

COMM901 – Source Coding and Compression

Class Project – WS 2013/14

Summary:

Your task is to develop an efficient algorithm for compression of 8-bit grayscale images of 512×512 samples. You can apply things you have learned in the course, incorporate techniques from image and video coding standards or use ideas from the literature.

Organization:

- The image coding algorithm has to be developed in **groups of 4 students**
- The efficiency of the algorithm is checked by a different group (bitstreams and decoder has to be provided to the crosscheck partner)

Implementation:

You have to implement your encoder and decoder in MATLAB. Your algorithms have to be implemented from scratch. You can use any function from any MATLAB toolbox, but are not allowed to use any software parts written by others.

Encoder: Input: Test grayscale image in PGM format and a quality parameter
 Output: Bitstream

Decoder: Input: Bitstream
 Output: Reconstructed grayscale image in PGM format

Test images are provided in PGM format. These images can be read using the “imread” function of MATLAB and they can be written using the “imwrite” function of MATLAB.

Algorithm evaluation:

For all of the provided test images you have to create a bitstream that yields

a PSNR of at least 35.00~dB.

You can control the PSNR using the quality parameter of your encoder. Note that larger PSNR values will require larger bitstreams. The efficiency of your algorithm is evaluated using the file sizes of your bitstreams (for PSNR values greater than or equal to 35 dB).

The PSNR of 8-bit grayscale image of size 512×512 is defined by

$$PSNR = 10 \cdot \log_{10} \frac{255^2}{MSE} \quad \text{with} \quad MSE = \frac{1}{512^2} \sum_{x=1}^{512} \sum_{y=1}^{512} (s[x, y] - s'[x, y])^2,$$

where $s[x, y]$ and $s'[x, y]$ represent the original and reconstructed image signal, respectively.

Provided is a Windows executable by which the PSNR between two images in PGM format can be measured. You can also implement your own tool in MATLAB.

Test images:

Six grayscale test images with a size of 512×512 samples and a bit depth of 8 bit are used for testing your algorithm. They are provided in PGM format and are shown below.



barbara



boat



clown



houses



kiel



lena

Reference for algorithm evaluation:

The goal for your image coding algorithm is to provide a coding efficiency that is at least comparable to that of a typical JPEG implementation. It is encouraged to develop an algorithm that is better than JPEG.

As a reference, the simulation results for a JPEG implementation are shown in the table below. The JPEG compression was done with the freeware program "Image Magick"; the quality is controlled by a parameter called "JPEG quality".

image	JPEG quality	file size [byte]	PSNR [dB]	rel. size
barbara	71	41 045	35.1273	15.66 %
boat	70	36 920	35.1172	14.08 %
clown	37	18 574	35.0505	7.09 %
houses	82	74 062	35.0817	28.25 %
kiel	82	59 739	35.0309	22.79 %
lena	38	16 960	35.0008	6.47 %
average		41 217	35.0681	15.72 %

Suggestion:

Start with an implementation similar to JPEG consisting of

- block-wise discrete cosine transform
- scalar quantization (fixed quantization step size for all transform coefficients); the quantization step size can be used as quality parameter
- entropy coding of quantized transform coefficients (e.g., using zig-zag scanning and run-level coding)

This first algorithm can then be later improved by several techniques.

Promising techniques may be:

- Prediction of transform coefficients from neighbouring blocks
- Block-wise spatial intra prediction
- Adaptive scanning of transform coefficients
- Improvements to entropy coding of transform coefficients, such as run-level-last coding
- Rate-distortion optimized quantization

Note: Talk to your lecturer about your ideas. Tell him about your progress and problems.

Project submission:

- MATLAB code for encoder and decoder
- Bitstream for the test images
- Presentation slides (**10 slides at maximum**)

The presentation slides have to be provided in pdf or Microsoft PowerPoint format. They should represent the project report and should contain:

- Description of algorithm
- Results for test images
- Who did what

Timeline:

Wed, Dec 11: Give bitstreams and decoder to your crosscheck partner

Fri, Dec 13: Provide your MATLAB code, bitstreams and presentation slides

Tue, Dec 17: Presentation of your work during lecture

Suggestion: Develop a first working encoder and decoder until November, 19

Project groups:

Groups	Students	Crosscheck Partner
Group 1	(a) (b) (c) (d)	Group 2
Group 2	(a) (b) (c) (d)	Group 3
Group 3	(a) (b) (c) (d)	Group 4
Group 4	(a) (b) (c) (d)	Group 1