Exercises: Implementing a Codec with Prediction

The following tasks can be implemented in MATLAB.

1. Implement a codec.

   (a) Download the sample picture from the homepage http://r0k.us/graphics/kodak/kodim08.html.

   Read the image with entries denoted here by
   \[ f(i, j) \] with \( i = 1, 2, \ldots, \text{dim1} \), \( j = 1, 2, \ldots, \text{dim2} \),

   where \text{dim1} \ and \ \text{dim2} \ are the image dimensions.

   (b) Convert the image into the YCbCr color space. Use only the Luma component. In MATLAB use: rgb2ycbcr.

   (c) Partition the image into blocks of size \( 8 \times 8 \):

      i. On each block, perform a 2D Discrete Cosine Transform. In MATLAB use: dct2.

      ii. Quantize the result \( F \) for a uniform quantization parameter \( QP \in \{10, 15, 22, 32\} \):

         \[
         \text{level}(i, j) = \text{round} \left( \frac{F(i, j)}{QP} \right)
         \]

      iii. Scale the level back

         \[
         F(i, j) = \text{level}(i, j) \cdot QP
         \]

      and transform it back to spatial domain.

   (d) Put the image back together and display.

2. Implement a codec with predictive coding. Repeat steps [1a] and [1b]. Partition the image into blocks of size \( 8 \times 8 \):

   - For each block, construct the DC prediction \( P \) at the encoder side which is given by an \( 8 \times 8 \) matrix

     \[
     P_{ij} = m \quad \forall \ i, j = 1, \ldots, 8,
     \]

     where \( m \) is the average value of the boundary values on the left \( L \) and above \( T \) of the current block consisting of reconstructed samples:

     \[
     m = \frac{1}{16} \left( \sum_{i=1}^{8} T(i) + \sum_{j=1}^{8} L(j) \right).
     \]

     In case the block is located at the border of the image, set \( m = 128 \).
• Compute the residual $\text{Res}$ by subtracting prediction $P$ from the original block samples.

• Repeat steps [1(c)i] to [1(c)iii], replacing $F$ with $\text{Res}$.

• Add the result to the prediction $P$ to obtain the reconstructed sample block.

• Repeat step [1d]

3. Create a Rate Distortion Plot for 1 and 2

(a) For each $QP \in \{10, 15, 22, 32\}$ approximate the rate $R$ by calculating the number of samples times the entropy $H$

$$H = - \sum_{i=1}^{\text{dim}_{1}} \sum_{j=1}^{\text{dim}_{2}} p(i,j) \cdot \log_2(p(i,j)),$$

where $p(i,j)$ is the probability corresponding to $\text{level}(i,j)$. To approximate the probability $p$ of each level, count how often the value $\text{level}(i,j)$ occurs in the image and denote this by $\#\text{level}(i,j)$. Then it holds

$$p(i,j) = \frac{\#\text{level}(i,j)}{\text{dim}_{1} \cdot \text{dim}_{2}}$$

**Hint:** In MATLAB you can use

```matlab
hist(level,[min(level):1:max(level)]).
```

Here, make sure to take the zeros out before you calculate the entropy.

(b) For each $QP$ calculate the distortion $D$ using the mean squared error (MSE).

(c) Plot the pairs $(R, D)$ for each $QP$.

4. Questions:

(a) Compare the different Rate Distortion Curves. Which implementation is more efficient in terms of the Rate Distortion?

(b) Why did we not use the original image samples in 2 to compute the prediction?