

ECE 285, Spring 2020
Image and Video Compression
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
UNIVERSITY OF CALIFORNIA, SAN DIEGO

LAB 1: Binary Arithmetic Coding

This lab requires to compress a sequence of the nonbinary coding symbols s_m using binary arithmetic coder (BAC), which includes binary arithmetic encoder (BAE) and binary arithmetic decoder (BAD), and verify lossless compression after decoding the bitstream.

The coding symbols, s_i are 1, 5 and 7. Use the unary encoder to generate a sequence of coding bins b_m from the sequences of the coding symbols s_m , and the unary decoder to reconstruct s_m from b_m . Here, we assume all the coding symbols are positive. That is, we don't need to encode or decode the sign of the input symbols. Verify your lossless compression from your decoder.

The bin random variables, b_i are assumed to be independent and identically distributed (IID). The probability that a bin is equal to 0, $pb(0)$, where is short for $pb(b = 0)$, and the probability that a bin is equal to 1, $pb(1)$, where is short for $pb(b = 1)$, are estimated by frequency of the coding bins after binarization of all the coding symbols. In this lab, we do not update the probability of coding bins and also assume this probability information is available to the decoder. Thus, the encoder and the decoder maintain the same default probability information for the binary arithmetic coder. The decoder also knows how many symbols are coded in advance.

We limited the precision of the Range (A) and the Low (C) as 4 bits. Thus, the Range spans $(0000)_2$ to $(1111)_2$. In order to assign nonoverlapped regions to $pb(0)$ and $pb(1)$ always, use $Range0 = round(Range \times pb(0))$ for subdivision of $pb(0)$ and $Range1 = Range - Range0$ for subdivision of $pb(1)$. The renormalization is performed **whenever** the Range is **less than** half $(1000)_2$. This lab does not require that BAE and BAD support to code more than 20 coding bins (**No overflow prevention**). Overall procedure is illustrated as the below block diagram.

- You should build (program) that Unary Encoder, Unary Decoder, BAE and BAD can process any symbols and a sequence of symbols.
- You should process entire symbols at each step. For example, BAE should sequentially process all binary symbols after binarizing all coding symbols. Do not process one symbol by one symbol through entire steps from Unary Encoder to Unary Decoder.

Programming language: Matlab or Python

