BIT-DEPTH EXPANSION USING MINIMUM RISK BASED CLASSIFICATION

*Gaurav Mittal, †Vinit Jakhetiya, †Sunil Prasad Jaiswal, †Oscar C. Au, *†Anil Kumar Tiwari, †Dai Wei

* International Institute of Information Technology, Hyderabad, India
† The Hong Kong University of Science and Technology, Hong Kong
*† Indian Institute of Technology, Rajasthan, India
Email: *gaurav.mittal@research.iiit.ac.in, †vjakhetiya@ust.hk,
†spjaiswal@ust.hk, †eeau@ust.hk, *†akt@iitj.ac.in, †weidai@ust.hk

ABSTRACT

Bit-depth expansion is an art of converting low bit-depth image into high bit-depth image. Bit-depth of an image represents the number of bits required to represent an intensity value of the image. Bit-depth expansion is an important field since it directly affects the display quality. In this paper, we propose a novel method for bit-depth expansion which uses Minimum Risk Based Classification to create high bit-depth image. Blurring and other annoying artifacts are lowered in this method. Our method gives better objective (PSNR) and superior visual quality as compared to recently developed bit-depth expansion algorithms.

Index Terms— Bit-Depth expansion, Minimum risk based classification, Prediction, Posterior probability, Risk calculation.

1. INTRODUCTION

Bit-depth is the number of bits used to represent the intensity value of a pixel. Bit-depth expansion is an important phenomenon when a low bit depth-image is displayed on a high bit-depth monitor/projector. The existing methods for bit-depth extension include zero-padding (ZP), multiplication by an ideal gain (MIG) [2], Bit-replica (BR) [2], Inverse halftoning [1] and high dynamic range imaging. Inverse halftoning and high dynamic range imaging requires a collection of low bit-depth images in order to produce high bit-depth Image, but in real time scenarios we have only one low-bit depth image (i.e in high efficiency video coding (HEVC) uses bit-depth expansion). Hence, in this research work our focus is only on those methods which requires only one low bit-depth image in order to produce a high bit-depth image (i.e ZP, MIG and BR methods).

In conventional zero padding [3] method, given L bit-depth image is converted into H bit depth image by appending $(H - L)$ number of zeros as LSB's. In MIG a multiplication term $(M = \frac{2^H-1}{2^L-1})$ is obtained and intensity values of pixels in L bit-depth image is multiplied by M to get H bit-depth image. Bit-replica (BR) [2] appends n MSB's of low bit-depth image at LSB position in newly created high bit-depth image. Here n refers to the bit-depth difference between high bit-depth intensity value and low bit-depth intensity value. All of these methods blindly produce higher bit-depth image by either multiplying low-bit depth image by fixed number or repeating a given pattern, without using the characteristics of neighboring pixels. All these methods map each low depth value to a particular high depth value irrespective of the neighbor's behavior. Hence these non-adaptive methods tend to give the blurring and other annoying artifacts.

The main contribution of this paper is to propose an efficient algorithm which generates high bit-depth intensity value using the low bit-depth intensity value depending upon the minimum risk based classification. Our method chooses the particular high bit-depth value out of all possible values which have associated minimum risk. Our proposed algorithm gives better objective and subjective quality as compared to recently developed bit-depth expansion algorithms.

The rest of the paper is organized as follows. Section 2 gives the quick review of minimum risk classification. Proposed method is presented in section 3. Section 4 proposes a modified algorithm for lowering down the computational complexity. Experimental results for standard test Images are presented in section 5. Final conclusion of this work is given in section 6.

2. OVERVIEW OF MINIMUM RISK CLASSIFIER

Suppose we want to convert an image of L bit-depth(I_L) into another image of $H(H{>}L)$ bit-depth(I_H). We can copy all L bits from low-bit depth image and consider these bits as MSB's of the higher bit-depth (H bit) image. Now the problem is to fill up last $(H - L)$ bits efficiently and effectively. The possible ways to fill up the last $(H - L)$ bits are $2^{(H-L)}$. We have to classify (map) every low bit-depth value into one of the high bit-depth value (out of 2^H). Minimum risk based classifier first finds risk associated with each high bit-depth category($2^{(H-L)}$) and then classify the input low bit-depth

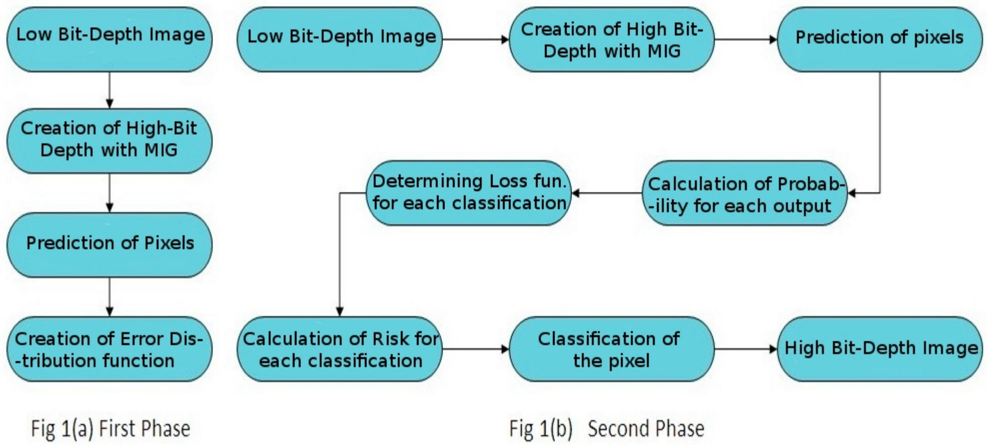

Fig. 1. Overview of Bit-depth Expansion system

value in the category which have associated minimum risk value. So minimum risk classifier tries to lower down total expected error. Suppose there are total $2^{(H-L)}$ possible output categories $[O_1, O_2, ...O_{2^{(H-L)}}]$, then the risk associated in classifying the pixel in j^{th} category can be found as:

$$R_j = \sum_{i=1}^{2^{(H-L)}} P(O_i) \times \lambda_{ji} \tag{1}$$

Where λ_{ji} is the associated loss when a i^{th} category pixel is classified into j^{th} category. $P(O_i)$ represents the posterior probability corresponding to class i. In order to find all the risk values (R_1 to $R_{2^{(H-L)}}$) we need:

1. Posterior probability of each category i.e. $(P(O_1)$ to $P(O_{2^{(H-L)}}))$.
2. Risk associated with each classification i.e. $\lambda_{ij}, i \in (1, 2^{(H-L)})$ $j \in (1, 2^{(H-L)})$.

Calculation of posterior probabilities $[P(O_1)$ to $P(O_{2^{(H-L)}})]$ and loss function is explained in following section.

3. PROPOSED MINIMUM RISK BASED BIT-DEPTH EXPANSION METHOD

The proposed algorithm can be divided into two phases. First phase creates prediction error histogram and second phase finds risks associated with each category and classify the low bit-depth pixel into high bit-depth pixel, thus creating high bit-depth image.

3.1. First phase

The first phase can be divided in the following 3 steps as shown in Fig.1 (a)

3.1.1. *Creating the High Bit-Depth image using MIG*

Our method first converts low bit-depth image into high bit-depth image using MIG [2].

3.1.2. *Prediction of pixels*

We predict each pixel location in high bit-depth image by taking average of it's 4-connected neighbors and create the error image (E). Suppose that a given point (i, j), if the pixel value in the high bit-depth image(created using MIG method) is $I_{mig}(i, j)$ and predicted value is $P(i, j)$, then:-

$$P(i,j) = \begin{array}{l}[I_{mig}(i,j-1) + I_{mig}(i-1,j) + \\ I_{mig}(i,j+1) + I_{mig}(i+1,j)]/4\end{array}$$

$$E(i,j) = I_{mig}(i,j) - P(i,j) \tag{2}$$

3.1.3. *Creating the Error distribution function*

Error distribution function (EDF) gives the distribution of Error samples in the whole error image. It generally peaks at zero value and its amplitude decreases as we move away from the center. If the value of Error distribution function at a point 'X' is N then it means that there are total N pixels in the input high bit-depth image such that

$$X = I_{mig}(i,j) - P(i,j) \tag{3}$$

This error distribution is now normalized by dividing each value of histogram by total number of pixels in the error image. A plot of error histogram for Lena image is shown in Fig.2

3.2. Second Phase

In second phase we first create high bit-depth image using MIG [2] and then predict the pixel by taking average of its

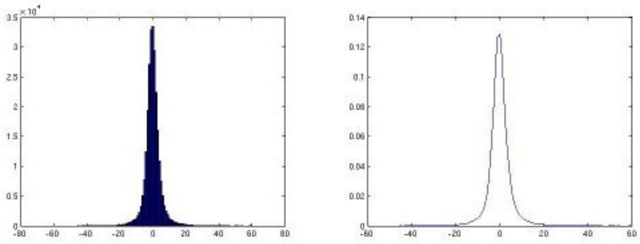

Fig. 2(a) Fig. 2(b)

Fig. 2. (a). Error distribution function (EDF) for Lena image (b). Normalized Error Distribution

4-connected neighbor, as we did in first phase. The second phase can be divided in the following 4 steps as shown in Fig.1 (b) and they are as follows:-

3.2.1. Calculating the posterior probability of each possible output

For a given point (i, j), difference between predicted value $(P(i, j))$ and each possible high bit-depth value$[(I_L(i, j) \times 2^{(H-L)})$ to $(I_L(i, j) \times 2^{(H-L)} + 2^{(H-L)} - 1)]$ is calculated:

$$\begin{aligned} Diff_1 &= P(i,j) - (I_L \times 2^{(H-L)}) \\ Diff_2 &= P(i,j) - (I_L \times 2^{(H-L)} + 1) \\ &. \\ &. \\ Diff_{2^{(H-L)}} &= P(i,j) - (I_L \times 2^{(H-L)} + 2^{(H-L)} - 1) \end{aligned} \tag{4}$$

Here I_L is low bit-depth image, H and L are bit-depths of high bit-depth and low bit-depth image respectively. Now if we donate error distribution function by EDF, then posterior probabilities are given by:

$$\begin{aligned} P(O_1) &= EDF(Diff_1) \\ P(O_2) &= EDF(Diff_2) \\ &. \\ &. \\ P(O_{2^{(H-L)}}) &= EDF(Diff_{2^{(H-L)}}) \end{aligned} \tag{5}$$

For example suppose if the predicted pixel$(P(i, j))$ is equal to $(I_L \times 2^{(H-L)} + k - 1)$, then $Diff_k$ will be zero and the posterior probability for that output category$(P(O_k))$ will be maximum since it will be given by $EDF(0)$. Similarly posterior probabilities for categories $[O_1$ - $O_{k-1}]$ will be $[EDF(-k + 1)$ - $EDF(-1)]$ and posterior probabilities for categories $[O_{k+1}$ - $O_n]$ will be given by $[EDF(1)$ - $EDF(n - k)]$.

After finding probabilities $[P(O_1)$ to $P(O_{2^{(H-L)}})]$, we will have to normalize it. The normalized values are given as-

$$P(O_j) = \frac{P(O_j)}{\sum_{k=1}^{2^{(H-L)}} P(O_k)} \tag{6}$$

Table 1. Results for various bit depth expansion, Here Exp. refers to Expansion and X refers to ABDE-P1 [3]

Images	Exp.	ZP	MIG	BR	X	Prop.
Lena	$6 \rightarrow 8$	42.69	45.88	44.99	47.07	**47.35**
Baboon	$6 \rightarrow 8$	42.68	46.14	45.58	46.52	**46.66**
Barbara	$6 \rightarrow 8$	42.68	45.44	44.71	47.01	**47.07**
Peppers	$6 \rightarrow 8$	42.70	45.70	45.11	46.52	**46.98**
Sailboat	$6 \rightarrow 8$	42.70	45.27	44.16	**46.93**	46.90
Airplane	$6 \rightarrow 8$	42.69	45.09	43.64	47.52	**47.62**
Goldhill	$6 \rightarrow 8$	42.67	45.75	45.02	46.97	**47.02**
Boats	$6 \rightarrow 8$	42.64	45.49	45.17	**47.31**	46.74
Average	$6 \rightarrow 8$	42.68	45.60	44.80	46.98	**47.04**
Lena	$5 \rightarrow 8$	35.72	39.30	39.16	-	**41.96**
Baboon	$5 \rightarrow 8$	36.69	39.61	39.44	-	**41.05**
Barbara	$5 \rightarrow 8$	35.69	39.04	38.76	-	**41.49**
Peppers	$5 \rightarrow 8$	35.71	39.03	38.82	-	**41.51**
Sailboat	$5 \rightarrow 8$	35.70	38.54	38.16	-	**41.20**
Airplane	$5 \rightarrow 8$	35.74	38.36	38.17	-	**42.07**
Goldhill	$5 \rightarrow 8$	35.69	39.20	38.93	-	**41.43**
Boats	$5 \rightarrow 8$	35.59	39.44	39.26	-	**41.40**
Average	$5 \rightarrow 8$	35.69	39.07	38.83	-	**41.51**
Lena	$4 \rightarrow 6$	30.50	33.07	32.57	35.75	**35.93**
Baboon	$4 \rightarrow 6$	30.55	33.73	33.16	34.91	**35.30**
Barbara	$4 \rightarrow 6$	30.48	32.82	32.38	35.45	**35.47**
Peppers	$4 \rightarrow 6$	30.51	33.26	32.69	**35.89**	35.84
Sailboat	$4 \rightarrow 6$	30.57	32.61	31.82	35.47	**35.47**
Airplane	$4 \rightarrow 6$	30.40	32.16	31.43	**35.59**	35.13
Goldhill	$4 \rightarrow 6$	30.61	33.07	32.53	35.64	**35.77**
Boats	$4 \rightarrow 6$	30.30	33.37	33.09	35.66	**35.95**
Average	$4 \rightarrow 6$	30.49	33.01	32.46	35.54	**35.60**

3.2.2. Calculating loss function with each classification

Loss function gives the loss associated with each classification. Here pixel with any original value $[(I_L(i, j) \times 2^{(H-L)})$ to $(I_L(i, j) \times 2^{(H-L)} + 2^{(H-L)} - 1)]$ can be classified into any other category, so loss function have $2^{2\times(H-L)}$ values here. λ_{ji} gives the loss in classifying pixel values with category i into category j. Our aim here is to maximize the PSNR of newly created image with respect to original image. The PSNR is inversely proportional to square error. The square error, when pixel with j^{th} category classified into i^{th} category is,

$$Sq_{error} = (i - j)^2 \tag{7}$$

Table 2. Expanding method used to reduce complexity

Depth Diff.	Method used
Even	$L \rightarrow L+2$ & $L+2 \rightarrow l+4$ & ... ...$H-2 \rightarrow H$
Odd	$L \rightarrow L+2$ & $L+2 \rightarrow l+4$ & ... ..$H-3 \rightarrow H-1$ & $H-1 \rightarrow H$

so the risk value for the same classification

$$\lambda_{ij} = (i-j)^2 \tag{8}$$

that means if a value is classified in original category (i=j) the loss value is zero, Otherwise it is proportional to the square of the difference.

3.2.3. *Calculating the risk with each possible output*

After getting all posterior probabilities and loss functions, we can calculate the risk associated in classifying the pixel in each category.

Risk value associated in classifying the pixel in j^{th} category can be given as

$$R(O_j) = \sum_{i=1}^{2^{(H-L)}} P(O_i)\lambda_{ji} \tag{9}$$

3.2.4. *Classifying the pixel*

After finding risk values with each possible high bit-depth output category, we classify the pixel into the category with minimum risk. Thus, high bit-depth image(I_{out}) is created using minimum risk based classification.

4. MODIFIED PROPOSED ALGORITHM

Proposed algorithm needs to find risk associated with each of $2^{(H-L)}$ category, which increases exponentially when the bit-depth difference i.e. $(H-L)$ increases. In order to reduce the time complexity, we propose that instead of expanding directly from L to H, expansion should be done in the following manner

$$L \text{ to } L+i_1 \text{ \& } L+i_1 \text{ to } L+i_2 \text{ \& } ...L+i_n \text{ to } H \tag{10}$$

Where i_1, i_2, ...,i_n are the points chosen by algorithm. Here the complexity from L to H, which is $2^{(H-L)}$ reduces to $2^{i_1} + 2^{i_2-i_1} + ... + 2^{H-(L+i_n)}$. The expanding method that our algorithm follows is given in Table 2.

The results for modified proposed algorithm is presented in table 3. It can be noted that the time required to expand using modified algorithm is lower then the original method and the complexity of modified algorithm is linearly proportional to the bit-depth difference.

Table 3. Results for modified proposed algorithm, here Exp. refers to expansion, Prop. refers to Proposed and M.P. refers to Modified Proposed. Time shown is calculated on a computer with 2 GB ram, core 2 duo 1.80 GHz processor, Ubuntu 12.04 and Matlab R2012a platform

Images	Exp.	PSNR Prop.	PSNR M.P.	Time Prop.	Time M.P.
Lena	$3 \rightarrow 8$	28.96	27.76	47.09	23.84
Baboon	$3 \rightarrow 8$	29.48	29.38	47.11	24.14
Barbara	$3 \rightarrow 8$	28.60	29.04	47.46	23.90
Lena	$4 \rightarrow 8$	35.90	35.82	24.07	17.55
Baboon	$4 \rightarrow 8$	35.17	35.11	24.08	17.32
Barbara	$4 \rightarrow 8$	35.29	35.39	24.89	17.87

5. SIMULATION RESULTS

We tested our algorithm on various standard images with various bit-depth expansions. We created low bit-depth image by dividing high bit-depth value by $2^{(H-L)}$ and taking floor.

$$I_L = floor(I_H/2^{(H-L)}) \tag{11}$$

We tested algorithm on standard images with 512 $\times$ 512 dimension and performed mainly 3-sets of bit-depth expansion and those are 4-bit to 6-bit, 6-bit to 8-bit, 5-bit to 8-bit. Result of 6-bit depth to 8-bit depth expansion for the shoulder part of standard image Lena using different methods including ABDE-P1 [3] is presented in Fig. 3.

Peak Signal to Noise Ratio (PSNR):- PSNR is a measure of quality which is used to compare the performance of different algorithms. PSNR for 8 bit-depth is given by:

$$PSNR = 10 \times (log_{10}\frac{255^2}{MSE}) \tag{12}$$

where MSE(Mean Square Error) is given by:

$$MSE = SE/(512 * 512) \tag{13}$$

and finally SE(Square error) is given by:

$$SE = \sum_{i=1}^{m}\sum_{j=1}^{n}(I(i,j) - I_{out}(i,j))^2 \tag{14}$$

Where m,n are the number of rows and column respectively, I is the original image and I_{out} is the created high bit-depth image(using I_L). We computed the PSNR between original high bit-depth image and bit-depth expanded image in order to compare performance. The PSNR plots for all 3-sets of expansions for standard images are given in Table 1.

The results can be further improved if we do frequency based filtering [3] in the resultant image in order to remove contours.

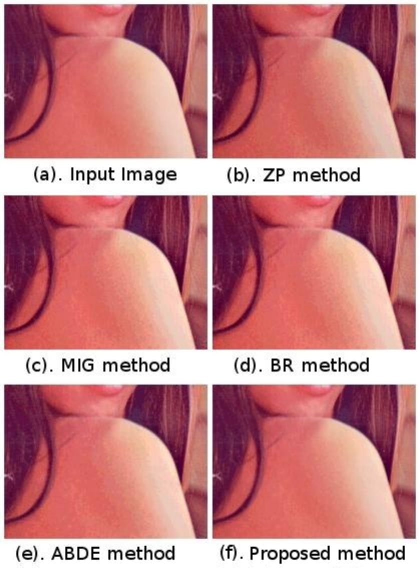

Fig. 3. Comparison of different Bit-Depth expansion algorithm based on visual quality. (a) is original high bit-depth image and (b),(c),(d),(e),(f) are created high bit-depth image using low bit-depth image.

6. CONCLUSIONS

In this work, we propose to obtain high bit-depth image from a low bit-depth image by using some characteristics of the neighboring pixels. Our method utilizes knowledge of minimum risk classification in order to perform bit-depth expansion. The proposed algorithm has been tested on various images and it always gives satisfactory results. Proposed algorithm can be used for any bit-depth to any bit-depth expansion. Furthermore a modification in the original algorithm is proposed to reduce time-complexity. Results for proposed and modified proposed algorithms are also reported.

7. REFERENCES

[1] M.S. Fu and Oscar C. Au, “ Hybrid Inverse Halftoning using Adaptive Filtering,” in Proc. of IEEE Int. Sym. on Circuits and Systems, May 1999.

[2] Robert Ulichney and Shiufun Cheung, “Bit-Depth Expansion by Bit Replication. Color Imaging:” Device Independent Color (Proceedings of SPIE Volume 3300) San Jose, CA, p. 232-241, 1998.

[3] Chun Hung LIU, Oscar C. AU, P. H. W. WONG, and M. C. KUNG ‘Bit-Depth Expansion By Adaptive Filter ,” in . of IEEE Int. Sym. on Circuits and Systems, May 2008.

[4] Cheuk-Hong CHENG, Oscar C. AU, Chun-Hung LIU, Ka-Yue YIP, ‘Bit-depth expansion by contour region reconstruction,” in . of IEEE Int. Sym. on Circuits and Systems, May 2009

[5] M.S. Fu, Oscar C. Au, “Fast Adaptive Spatial Varying Filtering for Inverse Halftoning,” in . Of SPIE Conf. On Visual Communication and Image Processing, Jan 2001.

[6] Z. Xiong, M. T. Orchard and K. Ramchandran, ‘Wavelet-Based Approach to Inverse Halftoning,” in *”Proc. of IS&T/SPIE Symposium on Electronic Imaging Science and Technology, 1997*.

[7] R. L. Stevenson ‘Inverse Halftoning via MAP Estimation,” in Transactions on Image Processing, vol. 5, no. 4, pp. 574-583, April 1997.