
Algorithm 2 Actual use in transmitting and storing images with soft compression algorithm

Encoding stage:

Load codebooks

Divide a multi-component I into m single component images

Perform reversible component transformation to generate I_1, I_2, \dots, I_m

For each component I_1, I_2, \dots, I_m **do**

- 1) Obtain prediction error by using prediction coding
- 2) Map prediction error to non-negative value
- 3) Layer separation to get the shape layer and detail layer
- 4) Fill the shape layer with the shapes in the codebook for shape layer
- 5) Sort shapes used by absolute location
- 6) Use Golomb coding for location difference
- 7) Record locations and codewords of shapes used for filling
- 8) Encode the detail layer according to the codebook for detail layer
- 9) Generate the compressed data by combing the coding data of the shape layer and detail layer, as well as supplementary information

Generate the final compressed data by combing the compressed data of each component

Decoding stage:

Load codebooks

Divide the compressed data into m parts

For each part **do**

- 1) Decode the shape layer and detail layer according to codebooks
- 2) Layer merging
- 3) Inverse mapping and anti-predictive coding to recover the single component image

Reconstruct the original image by merging m single component images and transformation
