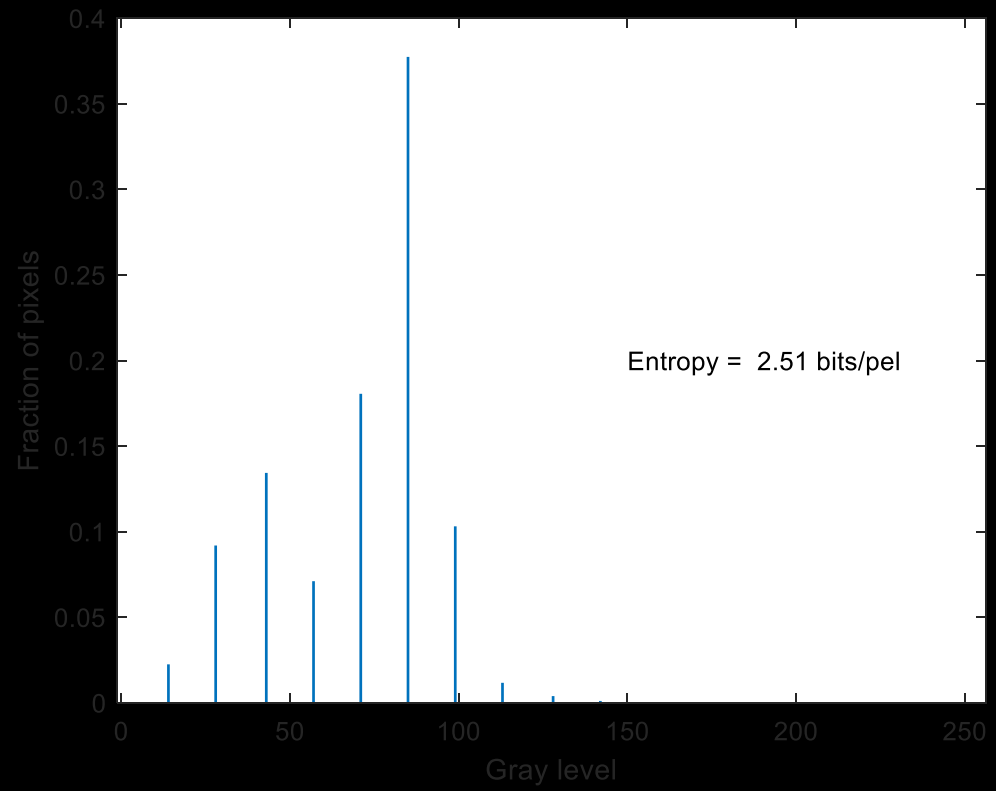
Raw 8bit image 1080p



Shannon's theory tells us that 6.28 bits/pel is the best we can do

COMPRESSION BY QUANTISING THE IMAGE ITSELF

THROW AWAY BITS IN THE RAW IMAGE TO ACHIEVE COMPRESSION. BUT IT COMES AT THE COST OF REDUCED IMAGE QUALITY



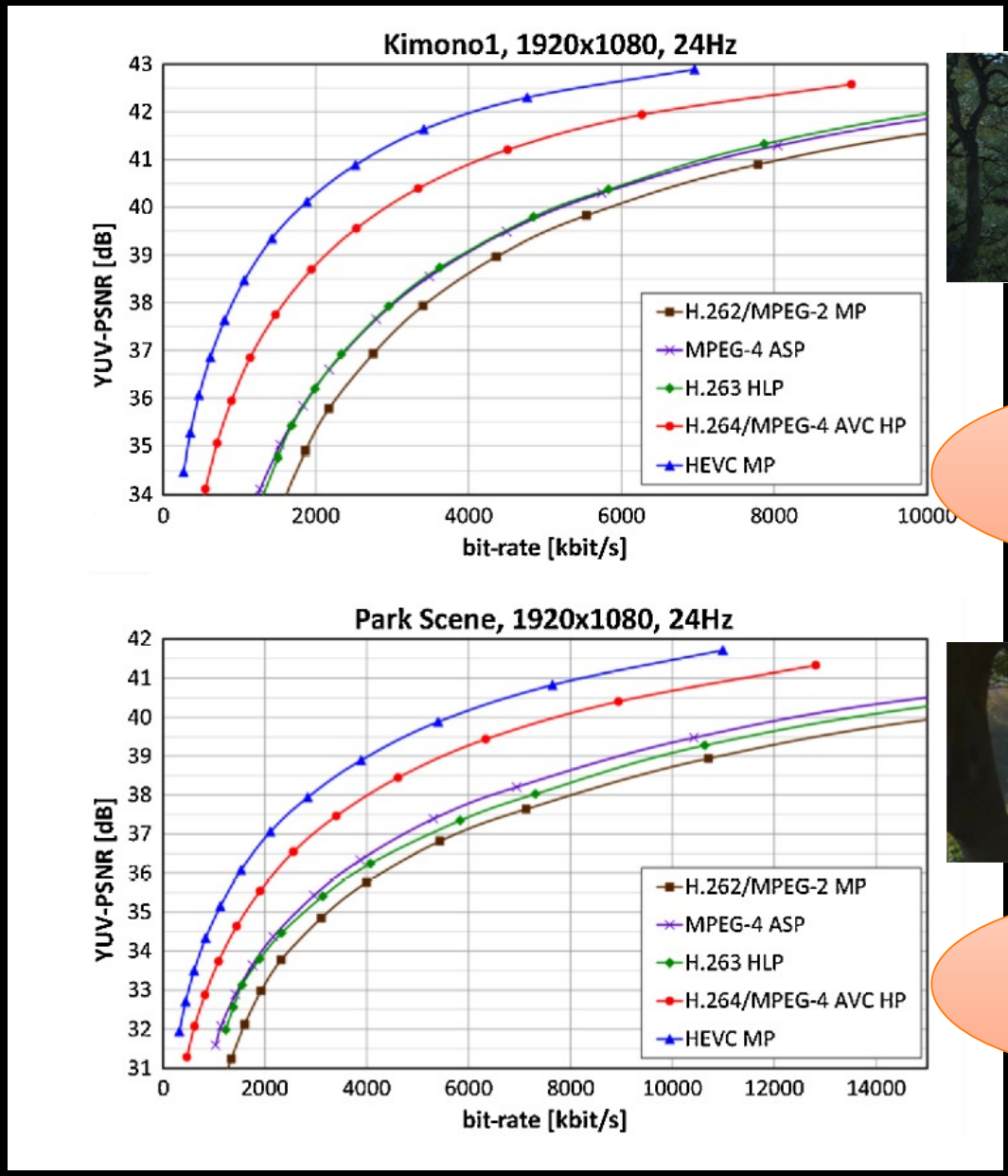
Raw 8bit image 1080p Quantised using Qstep = 15
i.e. gray values rounded off to $0: 15: 255$
PSNR doesn't seem that bad … but banding "perceptually" really bad
More evidence that PSNR isn't that great

Shannon's theory now tells us that 2.51 bits/pel is the best we can do. We have achieved compression!

COMPRESSION PERFORMANCE: RATE-DISTORTION (RD) CURVES

- Summarise the performance of a compression scheme
- VERY IMPORTANT PERFORMANCE INDICATOR
- For a particular set of parameter settings, plots Bitrate and Image Quality Achieved
- As bitrate increases, so does quality.
- These curves allow fair comparison of codecs and can be used to select appropriate settings for transcoding video content.

YOUR FIRST LAB IS ABOUT THIS



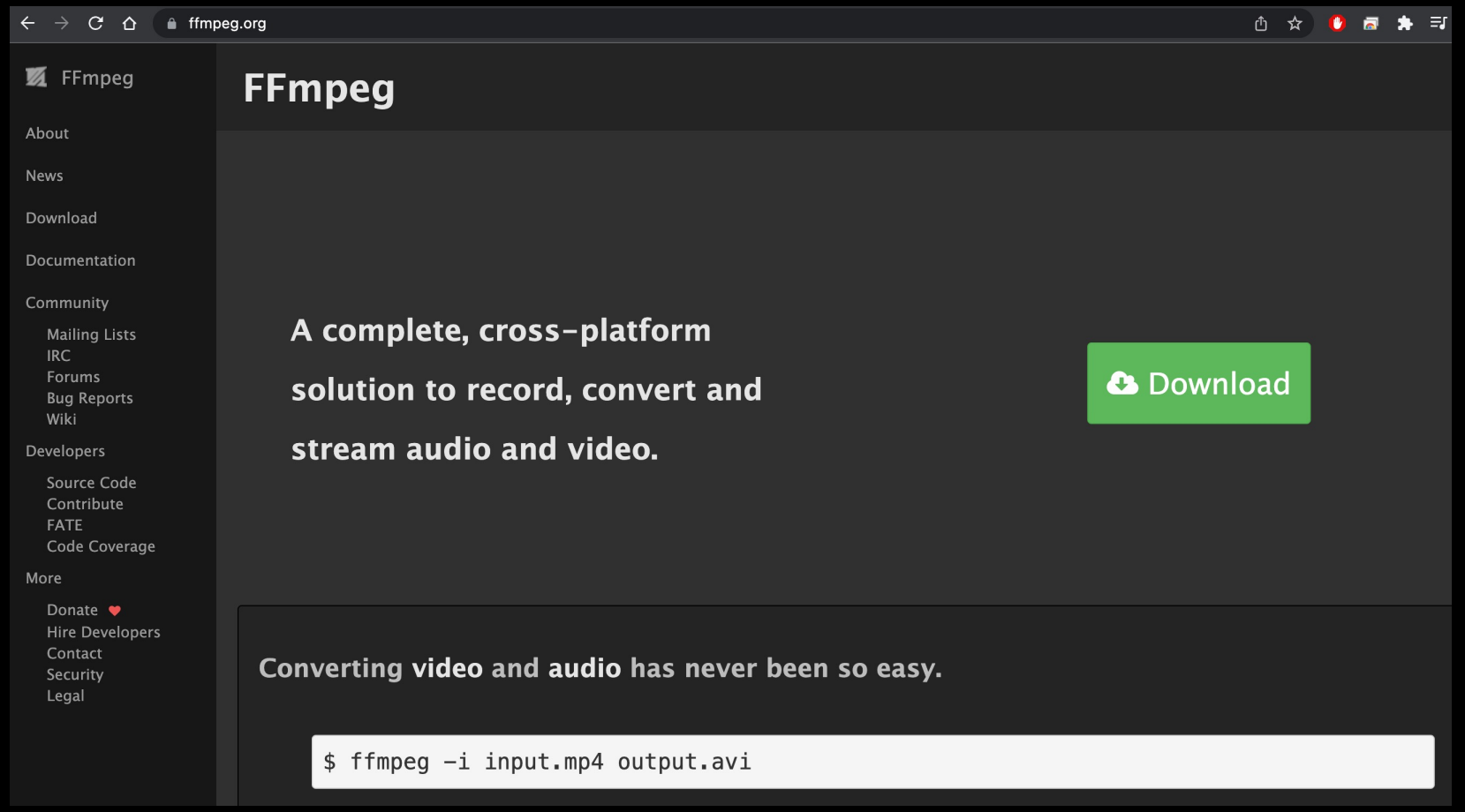
Which codec is the best for these sequences?

How can we measure the RD curves differences?

INSTRUCTIONS

- Video File: "dancing_org.mp4"
- You will need to use ffmpeg to answer some of these questions. The lab is done like a text in which you will fill in the blanks and answer questions in a short form.
- You will have 2 weeks to finish this lab. This is due on Tuesday 29th March 2021 at 12 midday. On that day you will be issued with the final assignment which should take you till the end of the semester to complete.
- This laboratory is about transcoding for streaming.
- You investigate the tradeoff between picture quality and bitrate. A typical invocation for transcoding from one format to another is "ffmpeg -i filenameIn.mov -b:v 2048k filenameout.mp4". That command line specifies the input file to be filenameIn.mov, and the output file to be filenameout.mp4. The output bit rate is specified to be 2Mbits/sec. This implies constant bitrate encoding. FFmpeg reports the average bitrate _actually_ attained, as well as average PSNR of the output file, once it is finished. You can force this with the -psnr option on the command line, and also output detailed transcoding statistics with -vstats_file (which dumps the information to a file called). Using -vcodec libx264 enables the H.264 codec in ffmpeg.
- In the laboratory you will experiment with constant QP as well as constant bitrate encoding. For constant quality, you will need to use -qp X. For constant bitrate you will need to use -b:v XXXk . If you invoke ffmpeg manually with the modified command line each time the summary statistics are printed to the screen. It will be wise to figure out how to call ffmpeg from Matlab or Python using a system call. This will allow you to invoke ffmpeg with the different bitrates and QP automatically without having to manually invoke it for every parameter setting.

FFMPEG



Either call it using this path C:\ffmpeg\bin

or :

1. Type: `setx path "%path%;c:\ffmpeg\bin"`
2. Restart Windows Terminal and should be able to call it from any directory.

CHEATSHEET

- Constant Rate:

ffmpeg -i .\dancing_org.mp4 -b:v 2048k -vcodec: libx264 -psnr out_2048k.mp4

- Constant QP:

ffmpeg -i .\dancing_org.mp4 -vcodec: libx264 -qp 35 -psnr out35qp.mp4

- Matlab command to call ffmpeg or any other command from terminal:

system('command')