Error Minimization of 2-D DCT and Quantization Operations for a Gray Scale Images JPEG Compressor

Bruno Silveira Neves, Luciano Volcan Agostini

Group of Architectures and Integrated Circuits
Universidade Federal de Pelotas, DMEC, Pelotas, Rio Grande do Sul, Brazil
Campus Universitário, S/Nº – Caixa Postal 354 – CEP. 96010-900

Abstract

This paper presents the analysis designed to increase the quality of the data generated by a JPEG compressor designed in hardware. The minimization of the errors generated by 2-D DCT (Two Dimensional Discrete Cosine Transform) and quantization operations were the main focus of this paper. A generic description of the 2-D DCT was designed to validate the algorithm used into the hardware description. To detect the error generated by the 2-D DCT and quantization designed in hardware were made two descriptions in C for each operation, one generating ideal values and other generating real values, considering hardware restrictions. As expected, the main error generators were the multipliers used into the 2-D DCT and quantization operations. The suggested modifications increase de quality of the compressed image in a rate higher than 80%. The proposed modifications generates an estimated increment into the resources usage of 26% and 180% into de 2-D DCT and quantization architectures. The estimated increase of resources usage into the JPEG compressor was of 28%.

1. Introduction

The JPEG compressor designed in hardware [1], focus of this paper, uses the JPEG baseline compression mode[2]. The baseline compression mode can be divided in three main steps, as is showed in fig. 1: 2-D DCT (Two Dimensional Discrete Cosine Transform), quantization and entropy coding.

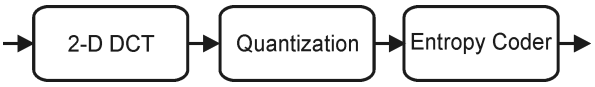

Figure 1 – JPEG baseline compression

The JPEG compression principle is the use of controllable losses to reach high compression rates. In this context, the information is transformed to the frequency domain through 2-D DCT. Since neighbor pixels in an image have high likelihood of showing small variations in color, the DCT output will group the higher amplitudes in the lower spatial frequencies [3]. Then, the higher spatial frequencies can be discarded by the quantization operation, generating a high compression rate and a small perceptible loss in the image quality.

2. Descriptions in C Language

The 2-D DCT described in hardware and in C used the 2-D DCT separability property, where 2-D DCT results are obtained through the use of two 1-D DCT (One Dimensional Discrete Co-sine Transform) architectures.

One important restriction considered into C description of the architecture used for each 1-D DCT calculation is the word width in the input of each operator used into the 1-D DCT pipeline. These word width are showed in fig. 2.

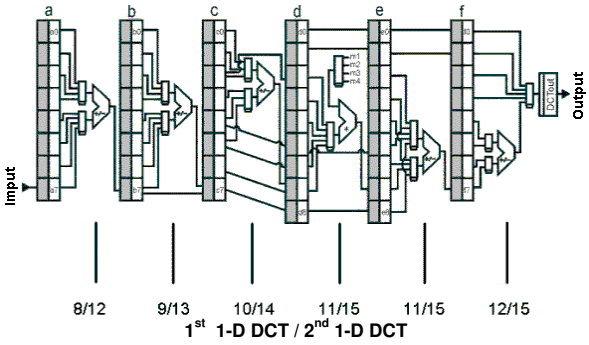

Figure 2 – Word widths in the input of each operator

Another important restriction considered into the 2-D DCT C description is the multiplier structure. This multiplier is presented in fig. 3 and it is inserted into the fourth stage of the pipeline presented in fig. 2. These multipliers make multiplications of the input data by four different constants [1].

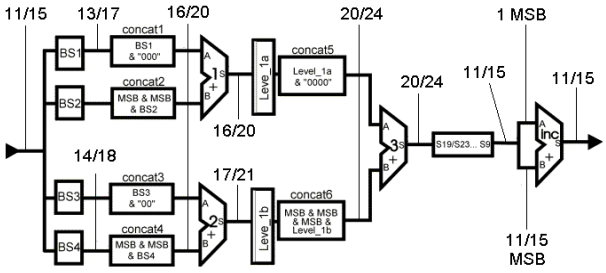

Figure 3 – Multiplier architecture

The multiplications are decomposed in the addition of four different shifts in the data inputs. This decomposition is responsible for the use of simplified values for the four constants used into the calculation of each 1-D DCT.

Another important aspect are the values generated in the multipliers outputs, that are truncated to preserve just the integer part of the results. These multipliers have an incrementer that are used just to correct the negative

outputs. This incrementer is not used to round the output values, that are truncated.

The multipliers used in the architectures of the 1-D DCT calculation are practically equal to multiplier used in the quantization architecture. The differences between these multipliers are the number of constants used like inputs and the rounding in the quantization multiplier output.

Two other C descriptions were designed without considering the hardware restrictions cited previously. These descriptions generate ideal results for the 2-D DCT and quantization calculations. These ideal results were used to make comparisons with the real values generated by the architectures descriptions. Comparing the real and ideal values was possible to evaluate the error generated by the 2-D DCT and quantization architectures and then to suggest architectural modifications to minimize these errors.

3. Architectural Simulations

C descriptions were stimulated with a set of thirty matrixes, which were used as input for all designed descriptions. This set of input matrixes was constructed with many types of input data, forming an ampler and representative set of possible input matrixes [4].

The comparison results allow the identification of imperfections in one barrel shifter used into the 1-D DCTs multipliers. The propagation of these imperfections through the structure of the 2-D DCT caused significant impacts in the data quality. This error were corrected and an increase in the quality of the generate results could be perceived.

The comparisons also allows to identify rounding and truncation errors for these architectures. Basically, the minimization of these errors was made through modifications in the rounding architecture used in the 1-D DCT multipliers and also through use of more accuracy constants sets to the 2-D DCT and quantization. The construction of these sets was based in the maximum number of ones used in the binary representation of each constant and in the use of truncation or rounding.

Like this, were generated five sets of truncated constants and five sets of rounding constants both to 2-D DCT and quantization.

These sets were organized in pairs *n* x *m*, where *n* is the number of ones used for the 2-D DCT constants and *m* is the number of ones used for the quantization constants.

Tab. 1 shows the image quality profits with the developed modifications, where the best solution is the use of seven bits into the 2-D DCT and quantization multipliers and rounding its results. The percentage presented in tab. 1 was generated through the comparison between the original error matrixes and the error matrixes obtained with the solutions presented in this paper.

Table 1 – Simulation results

Average Profits in Image Quality

	Truncating the Results				
	4 x 4	**5 x 5**	**6 x 6**	**7 x 7**	**8 x 8**
2-D DCT	47%	75%	77%	77%	77%
Quantization	54%	71%	76%	80%	78%
	Rounding the Results				
	4 x 4	**5 x 5**	**6 x 6**	**7 x 7**	**8 x 8**
2-D DCT	50%	77%	77%	77%	77%
Quantization	53%	72%	77%	81%	78%

4. Conclusions

Considering all corrections proposed in tis paper, it was possible to obtain a profit higher than 80% in terms of error minimization, considering just the multipliers of the 2-D DCT and quantization.

The implementation in VHDL of these suggested modifications is one future work indicated by this paper. Using estimated values, the impacts in terms of resources usage will increases until 26% and 180%, in 2-D DCT and quantization, respectively. These local increases will result in a maximum global increase in use of resources of 28% in the gray scale JPEG compressor architecture.

These obtained results encourage the implementation of the suggested architectural modifications because the increase into the image quality is very significant.

5. References

[1] L. Agostini. *Design of Architectures for JPEG Image Compression (portuguese)*. Master Dissertation – Federal University of Rio Grande do Sul. Informatics Institute. Pos-Graduation in Computer Science Program, Porto Alegre, Brazil-RS, 2002. 143p.

[2] The International Telegraph and Telephone Consultative Committee (CCITT). "Information Technology – Digital Compression and Coding of Continuous-Tone Still Images – Requirements and Guidelines". Rec. T.81, 1992.

[3] V. Bhaskaran and K. Konstantinides. *Image and Video Compression Standards Algorithms and Architectures – Second Edition*, Kluwer Academic Publishers, USA, 1999.

[4] B. Neves, L. Agostini. "Minimização dos Erros Gerados no Cálculo da DCT 2-D de um Compressor de Imagens JPEG Descrito em VHDL". In: XIV SIC – UFRGS - XIV Salão de Iniciação Científica da UFRGS, 2003, Porto Alegre, Brasil. (resume).