

NGHIÊN CỨU CẢI TIẾN CHẤT LƯỢNG NÉN ẢNH TRONG LCD

NGUYỄN HỮU TÀI¹, NGUYỄN THỊ HOÀNG LAN²

¹*Khoa Công nghệ Thông tin, Đại học Khoa học Huế*

²*Đại học Bách khoa Hà Nội, Viện CNTT & TT*

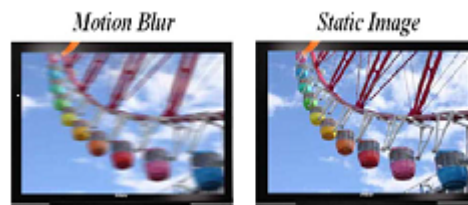
Tóm tắt. Trong bài báo này, chúng tôi trình bày nghiên cứu về kỹ thuật nén ảnh ứng dụng trong hệ thống overdrive của các thiết bị màn hình tinh thể lỏng (LCD). Overdrive trong LCD là hệ thống khắc phục hiện tượng hình ảnh chuyển động bị mờ khi hiển thị trên LCD. Đây là vấn đề đang được quan tâm nghiên cứu trong những năm gần đây. Các mô hình nén ảnh trong LCD đã được đề xuất bởi nhiều tác giả khác nhau. Trên cơ sở nghiên cứu các mô hình, chúng tôi đề xuất một cải tiến của mô hình Advanced Hybrid Image Codec (AHIC) vốn là một mô hình có cấu trúc đơn giản nhưng lại cho tỷ số nén cao và chất lượng ảnh nén khá tốt. Mô hình cải tiến được gọi tên là Mean-Adaptive Image Coding (MAIC), có khả năng thích nghi cao nhằm giảm tác động của nhiễu, loại bỏ được hiệu ứng nhiễu khối (blocking effect) mà AHIC mắc phải. Kết quả thử nghiệm trên 20 chuỗi khung hình chuyên dụng trong lĩnh vực xử lý video được chọn lọc ngẫu nhiên cho thấy, MAIC đã loại bỏ được hoàn toàn hiện tượng nhiễu khối, là loại nhiễu mà mắt người rất nhạy cảm khi quan sát trên chuỗi khung hình chuyển động, giúp hệ thống Overdrive hoạt động chuẩn xác hơn để mang lại hình ảnh chất lượng cao cho LCD.

Abstract. In this paper, we present a research on image compression technique in overdrive system of liquid crystal displays (LCD). The overdrive in LCD is a technique to reduce the response time of LCD, to remove drawback of motion blur caused by the slow response of liquid crystals. This problem has been studied in recent years. Image compression models in LCD are suggested in this paper. Based on research on the models, we propose a promotion of Advanced Hybrid Image Codec (AHIC), which has a rather simple structure, high rate of compression and rather good quality of compressed image. The promotion models is called Mean-Adaptive Image Coding (MAIC). It has high adaptive ability, so it can reduce effect of noise. Furthermore, it can remove blocking effect which has much in AHIC. To evaluate the proposed MAIC models, we use 20 test image sequences with different features. This is the set of dedicated image, which is used for evaluation in image or video processing. Experimental results show that MAIC completely remove blocking effect error. This is a kind of error which is sensitive to human visual system, especially when watching motion image sequences. Due to these improvements, MAIC can bring exact activity for Overdrive system and lead to enhancing image quality in LCD.

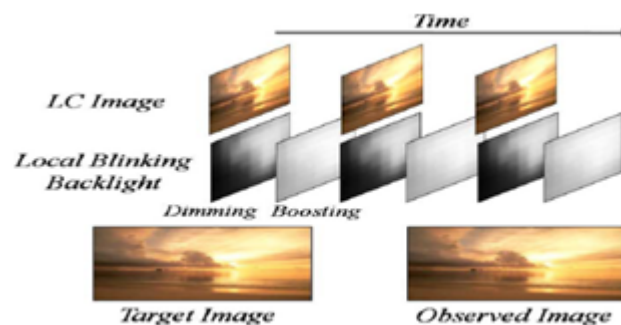
Keyword. Embedded Frame Delay Memory, LCD, Overdrive technique, Fixed Word Length Compression, Image Adaptive Coding.

1. GIỚI THIỆU

Ngày nay, màn hình tinh thể lỏng LCD (Liquid Crystal Displays) đã trở nên rất phổ biến trên thị trường thiết bị giải trí gia dụng cũng như trong công nghiệp. Tuy nhiên, điểm yếu của LCD là hiện tượng hiển thị không rõ nét đối với các cảnh chuyển động, được gọi là hiệu ứng motion-blur, được minh họa qua hình 1.1. Nguyên nhân là do các phần tử tinh thể lỏng đáp ứng lại sự thay đổi độ sáng khi chuyển đổi từ khung hình trước đó sang khung hình hiện thời với một tốc độ chậm (slow response). Để tăng tốc độ đáp ứng của các phần tử tinh thể nhằm khắc phục hiện tượng motion-blur, người ta áp dụng 2 kỹ thuật chính: Kỹ thuật thứ nhất là tăng tần xuất khung hình (high frame rate technology) bằng cách chèn khung hình thúc đẩy (boosting frame) vào giữa nhằm tăng tốc độ đáp ứng để tạo ra hình ảnh chuyển động mượt và sắc nét, được thể hiện qua hình 1.2 [7]; Kỹ thuật thứ 2 có tên gọi là overdrive, được thể hiện qua hình 1.3, sử dụng cơ chế mở rộng sự thay đổi giá trị độ sáng. Kỹ thuật overdrive đã được Oura cùng Nakanishi và các đồng nghiệp phát triển. Nó đòi hỏi phải biết giá trị độ sáng của cùng một điểm ảnh trên 2 khung hình: khung hình trước đó (previous frame) và khung hình hiện tại (current frame), để từ đó đưa ra giá trị mở rộng (hiệu chỉnh) thông qua một bảng tra được thiết lập sẵn (lookup table).



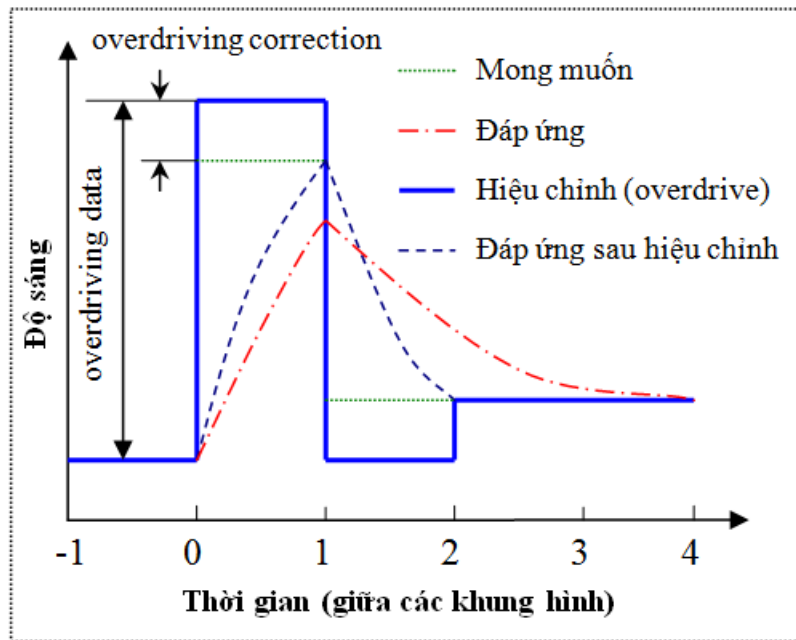
Hình 1.1. Minh họa hiện tượng Motion-Blur trên LCD



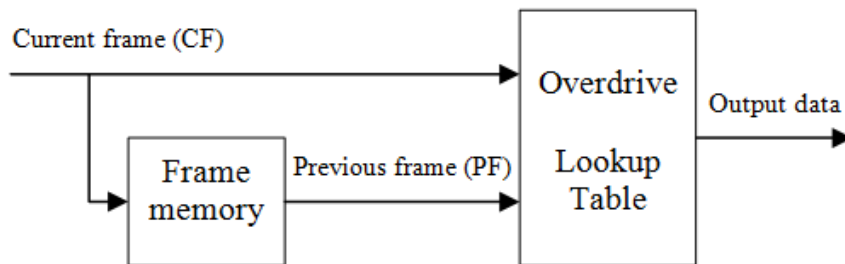
Hình 1.2. Minh họa cơ chế tăng tần xuất khung hình (high frame rate technology) trong LCD [7]

Hình 1.4 mô tả sơ đồ một hệ thống Overdriver, trong đó khối frame memory dùng để lưu trữ khung hình nhằm cung cấp thông tin phục vụ cho khối Overdrive.

Khi các chuẩn định dạng hình ảnh giải trí và truyền hình độ phân giải cao (High Definition & Full High Definition TV) ra đời và đi vào cuộc sống trên các sản phẩm thiết bị LCD có độ phân giải cao. Điều đó cũng đồng nghĩa với việc kích thước frame memory tăng lên, đẩy giá



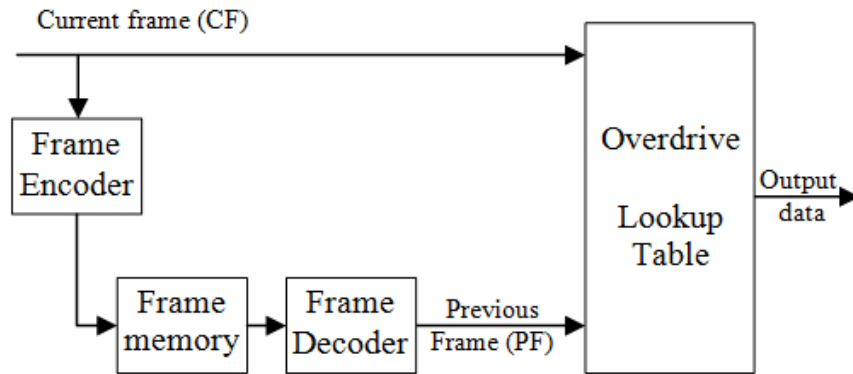
Hình 1.3. Minh họa cơ chế Overdrive trong LCD.



Hình 1.4. Sơ đồ hệ thống Overdrive.

thành của hệ thống overdrive tăng theo. Để giảm kích thước frame memory, nhằm giảm giá thành sản phẩm người ta tích hợp bộ phận nén (frame encoder) và giải nén (frame decoder) khung hình như trong hình 1.5. Một kỹ thuật nén muốn được áp dụng trên hệ thống overdrive cho LCD phải đảm bảo hai yếu tố: Thứ nhất, tỷ lệ nén cao song phải ít mất mát thông tin, nhằm đảm bảo hoạt động chính xác của hệ thống overdrive; Thứ hai, mô hình nén phải có tính giản tiện với độ dài từ mã cố định (**Fixed Word Length Compression**) thì chúng sẽ được thực hiện trên phần cứng một cách đơn giản nên cho chi phí thấp.

Chúng tôi đã nghiên cứu một số kỹ thuật nén khác nhau đã được đề xuất áp dụng cho hệ thống overdriver trong LCD như: “Scalable DCT-based compression” được trình bày trong [1]; “Fast discrete wavelet transform” được trình bày trong [3]; “Vector quantizer based block truncation coding for color image compression” được đề xuất trong [5]; hay giải pháp “Color space conversion” [2]. Trên cơ sở nghiên cứu tài liệu chúng tôi tập trung vào mô hình “Advanced Hybrid Image Codec” gọi tắt là AHIC đã được đề cập trong các bài báo [4] và [6]. Mô hình AHIC được đề xuất bởi J. Wang và J. W. Chong [6] có cấu trúc đơn giản nhưng lại cho tỷ số

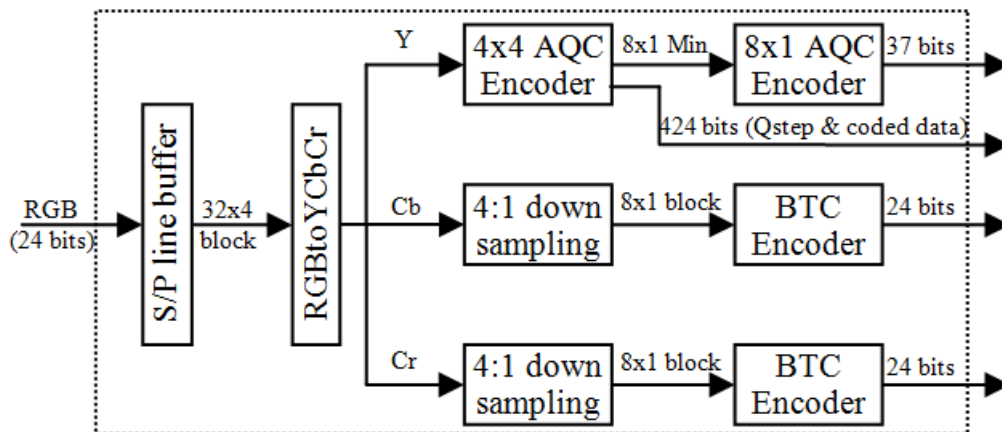


Hình 1.5. Sơ đồ hệ thống Overdrive có sử dụng kỹ thuật nén ảnh.

nén cao và chất lượng ảnh nén khá tốt. Kế thừa và phát triển mô hình AHIC của hai tác giả trên, chúng tôi nghiên cứu đề xuất mô hình Mean-Adaptive Image Coding (MAIC), nhằm loại trừ hiện tượng nhiễu khối (blocking effect) trên thành phần độ chói của ảnh, giúp cải thiện hơn nữa chất lượng ảnh nén khi so sánh với AHIC. Qua các kết quả thực nghiệm cho thấy MAIC đã cho chất lượng ảnh nén cao hơn, trung bình vào khoảng 1.34dB nếu chỉ xét trên thành phần ảnh độ chói (còn được gọi là Y/luma component), hay 0.45dB nếu xét trên ảnh màu RGB. Các phần dưới đây chúng tôi sẽ trình bày chi tiết các nghiên cứu.

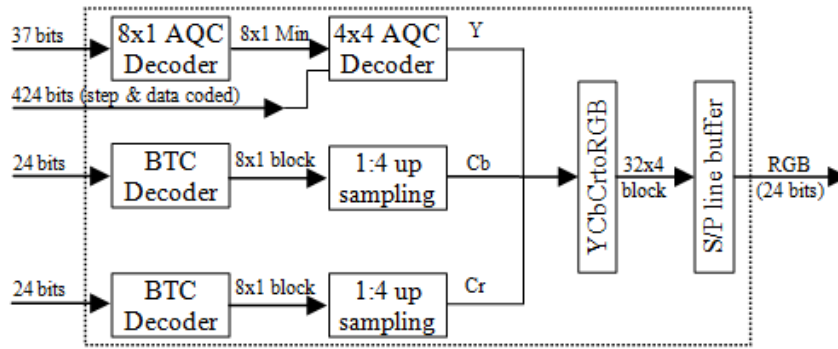
2. MÔ HÌNH AHIC

Sơ đồ khối cho AHIC Encoder và AHIC Decoder được đề xuất bởi Wang và Chong được mô tả qua hình 2.6 và hình 2.7 [6]. Trong đó AQC là một bộ mã hóa lượng tử thích nghi (Adaptive Quantization Coding) [4] được thể hiện qua hình 2.8. Thông tin chi tiết hơn về AQC có thể tham khảo trong [10].

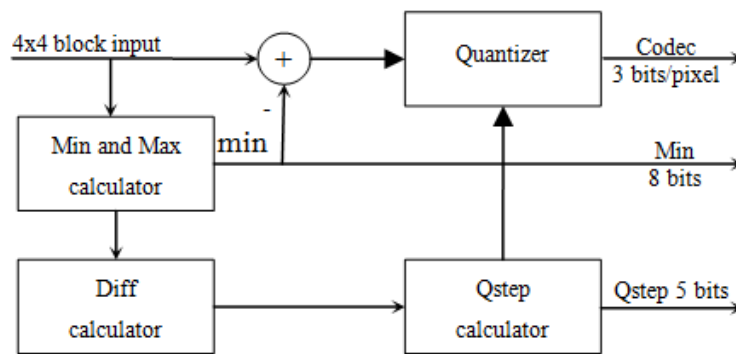


Hình 2.6. Sơ đồ bộ nén của AHIC (AHIC Encoder).

Sau khi xem xét mô hình AHIC, chúng tôi nhận thấy nó còn tồn tại một số nhược điểm



Hình 2.7. Sơ đồ bộ giải nén của AHIC (AHIC Decoder).



Hình 2.8. Sơ đồ bộ lượng tử hóa AQC.

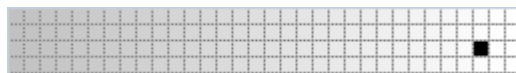
sau:

(1) Để đảm bảo tổng số bit ở đầu ra không vượt quá 512 bit, 8 giá trị Min có dung lượng 64 bit phải được mã hóa bởi AQC để giảm dung lượng còn lại 37 bit. Sau đó chúng sẽ được giải mã bởi AQC. Thông thường thì giá trị Min khôi phục lại sẽ sai lệch so với giá trị ban đầu, sự sai lệch này sẽ tác động đồng thời lên 16 giá trị trong khối, và tạo ra hiệu ứng Blocking khi giá trị giải mã Min được sử dụng để giải mã cho 4x4 giá trị điểm ảnh đã được mã hóa trước đó phụ thuộc vào Min.

(2) Khi ảnh có nhiễu, đặc biệt với loại nhiễu muối tiêu (salt & pepper noise), giá trị Min của block 4x4 sẽ dễ bị tác động. Với 8 giá trị Min đã bị tác động bởi nhiễu, nó có khuynh hướng làm cho khoảng chênh lệch (diff) lớn lên, dẫn đến khoảng lượng hóa (Qstep) cũng lớn theo và hệ quả là 8 giá trị Min được mã hóa với sai số lớn. Vì vậy khi giải mã 32x4 giá trị trong 8 khối sẽ xuất hiện sai số lớn (như đã trình bày trong mục (1)).

Để hiểu rõ hơn vấn đề chúng ta xem xét qua một ví dụ sau:

Ví dụ 1. Xét một vùng ảnh độ chói (Y-component) đầu vào với kích thước 32x4 như hình sau:



Đây là một ảnh thể hiện sự thay đổi đều của độ chói từ tối đến sáng, song gần đến cuối

ảnh có sự xuất hiện của một điểm đen. Dưới góc độ xử lý tín hiệu ảnh, điểm ảnh màu đen xuất hiện trong vùng ảnh trên có thể là kết quả tác động của nhiễu, hay còn được gọi là điểm nhiễu.

Ma trận số biểu diễn cho ảnh trên là:

```
193 195 197 199 201 203 205 207 209 211 213 215 217 219 221 223 225 227 229 231 233 235 237 239 241 243 245 247 249 251 253 255
193 195 197 199 201 203 205 207 209 211 213 215 217 219 221 223 225 227 229 231 233 235 237 239 241 243 245 247 249 251 253 255
193 195 197 199 201 203 205 207 209 211 213 215 217 219 221 223 225 227 229 231 233 235 237 239 241 243 245 247 249 0 253 255
193 195 197 199 201 203 205 207 209 211 213 215 217 219 221 223 225 227 229 231 233 235 237 239 241 243 245 247 249 251 253 255
```

Mã hóa 8 block trên bởi AQC encoder 3bit/value sẽ thu được các giá trị dữ liệu được mã hóa cho các điểm ảnh cùng giá trị Min và Qstep của chúng.

Coded data cho 8 block:

```
0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 7 7 7 7
0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 7 7 7 7
0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 7 0 7 7
0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 0 2 4 6 7 7 7 7
```

(a)

Và 8 giá trị Min đi kèm:

```
193 201 209 217 225 233 241 0
```

(b)

Cùng 8 giá trị Qstep:

```
1 1 1 1 1 1 1 32
```

8 giá trị Min ở (b) cần được nén lại trong 37 bit bởi bộ mã hóa AQC encoder 3bit/value. Chúng ta thấy nếu chỉ xét 7 giá trị đầu là 193 201 209 217 225 233 241, thì $\text{Diff} = \text{Max} - \text{Min} = 241 - 193 = 48$. Song sự xuất hiện của giá trị thứ 8 có giá trị 0, cũng chính là giá trị nhiễu xuất hiện trong block thứ 8, đã làm cho giá trị Diff xét trên 8 giá trị Min lúc này thay đổi mạnh.

$$\text{Diff} = \text{Max} - \text{Min} = 241 - 0 = 241$$

Đưa đến giá trị bước lượng tử là:

$$\text{Qstep} = \text{round} \lfloor \text{Diff} / 7 \rfloor = 34$$

Giá trị này vượt quá khả năng biểu diễn trong 5bit (từ 1 đến 32 vì vậy kết quả nó cần được cắt thành 32. Lượng tử hóa 8 giá trị Min trên với $\text{Qstep} = 32$ chúng ta thu được coded data cho 8 giá trị Min là:

```
Coded data:
6 6 7 7 7 7 7 0
```

(c)

Giải mã các giá trị ở (c) ta khôi phục lại gần đúng 8 giá trị Min ban đầu:

```
192 192 224 224 224 224 224 0
```

(d)

Với sai số trong mã hóa các giá trị Min là:

```
-1 -9 15 7 -1 -9 -17 0
```

(e)

Sai số này có thể nói là rất lớn do sự tác động của giá trị Min thứ 8 là 0, mà cũng chính là giá trị nhiễu tác động lên ảnh, giá trị 0 này đã kéo giá trị Diff lên 241, đưa đến bước lượng tử hóa $\text{Qstep} = 34$, gây ra sai số lớn nhất lên đến $1/2\text{Qstep} = \pm 17$.

Theo sơ đồ AHIC chúng ta buộc phải sử dụng 8 giá trị Min khôi phục được ở (d) để giải mã các giá trị đã được mã hóa ở (a). Kết quả chúng ta khôi phục lại được 8 block 4x4 đầu vào với giá trị như sau:

Reconstructed data:

```

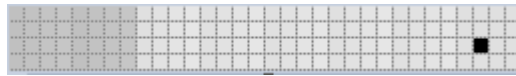
192 194 196 198 192 194 196 198 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 224 224 224
192 194 196 198 192 194 196 198 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 224 224 224
192 194 196 198 192 194 196 198 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 0 224 224
192 194 196 198 192 194 196 198 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 226 228 230 224 224 224 224
    
```

Sai số trong mã hóa 8 block sẽ là:

```

-1 -1 -1 -1 -9 -9 -9 -9 15 15 15 15 77 77 -1 -1 -1 -1 -9 -9 -9 -9 -17 -17 -17 -17 -25 -27 -29 -31
-1 -1 -1 -1 -9 -9 -9 -9 15 15 15 15 77 77 -1 -1 -1 -1 -9 -9 -9 -9 -17 -17 -17 -17 -25 -27 -29 -31
-1 -1 -1 -1 -9 -9 -9 -9 15 15 15 15 77 77 -1 -1 -1 -1 -9 -9 -9 -9 -17 -17 -17 -17 -25 0 -29 -31
-1 -1 -1 -1 -9 -9 -9 -9 15 15 15 15 77 77 -1 -1 -1 -1 -9 -9 -9 -9 -17 -17 -17 -17 -25 -27 -29 -31
    
```

Ảnh khôi phục cho thành phần Y



Khi loại bỏ điểm nhiễu màu đen ở ảnh đầu vào bởi giá trị trung bình của các điểm lân cận, kết quả quá trình mã hóa sẽ cho ra ảnh khôi phục giống hoàn toàn với ảnh đầu vào. Hay nói cách khác lúc này quá trình mã hóa không bị sai số.

Khi loại bỏ điểm nhiễu màu đen ở ảnh đầu vào bởi giá trị trung bình của các điểm lân cận, kết quả quá trình mã hóa sẽ cho ra ảnh khôi phục giống hoàn toàn với ảnh đầu vào. Hay nói cách khác lúc này quá trình mã hóa không bị sai số.

Nhận xét. Rõ ràng là sai số trong việc mã hóa giá trị Min của 8 block sau đó đã tác động trực tiếp lên đồng loạt 16 giá trị của block. Tác động kép của sai số trong lượng hóa các điểm ảnh và sai số trong lượng hóa các giá trị Min đã khiến cho sai số trong mã hóa ảnh tăng vọt, và tạo ra hiệu ứng khối trên ảnh mà chúng ta có thể nhìn thấy qua ảnh khôi phục.

Nhằm khắc phục nhược điểm của AHIC để nâng cao hơn nữa chất lượng ảnh nén, chúng tôi đã nghiên cứu và đề xuất mô hình mới MAIC.

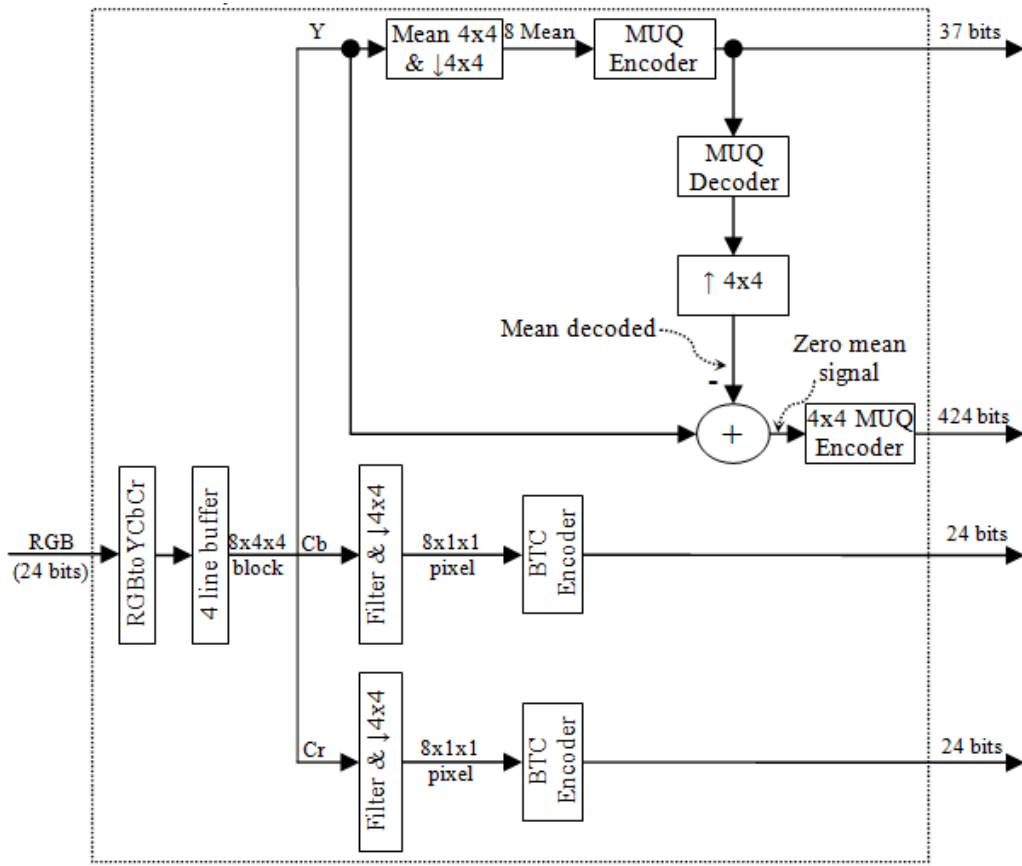
3. ĐỀ XUẤT MÔ HÌNH MỚI

3.1. Mean Adaptive Image Coding (MAIC)

Trên cơ sở AHIC của Jun Wang và Jong-Wha Chong, chúng tôi đề xuất MAIC được thể hiện qua các sơ đồ được trình bày trên hình 3.9 và hình 3.10. Có ba điểm khác biệt của mô hình mới MAIC so với mô hình AHIC như sau: Thứ nhất, thay vì dùng giá trị Min Max là những giá trị thường thiếu ổn định do dễ bị tác động bởi nhiễu, chúng tôi dùng giá trị trung bình (mean) là giá trị có tính ổn định cao hơn trong tín hiệu ảnh, nó ít bị tác động bởi nhiễu. Thứ hai, chúng tôi thực hiện trừ giá trị tín hiệu đầu vào bởi giá trị trung bình đã giải mã (mean decoded) để thu được tín hiệu "zero mean" trước khi được lượng tử hóa với 3bit/giá trị, bằng con đường này, sẽ tránh được hiệu ứng lỗi kép khi giải mã các giá trị điểm ảnh, vốn là nguyên nhân gây ra nhiễu khối mà mô hình AHIC mắc phải. Thứ ba, là chúng tôi sử dụng bộ lượng tử hóa MUQ (Midrise Uniform Quantizer) thay vì sử dụng AQC như trong AHIC.

3.2. Midrise Uniform Quantizer

Midrise Uniform Quantizer là một phương pháp lượng tử hóa được đề cập trong [8], cho phép lượng tử hóa các giá trị có phân phối xác suất đều với sai số tối thiểu. Quy trình thực



Hình 3.9. Sơ đồ bộ nén MAIC (MAIC Encoder).

hiện như sau:

Lượng tử hóa với b bit, hay $L=2^b$ là số mức lượng tử.

Gọi ZMI là giá trị đầu vào có dạng zeros mean signal & QS là bước lượng tử (quantifier step size)

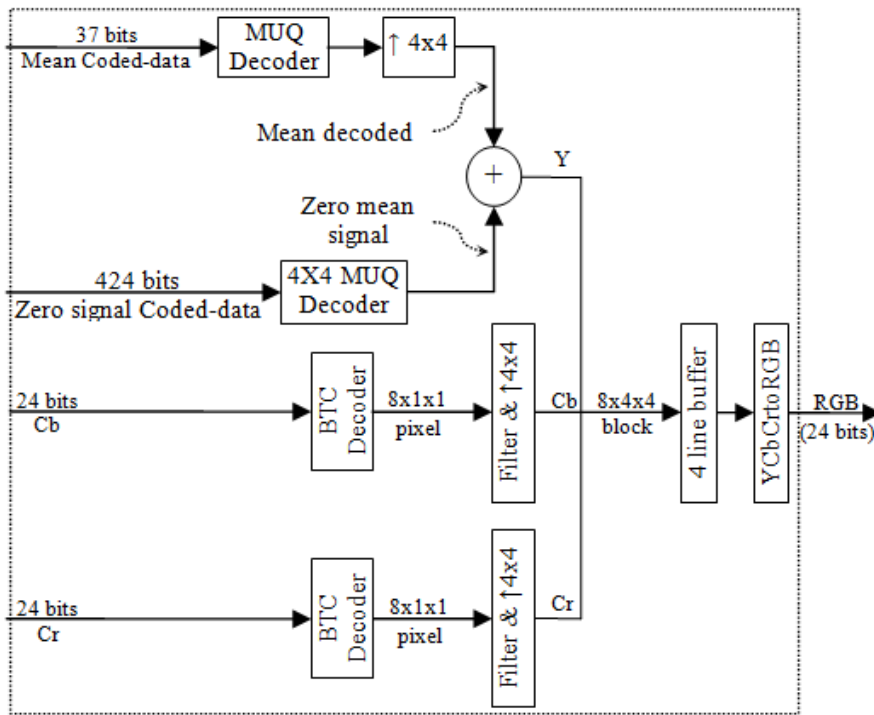
$$Code = Integer \left[\frac{ZMI}{QS} + 2^{b-1} \right], \quad (1)$$

$$0 \leq Code \leq 2^b - 1,$$

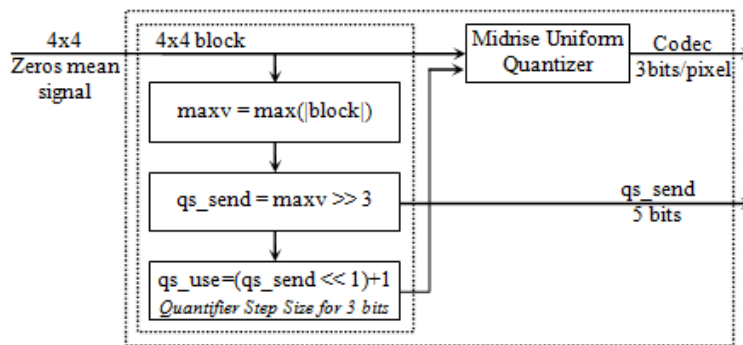
$$Decoded Output = (Code - 2^{b-1} + 0.5).QS. \quad (2)$$

3.3. Đánh giá độ phức tạp của MAIC so với AHIC

MAIC và AHIC chỉ khác nhau ở công đoạn xử lý thành phần độ chói Y. Mà cụ thể hơn là các bộ mã hóa và giải mã AQC (AQC encoder & decoder) được thay bằng các bộ mã hóa và giải mã MUQ (MUQ encoder & decoder). Về mặt số lượng các bộ tính toán thì MAIC nhiều hơn AHIC một bộ tính giá trị trung bình (mean) của 16 giá trị và một bộ giải mã AQC



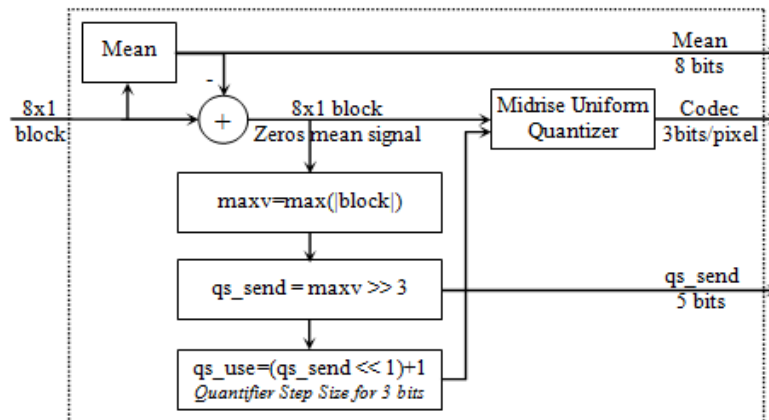
Hình 3.10. Sơ đồ bộ giải nén MAIC (MAIC Decoder).



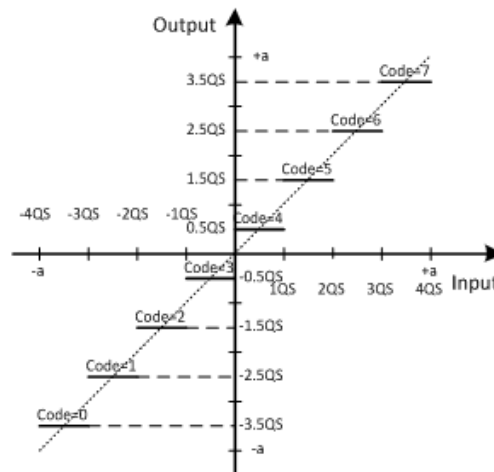
Hình 3.11. Sơ đồ bộ lượng tử hóa MUQ (MUQ Encoder) với đầu vào 4x4 giá trị có dạng *zeros mean signal*. Các ký hiệu » và « biểu diễn cho các phép toán dịch bit. Ở đây giá trị lượng tử hóa qs_use chính là giá trị $(maxv/4)$ được làm tròn về số nguyên ở cận trên của $(maxv/4)$.

nhằm giải mã 8 tín hiệu Min để tổng hợp với tín hiệu Y cho ra tín hiệu Y ở dạng "zeros mean signal". Đây là một trong những điểm mấu chốt giúp MAIC tránh được hiệu ứng lỗi kép gây nên hiệu ứng nhiễu khối trên ảnh giải nén.

Để có cái nhìn cụ thể hơn về độ phức tạp tính toán của AQC encoder & decoder so với MUQ encoder & decoder với giả thiết dữ liệu đầu vào ở dạng số nguyên có N phần tử (N=16 hoặc N=8), chúng ta xem xét bảng thống kê dưới đây:



Hình 3.12. Sơ đồ bộ lượng tử hóa MUQ với đầu vào 8 giá trị Mean.



Hình 3.13. Đồ thị của hàm lượng tử hóa MUQ với 8 mức lượng tử, các giá trị đầu vào có phân phối xác suất đều trong khoảng $(-a, a)$. Giá trị bước lượng tử hóa $QS = a/4$.

Qua bảng phân tích và so sánh trên đây về số lượng tính toán chúng ta có thể thấy khi tích hợp vào hệ thống mạch overdrive (overdrive circuit) thì hai giải pháp AQC và MUQ sẽ có độ phức tạp xấp xỉ nhau về các bộ tính toán (như bộ cộng/trừ, bộ nhân/chia và các bộ điều hướng). Xét về mặt thời gian thì độ trễ của tín hiệu đầu ra khi qua các bộ AQC và MUQ cũng xấp xỉ nhau. Tốc độ thực hiện của AHIC và MAIC phải luôn đảm bảo theo thời gian thực, vì thế nó sẽ phải hoạt động theo xung nhịp khác nhau trên các màn hình LCD có độ phân giải và tần xuất khung hình khác nhau.

4. KẾT QUẢ THỰC NGHIỆM

Chúng tôi đã tiến hành cài đặt cả hai mô hình AHIC và MAIC, rồi cho chạy chương trình kiểm định và so sánh trên cùng các chuỗi khung hình ảnh mẫu, đó là bộ ảnh chuyên dùng cho việc đánh giá thực nghiệm trong lĩnh vực xử lý ảnh và video được lấy từ nguồn các trang

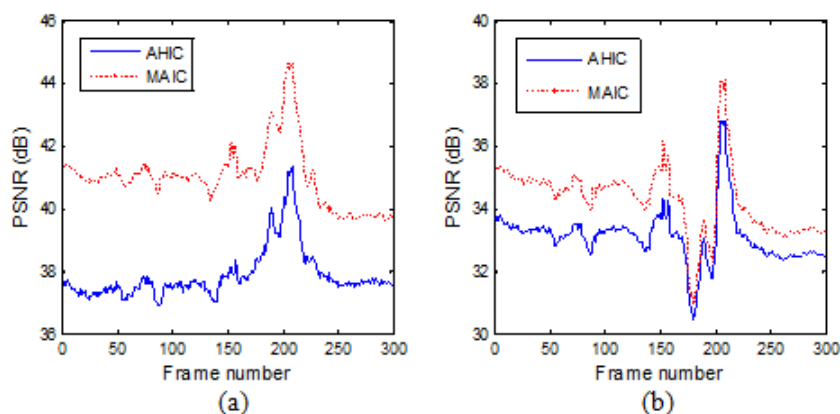
AQC encoder	MUQ encoder
<ul style="list-style-type: none"> • Phần khởi động: Để có được Min và Max cần + 1 phép tìm kiếm Min và Max với độ phức tạp tuyến tính theo N. + 1 phép tính trừ $\text{Diff} = \text{Max} - \text{Min}$ + Một phép tính chia và làm tròn kết quả về số nguyên: $\text{Qstep} = \text{Integer}(\text{Diff}/7.0+0.5)$ + Một phép kiểm tra và hiệu chỉnh Qstep trong miền giá trị 1..32 sau đó chuyển về 5 bit để chuyển đi như sau: Nếu $\text{Qstep} > 32$ thì gán $\text{Qstep} = 32$ Rồi tính $\text{Qstep_send} = \text{Qstep}-1;$ Và $\text{Qstep_use} = \text{Qstep_send} + 1$ • Phần mã hóa dữ liệu $\text{Code} = \text{Integer}[(\text{Input}-\text{Min})/\text{Qstep_use}]$ và Nếu $\text{Code} > 7$ thì gán $\text{Code} = 7$ <p>Sẽ gồm: + N phép trừ + N phép chia lấy phần nguyên + Một phép kiểm tra và hiệu chỉnh giá trị</p>	<ul style="list-style-type: none"> • Phần khởi động: Để có được giá trị maxv cần + 1 phép tìm kiếm Max của trị tuyệt đối các giá trị đầu vào với độ phức tạp tuyến tính theo N. + Một phép dịch sang trái 3 bit để thu được giá trị Qstep_send: $\text{Qstep_send} = \text{maxv} \gg 3$ + Một phép dịch bit và một phép cộng nhằm khôi phục giá trị lượng tử hóa sử dụng cho phần mã hóa: $\text{Qstep_use} = (\text{Qstep_send} \ll 1) + 1$ • Phần mã hóa dữ liệu $\text{Code} = \text{Integer}(\text{Input}/\text{Qstep_use}) + 4;$ và Nếu $\text{Code} > 7$ thì gán $\text{Code} = 7$ <p>Sẽ gồm: + N phép cộng + N phép chia lấy phần nguyên + Một phép kiểm tra và hiệu chỉnh giá trị</p>
AQC decoder	MUQ decoder
<ul style="list-style-type: none"> • Phần khởi động: + Tính Qstep_use từ Qstep_send bằng một phép cộng. $\text{Qstep_use} = \text{Qstep_send} + 1$ • Phần giải mã dữ liệu Decode = $\text{Integer}[(\text{Code} + 0.5) * \text{Qstep_use}]$ <p>Sẽ gồm: + N phép cộng + N phép nhân và lấy kết quả là phần nguyên</p>	<ul style="list-style-type: none"> • Phần khởi động: + Tính Qstep_use từ Qstep_send bằng phép dịch bit và phép cộng. $\text{Qstep_use} = (\text{Qstep_send} \ll 1) + 1$ • Phần giải mã dữ liệu Decode = $\text{Integer}[(\text{Code} - 4 + 0.5) * \text{Qstep_use}]$ <p>Sẽ gồm: + N phép cộng và trừ + N phép nhân và lấy kết quả là phần nguyên</p>

Web [11] và [12].

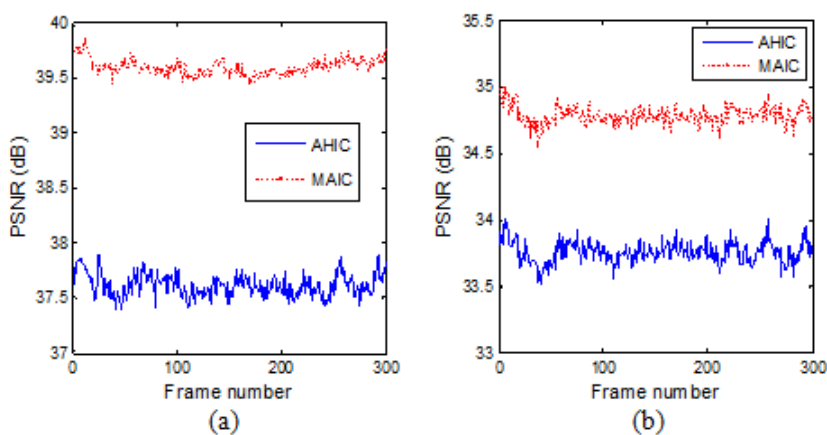
Hình 4.14 & 4.15 & 4.16 thể hiện sự so sánh hiệu năng của AHIC với MAIC. Trong hình 4.14 (a) cho thấy sử dụng MAIC thu được kết quả nén cao hơn AHIC vào khoảng 2dB đến 3.5dB khi chỉ đánh giá trên ảnh độ chói Y. Có được kết quả này là do chuỗi khung hình CIF Foreman có nhiều điểm nhiễu (noise). Hình 4.14 (b) cho thấy sử dụng MAIC thu được kết quả nén cao hơn AHIC vào khoảng 0.8dB đến 2dB khi đánh giá trên cả 3 thành phần màu RGB. Hình 4.15 (a) & (b) là sự đánh giá và so sánh tương tự giữa MAIC và AHIC nhưng trên chuỗi khung hình CIF Hall.

Hình 4.16 (a) & (b) thể hiện sự so sánh hiệu năng của AHIC với MAIC trên chuỗi khung hình CIF Bus. Qua đó cho thấy trong tình huống xấu nhất thì MAIC có thể cho chất lượng thấp hơn AHIC khoảng 0.7dB trên thành phần độ chói hay khoảng 0.25dB khi đánh giá trên cả ba thành phần màu RGB.

Hình 4.17 & 4.18 & 4.19 so sánh sự mất mát thông tin (hay error) giữa 2 mô hình nén AHIC và MAIC. Qua hình 4.17(b) cho thấy điểm yếu của AHIC, chỉ xét trên ảnh độ chói Y,



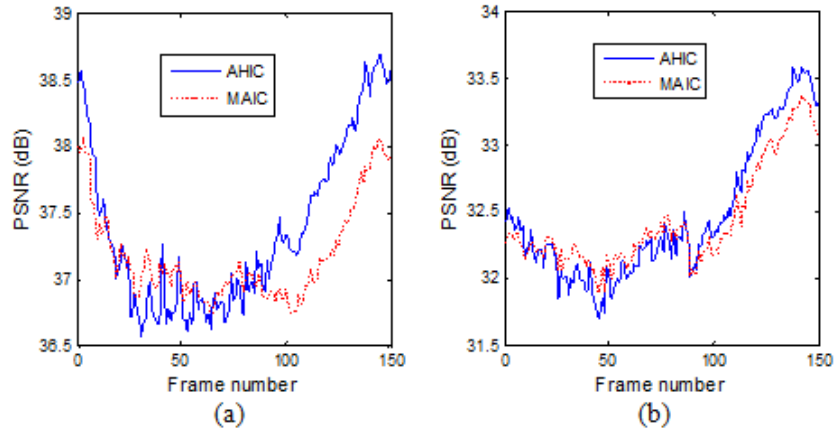
Hình 4.14. So sánh hiệu năng của AHIC với MAIC trên chuỗi hình CIF Foreman. (a) so sánh trên ảnh độ chói Y. (b) so sánh trên ảnh màu RGB.



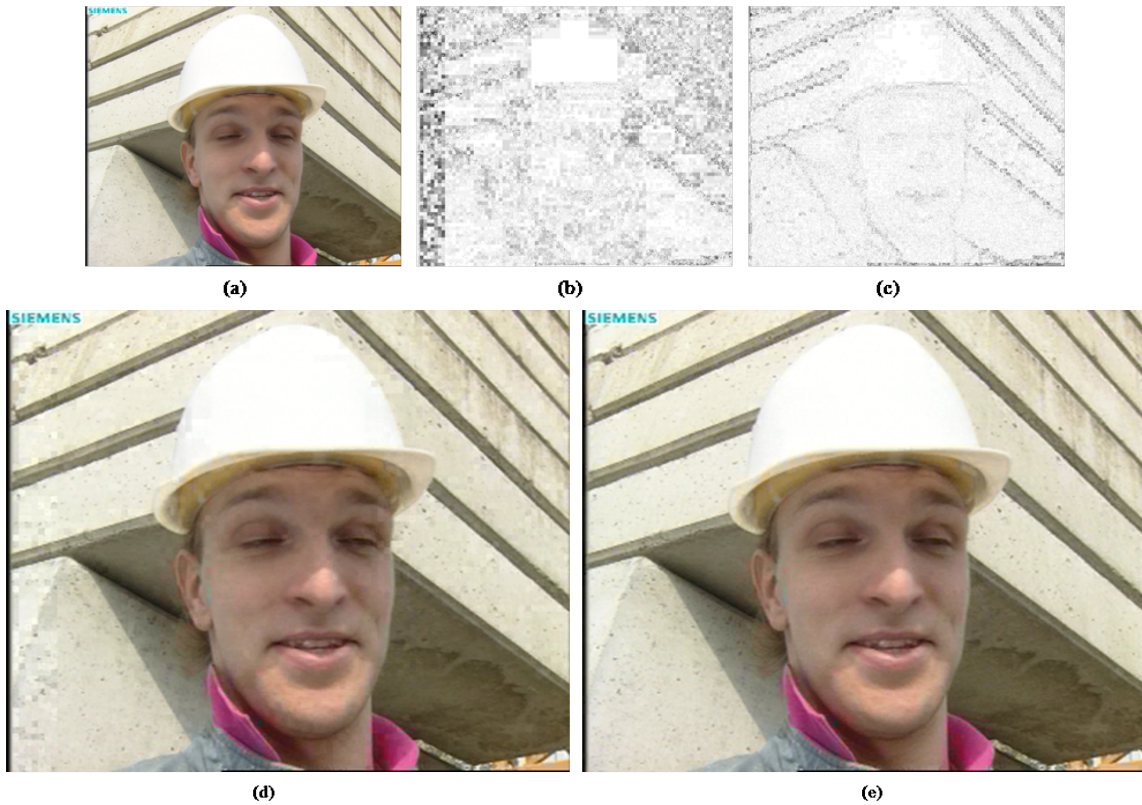
Hình 4.15. So sánh hiệu năng của AHIC với MAIC trên chuỗi hình CIF Hall. (a) so sánh trên ảnh độ chói Y. (b) so sánh trên ảnh màu RGB.

nó có nhiều error và bị hiệu ứng blocking. Hình 4.17(c) cho thấy error đã được giảm đáng kể và không còn bị hiệu ứng blocking, kết quả này thể hiện điểm mạnh của MAIC phù hợp với những nhận định trước đó. Hình 4.17 (d) & (e) lần lượt là ảnh giải nén của mô hình nén AHIC và MAIC khi xét trên cả ba thành phần màu RGB. Trên hình 4.17 (d) chúng ta có thể nhìn thấy rõ nhiều khối xuất hiện trên nhiều vùng của ảnh. Các khối nhiễu này tất yếu sẽ làm cho hệ thống overdrive điều khiển sai lệch và gây ra hiện tượng nhiễu khối hiển thị trên LCD. Đặc biệt khi đặt vào bối cảnh ảnh chuyển động qua một loạt các khung hình, thì sự xuất hiện lốm đốm của các khối nhiễu một cách ngẫu nhiên sẽ rất nhạy cảm với mắt người quan sát, gây cho người xem một cảm giác khó chịu và dễ dàng đưa ra đánh giá thấp về chất lượng của màn hình LCD. Hình 4.18 (b) & (c) & (d) & (e) là sự đánh giá và so sánh tương tự giữa MAIC và AHIC nhưng trên chuỗi khung hình CIF Hall.

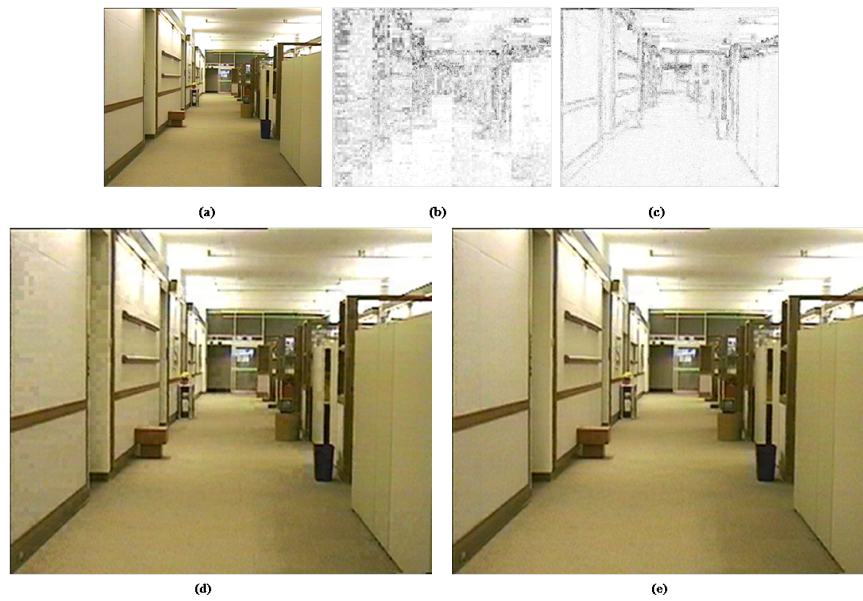
Hình 4.19 (b) & (c) & (d) & (e) là sự đánh giá và so sánh giữa MAIC và AHIC trên chuỗi khung hình CIF Bus. Đây là tình huống xấu khi chỉ số PSNR cho hình giải nén của MAIC thấp hơn của AHIC, song rõ ràng chỉ số PSNR nhằm đánh giá chất lượng ảnh giải nén không



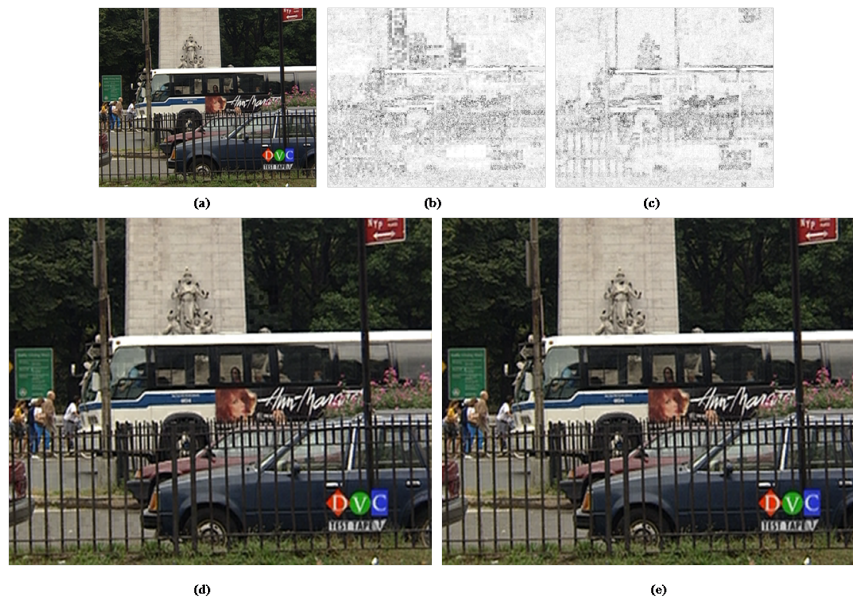
Hình 4.16. So sánh hiệu năng của AHIC với MAIC trên chuỗi hình CIF Bus. (a) so sánh trên ảnh độ chói Y. (b) so sánh trên ảnh màu RGB.



Hình 4.17. So sánh sự mất mát thông tin (hay error) giữa 2 mô hình nén AHIC và MAIC trên một khung hình của chuỗi khung hình CIF Foreman. (a) Ảnh gốc. (b) Ảnh Y-Error của AHIC. (c) Ảnh Y-Error của MAIC. (d) Ảnh giải nén của AHIC. (e) Ảnh giải nén của MAIC. Ở đây ảnh Y-Error ở (b) & (c) là sai số của của thành phần độ chói được lấy trị tuyệt đối và khuếch đại lên 12 lần sau đó biến đổi âm bản. Các bức ảnh (a) & (b) & (c) được thu nhỏ kích thước với tỷ lệ 1/2 theo cả 2 chiều.



Hình 4.18. So sánh sự mất mát thông tin (hay error) giữa 2 mô hình nén AHIC và MAIC trên một khung hình của chuỗi khung hình CIF Hall. (a) Ảnh gốc. (b) Ảnh Y-Error của AHIC. (c) Ảnh Y-Error của MAIC. (d) Ảnh giải nén của AHIC. (e) Ảnh giải nén của MAIC. Ở đây ảnh Y-Error ở (b) & (c) là sai số của của thành phần độ chói được lấy trị tuyệt đối và khuếch đại lên 12 lần sau đó biến đổi âm bản. Các bức ảnh (a) & (b) & (c) được thu nhỏ kích thước với tỷ lệ $1/2$ theo cả 2 chiều.



Hình 4.19. So sánh sự mất mát thông tin (hay error) giữa 2 mô hình nén AHIC và MAIC trên một khung hình của chuỗi khung hình CIF Bus. (a) Ảnh gốc. (b) Ảnh Y-Error của AHIC. (c) Ảnh Y-Error của MAIC. (d) Ảnh giải nén của AHIC. (e) Ảnh giải nén của MAIC. Ở đây ảnh Y-Error ở (b) & (c) là sai số của của thành phần độ chói được lấy trị tuyệt đối và khuếch đại lên 12 lần sau đó biến đổi âm bản. Các bức ảnh (a) & (b) & (c) được thu nhỏ kích thước với tỷ lệ $1/2$ theo cả 2 chiều.

đánh giá được tất cả các khía cạnh chất lượng mà chúng ta mong muốn. Hình 4.19 (c) cho thấy ảnh giải nén của MAIC bị lỗi khá nặng trên các chi tiết đường nét (egde) của ảnh so với AHIC, song trên hình 4.19 (e) là ảnh giải nén của MAIC chúng ta rất khó nhận ra sự khác biệt này so với ảnh gốc là hình 4.19 (a), do tính chất kém nhạy cảm với các chi tiết tần số cao của mắt người, kể cả khi đặt vào bối cảnh chuyển động trên các khung hình của CIF Bus. Điều ngược lại, đó là hiện tượng nhiễu khối xuất hiện rất nhiều trên hình 4.19 (b) và những vùng bị nhiễu khối nặng có thể dễ dàng nhận thấy trên ảnh 4.19 (d) là ảnh giải nén của AHIC. Điều này là minh chứng cho khả năng cải thiện chất lượng ảnh giải nén, cũng là cải thiện chất lượng xử lý của hệ thống overdrive hay nói xa hơn là cải thiện chất lượng hiển thị hình ảnh của LCD.

Bảng 1. Trung bình cộng kết quả chất lượng ảnh nén cho mỗi mô hình - đánh giá theo PSNR (dB)

Chuỗi ảnh (CIF format)	Ảnh số	Ảnh Y		Ảnh RGB	
		AHIC	MAIC	AHIC	MAIC
3inrow	360	39.01	41.5	35.77	37.14
BetesPasBetes	360	40.26	41.72	29.88	30.04
Bus	150	37.37	37.18	32.46	32.43
Cheerleaders	360	35.88	37.44	23.75	23.87
Coastguard	300	38.67	39.24	35.4	35.78
Container	300	38.29	38.96	33.6	33.96
FlowerGarden	360	35.42	35.8	25.34	25.42
Football	360	39.37	40.95	30.87	31.12
Foreman	300	37.88	41.03	33.1	34.28
Hall	300	37.62	39.59	33.76	34.78
LePoint	360	38.58	38.37	26.02	26
MobileCalendar	360	34.41	34.9	23.79	23.84
Mother-daughter	300	42.32	44.05	37.39	38
News	300	38.49	39.48	32.16	32.49
Paris	1065	35.85	36.57	27.67	27.81
Stefan	90	36.86	37.23	29.94	30.06
Vtc1nw	360	39.25	43.54	35.52	37.34
Vtc2zm	360	39.49	41.04	30.75	30.99
Washdc	360	36.92	38.88	31.37	32.02
Waterfall	260	39.6	40.82	28.19	28.28

BẢNG 1 thể hiện trung bình cộng kết quả chất lượng ảnh nén với các chuỗi khung hình khác nhau. Từ đó rút ra được thông tin đánh giá mức độ cải thiện chất lượng ảnh nén của MAIC so với AHIC trong BẢNG 2. Qua số liệu trong BẢNG 2 chúng ta thấy được khả năng cải thiện chất lượng của MAIC so với AHIC. Tình huống tốt nhất MAIC cho kết quả tốt hơn AHIC đến 4.28dB xét trên ảnh Y hay 1.83dB khi xét trên ảnh RGB. Trong tình huống xấu nhất thì MAIC cho chất lượng thấp hơn AHIC 0.21dB xét trên ảnh Y. Lấy trung bình qua các kết quả thực nghiệm, MAIC cho chất lượng cao hơn AHIC 1.34dB nếu chỉ xét trên thành phần ảnh Y, hay 0.45dB nếu xét trên ảnh màu RGB.

Bảng 2. Đánh giá mức độ cải thiện chất lượng ảnh nén của MAIC so với AHIC - đánh giá theo PSNR (dB)

Chuỗi ảnh (CIF format)	Số ảnh	Kết quả cải thiện xét trên ảnh Y	Kết quả cải thiện xét trên ảnh RGB
3inrow	360	2.49	1.37
BetesPasBetes	360	1.45	0.16
Bus	150	-0.19	-0.03
Cheerleaders	360	1.56	0.12
Coastguard	300	0.57	0.37
Container	300	0.67	0.36
FlowerGarden	360	0.38	0.08
Football	360	1.58	0.25
Foreman	300	3.15	1.18
Hall	300	1.98	1.03
LePoint	360	-0.21	-0.02
MobileCalendar	360	0.49	0.05
Mother-daughter	300	1.73	0.60
News	300	0.99	0.33
Paris	1065	0.72	0.14
Stefan	90	0.37	0.11
Vtc1nw	360	4.28	1.83
Vtc2zm	360	1.56	0.24
Washdc	360	1.95	0.65
Waterfall	260	1.22	0.09
Min		-0.21	-0.03
Max		4.28	1.83
Average		1.34	0.45

5. KẾT LUẬN

Qua bài báo này, chúng tôi đề xuất một mô hình nén mới có tên gọi là Mean Adaptive Image Coding (MAIC) kế thừa từ mô hình AHIC. Mô hình MAIC mới đề xuất cho phép tránh được hiện tượng nhiễu khối, là loại nhiễu mà mô hình AHIC mắc phải qua đó gián tiếp làm suy giảm chất lượng hình ảnh hiển thị trên LCD. Qua thực nghiệm trong các trường hợp, các kết quả thử nghiệm đều đã chứng tỏ rằng MAIC có khả năng tránh nhiễu khối và cải thiện chất lượng ảnh nén một cách đáng kể so với mô hình AHIC trên cùng một tỷ số nén và với một độ phức tạp tính toán có thể nói là tương đương (khác biệt không đáng kể).

Một hướng phát triển nữa của chúng tôi nhằm nghiên cứu tích hợp thêm khả năng cải thiện chất lượng nén ở các chi tiết đường nét nhằm cải thiện hơn nữa chất lượng ảnh nén và tạo chất lượng đồng đều trên mọi vùng ảnh đã được trình bày trong [9]. Các nghiên cứu về nén ảnh cho LCD đều cần lưu ý tính khả thi có thể áp dụng vào trong hệ thống LCD để giảm giá thành của sản phẩm.

TÀI LIỆU THAM KHẢO

- [1] R. H. M. Wubben and G. J. Hekstra, LCD overdrive frame memory reduction using scalable DCT-based compression, *SID Symposium Digest of Technical Papers* **35** 1 (May 2004) 1348–1351.
- [2] J. K. Sung, C. G. Kim, J. K. An, M. H. Park, S. D. Yeo, A new method for improvement of response time by data compression using color space conversion, *SID Symposium Digest of Technical Papers*, **36** 1 (May 2005) 474–477.
- [3] I. J. Chun, H. Mun, J. H. Sung, S. Y. Park, B. G. Kim, Overdrive frame memory reduction using a fast discrete wavelet transform, *Proc. of 21st Int. Technical Conf. on Circuits/Systems, Computer and Communications (ITC-CSCC'06)*, Chiang Mai, Thailand (July 2006) 161–164.
- [4] J. Wang, K. Y. Min, J. W. Chong, A hybrid image coding in overdrive for motion blur reduction in LCD, *Proc. of 6th Int. Computer Entertainment Computing (ICEC'07)*, Shanghai, China (Sep. 2007) 263–270.
- [5] J. W. Han, M.C. Hwang, S.J. Ko, Vector quantizer based block truncation coding for color image compression in LCD overdrive, *IEEE Transactions on Consumer Electronics* **54** 4 (Nov. 2008) 1839–1845.
- [6] J. Wang, J. W. Chong, High Performance Overdrive Using Improved Motion Adaptive Codec in LCD, *IEEE Transactions on Consumer Electronics* **55** 1 (FEBRUARY 2009) 20–26.
- [7] L. Y. Liao, C. W. Chen, Y. P. Huang, Local Blinking HDR LCD Systems for Fast MPRT With High Brightness LCDs, *Journal of Display Technology* **6** 5 (May 2010).
- [8] C.T. Le Dinh, University of Sherbrooke, private communication, 2010.
- [9] H.T. Nguyen et al., Content-based Bit Exchange Techniques for Frame Delay Memory Application, *accepted for presentation at the 6th IEEE Conference on Industrial Electronics and Applications (ICIEA 2011)*, Beijing, China, June 21 - 23, 2011.
- [10] <http://www.cs.waseda.ac.jp/gcoe/jpn/publication/symposium/img/090808JWChong.pdf>
- [11] <http://trace.eas.asu.edu/yuv/index.html>
- [12] <ftp://vqeg.its.bldrdoc.gov/MM/cif/>

Ngày nhận bài 09 - 5 - 2012

Nhận lại sau sửa ngày 19 - 12 - 2012