Abstract: JPEG is a popular DCT-based still image compression standard, which has played an important role in image storage and transmission since its development. Advanced Image Coding (AIC) is a still image compression system which combines the intra frame block prediction from H.264 with a JPEG-based discrete cosine transform followed by context adaptive binary arithmetic coding (CABAC). It performs much better than JPEG and has the best performance at low bit rates. In this paper, we propose a modified AIC (M-AIC) by replacing the CABAC in AIC with a Huffman coder and an adaptive arithmetic coder. The results are compared with other image compression techniques like JPEG, JPEG2000, JPEG-LS, JPEG-XR and H.264 on various sets of test images. This paper considers only main and (FRExt) high profiles in H.264/AVC I-frame coding, JPEG using baseline method and all the codecs are considered in lossy compression. Simulation results are evaluated in terms of bit-rate, quality-PSNR and structural similarity index (SSIM).
Advanced image coding and its comparison with various codecs

Radhika Veerla¹, Zhengbing Zhang¹,² and K.R. Rao¹, IEEE Fellow
¹Electrical Engineering Department, University of Texas at Arlington, Arlington, TX
²Electronics and Information College, Yangtze University, Jingzhou, Hubei, China
e-mail: (radhika.veerla, zhangz, rao)@uta.edu

Abstract— JPEG is a popular DCT-based still image compression standard, which has played an important role in image storage and transmission since its development. Advanced Image Coding (AIC) is a still image compression system which combines the intra frame block prediction from H.264 with a JPEG-based discrete cosine transform followed by context adaptive binary arithmetic coding (CABAC). It performs much better than JPEG and has the best performance at low bit rates. In this paper, we propose a modified AIC (M-AIC) by replacing the CABAC in AIC with a Huffman coder and an adaptive arithmetic coder. The results are compared with other image compression techniques like JPEG, JPEG2000, JPEG-LS, JPEG-XR and H.264 on various sets of test images. This paper considers only main and (FRExt) high profiles in H.264/AVC I-frame coding, JPEG using baseline method and all the codecs are considered in lossy compression. Simulation results are evaluated in terms of bit-rate, quality- PSNR and structural similarity index (SSIM).

Index Terms— adaptive arithmetic coding, advanced image coding, block prediction, DCT, Huffman, JPEG, M-AIC, SSIM

I. INTRODUCTION

JPEG [1] is a popular DCT-based still image compression standard, which has played an important role in image storage and transmission since its development. For instance, pictures taken by most of the current digital cameras are still in JPEG format, and as a result most of the images transferred on Internet are also in JPEG format.

JPEG provides very good quality of reconstructed images at low or medium compression (i.e. high or medium bit rates respectively), but it suffers from blocking artifacts at high compression (low bit rates). Bilsen provides an experimental still image compression system known as Advanced Image Coding (AIC) that encodes color images and performs much better than JPEG and close to JPEG-2000 [2]. AIC combines intra frame block prediction from H.264 [4] with a JPEG-style discrete cosine transform, followed by context adaptive binary arithmetic coding (CABAC) used in H.264. The aim of AIC is to provide better image quality with reduced complexity. It is also faster than existing JPEG2000 codecs. In this paper, a modified AIC (M-AIC) proposed in [3] is implemented and compared with various existing image codecs like JPEG [1], JPEG2000 [10], JPEG-XR [11], JPEG-LS [12] and H.264 [4] on various sets of test images. Although these compression techniques are developed for different signals, they work well for still image compression and hence worthwhile for comparison. The primary difference of the proposed M-AIC algorithm from AIC is that the CABAC is replaced by a Huffman coder and an adaptive arithmetic coder. This paper considers only main and (FRExt) high profiles in H.264/AVC I-frame coding, JPEG using baseline method and all the codecs are considered in lossy compression. The simulation results demonstrate that M-AIC performs much better than JPEG, outperforms JPEG-LS at low bit rates, performs close to JPEG-2000 and JPEG-XR and a little bit better than AIC in some low bit rate range. H.264 outperforms any other codec under objective quality assessment. Based on SSIM distortion metric [15], the structural quality of M-AIC is similar to any other codec at lower bit range and remains in competition for the entire bit range. Thus proving AIC has better structural quality in addition to objective quality.

II. OVERVIEW OF M-AIC ALGORITHM

M-AIC [3] is based on JPEG structure to which prediction block similar to H.264 is added in order to achieve best compression capability in terms of quality factor with less complexity. The AIC, shown in Fig. 1. and M-AIC shown in Figs. 2 and 3. are developed with the key concern of eliminating the artifacts, thereby increasing quality. The predictor is composed of five parts including IDCT, inverse quantization, Mode Select and Store, Block Predict and an Adder. The function of the predictor is to predict the current block to be encoded with the previously decoded blocks of the upper row and the left column. AIC uses CABAC entropy coding which uses position of the matrix as the context while the M-AIC takes up Huffman coding and adaptive arithmetic coding in combination in order to achieve similar performance.

A. M-AIC Encoder

M-AIC uses a DCT based coder, shown in Fig. 2. DCT coding framework is competitive with wavelet transform coding if the correlation between neighboring pixels is properly considered using entropy coding. This need is taken care of M-AIC. Here, the original image is converted from RGB domain to YCbCr domain in 4:4:4 sampling format. These YCbCr blocks are divided into 8x8 non-overlapping blocks which are encoded block by block in zig-zag scan order whereas the Bilsen’s
AIC uses scan-line order. Then it is followed by encoding each block in all the channels, the first step being selection of a block prediction mode, which minimizes the prediction error measured with sum of absolute difference (SAD), by full search among the predefined 9 intra-prediction modes in [3]. The 9 block prediction modes, Mode 0 through Mode 8, represent vertical, horizontal, DC, diagonal down-left, diagonal down-right, vertical-right, horizontal-down, vertical-left and horizontal-up predictions respectively. The same selected prediction mode which is used to store and predict the current block in Y is used for corresponding Cb and Cr blocks. The prediction residual (Res) of the block to be encoded is transformed into DCT coefficients using a fast floating point DCT algorithm. Then the DCT coefficients are uniformly scalar-quantized. The same quantization parameter (QP) is used to quantize the DCT coefficients of Y, Cb and Cr channels and transferred into a one-dimensional sequence with a zig-zag scan order. All the 64 coefficients including both the DC coefficient and the AC coefficients are encoded together with the same algorithm as that for encoding the AC coefficients in JPEG standard. The proposed M-AIC algorithm makes use of the chrominance AC coefficients Huffman table recommended in baseline JPEG to encode all channels of Y, Cb and Cr [1][5]. Then the selected prediction modes are encoded with a variable length code. If the prediction mode of the current block is the same as that of the previous block, output only 1 bit of 0, else output 1 bit of 1 followed by 3 bits of the mode number message, which is the mode index itself or the mode index minus 1 if the mode index of the current block is less than or greater that that of the previous block respectively. To form a compressed stream, 11 bytes are used to construct a stream header including stream format flag, algorithm version, quantization parameter (QP), image width, image height, pixel bit-count of the original image, and the code size of the compressed modes. The compressed bitstream is orderly composed of the header, the code of the prediction modes, the Huffman codes of Y-Res, the Huffman codes of Cb-Res and the Huffman codes of Cr-Res. An adaptive arithmetic coder (AAC) is added at the end of the encoder. The AAC [13] is fed with 8-bit symbols extracted byte-by-byte from the compressed stream (the header, the code of the prediction modes, and the Huffman codes of Y-Res, Cb-Res and Cr-Res) resulting in final bit stream.

B. M-AIC Decoder
The M-AIC decoding process shown in the Fig. 3 is reverse of encoding. The coded bitstream from the encoder is fed to AAD resulting in the stream header, the code of the prediction modes and the Huffman codes of the Y-Res, Cb-Res and Cr-Res. The code of the prediction modes is decoded into prediction modes and stored. The residual of the current block is obtained by a decoding algorithm similar to baseline JPEG decoder [5]. The code of the prediction modes is decoded into prediction modes and stored. The residual of the current block is obtained by a decoding algorithm similar to baseline JPEG decoder [5]. The prediction of the current block is produced from the previously decoded blocks according to its prediction mode. The reconstructed residual is added to the prediction to result in the reconstructed current block. After the reconstruction of all the Y’, Cb’ and Cr’ blocks, they are converted back to RGB domain.
III. DISCUSSION OF CODECS USED IN COMPARISON

Transformation and coefficient encoding are the main blocks of all the codecs which play a significant role in defining the compression quality of the system. Now, let us look at various codecs and their structures to study their impact on the compression quality and reconstruction. The reconstruction includes how each method is designed to avoid different kinds of artifacts. JPEG standard [1] is used in popular baseline mode which supports only lossy compression and gives good compression results with least complexity. It is based on block-based 8x8 DCT followed by uniform quantization, zig-zag scanning and Huffman entropy coding. JPEG2000 standard is considered for lossy compression. It is based on discrete wavelet transform, scalar quantization, context modeling, arithmetic coding and post compression rate allocation. JPEG2000 provides for resolution, SNR, parseable code-streams, error-resilience, arbitrarily shaped region of interest (ROI), random access (to the sub-band block level), lossy and lossless coding, etc., all in a unified algorithm. JPEG-XR standard [11] which supports HD photo file format, is designed explicitly for next generation of digital cameras and for storage of continuous-tone photographic content based extensively on Microsoft HD photo technology. It shares some of the features from JPEG2000 like bit-rate scalability, editing, region-of-interest decoding, integer implementation without division etc. on top of compression capability. HD photo minimizes objectionable spatial artifacts preserving high frequency detail and outperforms other lossy compression technologies in this regard. HD photo is a block-based image coder: color conversion, reversible integer-to-integer-mapping lapped bi-orthogonal transform (LBT), adaptive coefficient scanning, flexible scalar quantization, inter-block coefficient prediction and adaptive VLC table switching for entropy coding. JPEG XR supports a number of advanced pixel formats in order to avoid limitations and complexities of conversions between different unsigned integer representations allowing flexible approach to numerical encoding of image data enabling it to be used for low-complexity implementations in the encoder and decoder. JPEG-LS standard [12] is based on LOCO-I algorithm proposed by Hewlett Packard. JPEG-LS is based on prediction, context modeling and Golomb coder. It is used in near-lossless mode where the reconstructed image component is differed from original by a factor “NEAR”. The near-lossless compression has the feature to increase the compression ratio and speed of execution by specifying the tolerance error. It works well for cost sensitive applications which do not need ROI and error resilience. H.264 or MPEG-4 part 10 [4] is considered in this paper in main and high profiles in 4:2:0 sampling format. Significant coding efficiency is obtained using adaptive (directional) intra prediction and CABAC entropy coding in this codec. The result of applying spatial prediction and wavelet like 2-level transform iteration is effective in smooth image regions. This feature enables H.264 to be competitive with JPEG2000 in high resolution, high quality applications.
IV. EVALUATION METHODOLOGY

A. Codec Setting

In the coding simulations, publicly available software implementations are used for AIC, H.264/AVC, JPEG-baseline, JPEG2000, HD photo and JPEG-LS. Reference software (JM 13.2) (latest is JM 14.1) [6] is used for H.264/AVC encoder, and each frame of the test sequences is coded in the I–frame mode. For JPEG, JPEG baseline reference software [5] is used. This software can handle image data in many formats like PGM/PPM, GIF, windows BMP. For JPEG 2000 coding, M.D. Adams “JasPer” (version 1.900.1) software [7] is used. This software can handle image data in many formats like PGM/PPM, windows BMP, but it does not accept all the BMP files. In JPEG 2000, it is used to code each frame to reach target rate specification in terms of compression factors, which is well defined for multi-component images. HD photo reference software [8] supports BMP, TIF and HDR formats. Both JPEG and HD photo reference softwares are used to code each frame to reach the target quality factor to indirectly control bit rate for lossy coding. JPEG-LS reference software [9] provided by HP labs is used for lossy compression. It supports only PGM/PPM image formats as input to the encoder and JLS format as output at the encoder.

The configuration of the H.264/AVC JM13.2 encoder [6] is chosen as follows:
ProfileIDC = 77 # Profile IDC (77=main, FREXT Profiles: 100=High)
LevelIDC = 40 # Level IDC (e.g. 20 = level 2.0)
IntraProfile = 1 # Activate Intra Profile for FRExt (0: false, 1: true)
Deblocking filter = off
QPISlice = 12 # Quant. param for I Slices (0-51)
YUVFormat= 1 # YUV format (1=4:2:0, 3=4:4:4)

The command line arguments for JM13.2 software are:
Encoder: lencod –f encoder.cfg
Decoder: ldecod - i bitstream.264 - o output.yuv –r reference (input).yuv

For Microsoft HD Photo [8], all options are set to their default values with the only control coming from the quality factor setting:
• No tiling
• One-level of overlap in the transformation stage
• No color space sub-sampling
• Spatial bit-stream order
• All sub-bands are included without any skipping
WMPEncApp command line converts certain uncompressed file formats into equivalent HD photo files. WMPDecApp command line converts HD photo files to different uncompressed file formats.

The settings for JPEG-LS software [9] are as follows at the encoder. Decoder settings need not be changed from default as they follow the encoder settings.
• Images should be in ppm or pgm format.
• Line interleaved mode is considered in the project.
• Error value is varied from 1 to 60. Error value of zero corresponds to no compression.

• T1, T2, T3 are thresholds. While giving the settings the following condition need to be met. Error value+1<T1<T2<T3.
• Default RESET value of 64 is considered in this paper.

V. SIMULATION RESULTS

Several test images including Lena of different resolutions, Airplane, Couple, Peppers, Splash and Sailboat, Cameraman and Man have been tested based on the simulation platform built for M-AIC [3] encoder and decoder using Microsoft Visual C++ and using reference softwares such as AIC reference software for AIC, JPEG-baseline reference software for JPEG-baseline, Jasper software for JPEG2000, HD photo reference software for JPEG-XR, JPEG-LS LOCO-I for JPEG-LS. We include some of the simulation results based on the objective and structural quality assessment like PSNR and SSIM [15] respectively. The decoded outputs of various codecs are shown in Fig. 4 and Fig. 5(a). Fig. 5(b) shows the SSIM map. The simulation results based on PSNR and SSIM are shown in Fig. 6 and Fig. 7. respectively. Table I shows the simulation results of M-AIC based on PSNR and SSIM for varying quantization parameter.

Fig. 4. (a) Original Image, (b) Reconstructed output for Lena (512x512x24) image using H.264 main profile with quantization parameter -16, 2.83bpp, 46.81dB, M-SSIM - 0.97, (c) JPEG-baseline with quality - 94, 2.94bpp, 35.6dB, M-SSIM - 0.926, (d) JPEG2000 with rate - 0.12, 2.95bpp, 37.53dB, M-SSIM - 0.923 (e) JPEG-XR with quality – 28, 2.88bpp, 37.74dB, M-SSIM – 0.928, (f) JPEG-LS with error value – 11, 2.8bpp, 32.425dB, M-SSIM – 0.818.

Fig. 5. (a) Reconstructed output of M-AIC for Lena (512x512x24) image with quantization parameter – 5, 2.37bpp, 36.61dB (b) SSIM map of M-AIC for Lena (512x512x24) image with M-SSIM – 0.914
Fig. 6. (a) Simulation results for Lena (512x512x24) image based on objective quality metric (PSNR).

Fig. 6. (b) Simulation results for Airplane (512x512x24) image based on objective quality metric (PSNR).

Fig. 6. (c) Simulation results for Peppers (512x512x24) image based on objective quality metric (PSNR).

Fig. 6. (d) Simulation results for Sailboat on Lake (512x512x24) image based on objective quality metric (PSNR).

Fig. 6. (e) Simulation results for Couple (256x256x24) image based on objective quality metric (PSNR).

Fig. 6. (f) Simulation results for Cameraman (256x256x8) image based on objective quality metric (PSNR).
TABLE I

<table>
<thead>
<tr>
<th>M-AIC parameter</th>
<th>size bit rate</th>
<th>PSNR</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>384142</td>
<td>11.65</td>
<td>44.11</td>
</tr>
<tr>
<td>2</td>
<td>253035</td>
<td>7.72</td>
<td>41.39</td>
</tr>
<tr>
<td>3</td>
<td>177223</td>
<td>5.41</td>
<td>39.21</td>
</tr>
<tr>
<td>5</td>
<td>96647</td>
<td>2.37</td>
<td>36.61</td>
</tr>
<tr>
<td>10</td>
<td>30246</td>
<td>1.07</td>
<td>34.01</td>
</tr>
<tr>
<td>30</td>
<td>8894</td>
<td>0.27</td>
<td>30.16</td>
</tr>
<tr>
<td>70</td>
<td>4998</td>
<td>0.15</td>
<td>26.91</td>
</tr>
<tr>
<td>90</td>
<td>3985</td>
<td>0.12</td>
<td>25.61</td>
</tr>
</tbody>
</table>

VI. CONCLUSIONS AND FUTURE WORK

The M-AIC algorithm was successfully implemented to obtain better compression with reduced complexity compared to existing codecs in terms of PSNR and best performance at low bit rates in addition to being competitive with any other codec in terms of SSIM over the entire bit range. The proposed algorithm is compared with the H.264/MPEG-4 part 10 AVC JM reference software [13.2] [6], JPEG-2000 JasPer software [7], JPEG reference software [5], JPEG-XR HD-photo reference software [8] and JPEG-LS LOCO-I software [9]. Thus by its performance, it finds wide range of applications in digital camera market, internet browsing, multimedia products such as mobile phones and entertainment appliances. From the results, it is observed that M-AIC is suitable for web images as it gives best outputs for low resolution images. This algorithm can be extended to compare the lossless compression. The implementation of CABAC can be a future study.

REFERENCES

K.R. Rao received his PhD degree in electrical engineering from the University of New Mexico, Albuquerque in 1966. Since 1966, he has been with The University of Texas at Arlington where he is currently a professor of electrical engineering. He has co-edited handbooks “The transform and data compression handbook” (CRC Press, 2001) and “Digital video image quality and perceptual coding” (with H.R. Wu, Taylor and Francis, 2006). He has published “Introduction to multimedia communications: applications, middleware, networking” (with Z.S. Bojkovic and D.A. Milovanovic, Wiley, 2006) and “Discrete cosine and sine transforms” (with V. Britanak and P. Yip, Elsevier, 2007). He has conducted workshops/tutorials on video/audio coding/standards worldwide. He has published extensively in refereed journals and has been a consultant to industry, research institutes and academia. He is a Fellow of the IEEE.
Radhika Veerla received her MS degree in Electrical Engineering from the University of Texas at Arlington, US in 2008, with her major being Image Processing and Wireless Communications. She received her Bachelor of Technology in Electronics and Communications Engineering from Jawaharlal Nehru Technological University, Hyderabad, India in 2006. Her research interests include image and video coding and wireless technologies - OFDM and MIMO.
Zhengbing Zhang received his PhD degree in Communication and Information Systems from Huazhong University of Science and Technology, China in 1998. Since 1998, he has been with Yangtze University where he is currently a professor of electrical engineering. In 2008, he is with the University of Texas at Arlington as a visiting professor of electrical engineering, supported by Yangtze University. His research interests include image and video coding, voice and video over IP, seismic data acquisition and compression.