PARTIALLY PARALLEL BLOCK ADAPTIVE ERROR DIFFUSION HALFTONING OF GREYSCALE IMAGES IN MULTI-PROCESSOR ENVIRONMENT

Mondal Sabyasachi and Arijit De
VLSI, Transwitch India, 2-9A Safdurjung Enclave, New Delhi, India

Abstract :

Though standard grey scale digital images contain large number of colour levels, printing and display devices can render only a very few levels. Beside this, many other applications (like image transmission in low bandwidth channel) require reduction in number of colour levels. However direct reduction of number of colour levels significantly affects textures and other features of the image, and hence, the visual quality. So instead of direct reduction of colour level, halftoning algorithms are used to retain various feature of parent image as far as possible. So in digital image processing applications like printing and display devices, halftoning methods are widely used. Among several halftoning algorithm, error diffusion halftoning produces most visually pleasant result. In error diffusion logic, the quantization error of the target pixel is diffused in appropriate proportion to the neighboring pixels through carefully designed filters. The problem with error diffusion is that it is highly sequential in nature. To increase the processing speed, Block Error Diffusion (B.E.D.)[1] methods for grey scale have been proposed earlier. In this type of algorithm, error diffusion is done within fixed size neighboring pixel blocks, instead of error diffusion within neighboring pixels(which is used in traditional error diffusion algorithm)But the output from this sort of block processing suffers from serious edge artefact. To reduce the artifacts and improve the output quality, an “block-wise” adaptive error diffusion filter (instead of fixed error diffusion filter) is designed which also works in block processing mode. In this scheme, the coefficients of the block adaptive filter are updated in each step using adaptive algorithms. A threshold modulation is also used to improve the sharpness control. This modulated threshold value is used in the quantizer. Finally a DBF (Deterministic bit flipping) quantizer is employed to break the limit cycles and other remaining artefacts which are produced at different stages due to the use of block processing. The algorithm proposed here is fully parallel in case of intra-block pixel processing. But in case of inter-block processing, it is sequential. However this sequentialism can be avoided to some extent if one can use modified Floyd-Steinberg algorithm in multi-processor environment as shown in section [2].In [2], multi-processor implementation has been done for the case pixel wise error diffusion algorithm. In this paper, a modified version of the abovementioned algorithm is designed for use with block wise error diffusion. An extensive timing analysis has been carried out to meet the scheduling criteria requirement. The problem of inter-block communication time has also been discussed.

In this proposed methodology, higher the block size, higher parallelism and speed are achieved but with larger memory requirement, and hence increased hardware. But it is a well-known fact that the two constraints, namely high speed and low area, cannot be satisfied together. So there should be a user defined trade-off between block size, speed and the memory requirement. In result section the proposed algorithm is applied on standard “Lena” image. Firstly we have shown the output for bi-level case(where only two level exists, 0 and 255). Though modern printers can print some few more level, still we check our algorithm against Bi-level rendering for worst case analysis. Then comparison of several cases, like output with no adaptive measure and no DBF, are provided. Finally the output of multi-level halftoning is shown and compared

References

[1] N. Damera-Venkata, B. L. Evans, “FM halftoning via block error Diffusion”, Image Processing, 2001 Proceedings. International Conference on, Volume: 2, 7-10 Oct. 2001 Pages: 1081 - 1084

[2] P.T. Mataxas, “Parallel Digital halftoning by error diffusion”, June 2003, ACM Proceedings of, the Paris C. Kanellakis memorial workshop on Principles of computing & knowledge.