A Survey of Imprecise Signal Processing

Karan Daei-Mojdehi
School of Electrical Engineering and Computer Science, University of Central Florida
Orlando, Florida, United States
k.mojdehi@knights.ucf.edu

***Abstract*— Approximate computations allow for trading off between precision and other desired qualities like low energy consumption in a portable device with a limited battery life, or reduction in arithmetic complexities in favor of faster computations in a station. Signal processing applications are widely used in almost every aspect of our modern life including multimedia. What's interesting is that signal processing algorithms can be structured to allow for approximate computations. Due to these applications' inherent error-resiliency, trading off between accuracy in computation and other qualities will result in a more efficient system in terms of resource utilization with little degradation in quality. In this work, after examination of recent approaches in this field, we will compare them and suggest a hybrid method for realization of an even more efficient systems. Also a hypothesis that can make realization of a simple control strategy, in favor of an autonomous unit, is suggested for investigation.**

Keywords—approximate computation; signal processing; discrete cosine transform (DCT); efficient resource utilization.

I. INTRODUCTION

It is favorable to structure algorithms to allow for trading off between accuracy and utilization of resources such as time, power, and bandwidth. Approximate computation is the tool which enables us to achieve this balance between precision and resource utilization, since it reduces computations complexities. Generally approximate computation is introduced at two different levels: algorithm-architecture level, and logic-circuit level [8].

In the *algorithm-architecture level*, using statistical or probabilistic techniques, computations are separated into two groups: significant and insignificant computations. The first group's elements are computations that have a much higher effect on the overall quality of the output compared to the second group, and are performed in an error-free manner without any approximations. However, the second group will be subject to approximations to allow for precision trade-offs with minimum possible degradation in quality.

On the other hand, at *logic-circuit level*, circuit complexity for certain arithmetic operations is reduced which significantly reduces power consumption. For instance number of transistors and capacitances in a mirror adder can be reduced ensuring minimal error in a full-adder. Using this approximate full adder for least significant bit of operations will result in a lower power consumption with minimum impact on output quality. It should be noted that optimum approximation is gained with a synergistic cross-layer exploitation of the introduced approximation levels.

In Digital Signal Pprocessing (DSP) applications with evolving environments, like communications, where transport of data such as voice or video has real-time constraints and is not at a constant rate, approximate computation techniques allow for trading off quality with less arithmetic complexity to avoid interruptions in data transport, i.e. we can achieve graceful degradation instead of system failure.[1]

There are two characteristics of DSP algorithms that make them a suitable candidate for approximate computations: inherent error-resiliency and natural incremental refinement of their structures. *Error resiliency* arises from the fact that DSP algorithms are designed to be noise tolerant, and statistical or probabilistic computations are used in several of them. *Incremental refinement* is a central concept in computer science in approximation processing, which can be summarized as follow: a computation structure with incremental refinement property consists of a succession of stages, each of which improves the answer produced by the previous stage. A simple example of a computation algorithm with incremental refinement property is Newton's root finding method. An example of a DSP method with incremental refinement property is Levinson recursion algorithm [9] for linear prediction which is widely used in speech processing. Exploiting error resiliency, imprecise calculations will result in minimum output degradation while the incremental refinement property allows for use of different approximate computation techniques.

Multimedia systems which embrace our modern life, exploit DSP algorithms in many aspects and involve computationally intensive data processing. The nature of multimedia systems adds two more points of interest from approximation perspective: there's significant redundancy in processing of large data sets, and since multimedia systems deal with human perception, a small amount of error in output won't be recognized. Surveyed papers exploit these properties and propose different approaches to a dynamic approximation scheme which allow for trading off between accuracy and resource utilization.

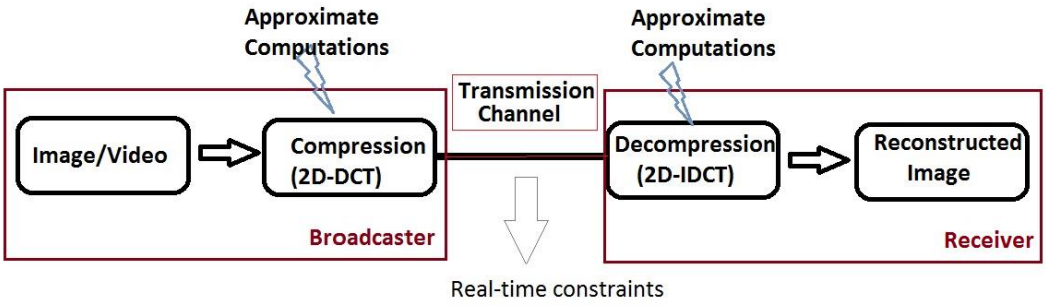

Figure 1. A simplified video broadcaster-receiver scenario showing target of approximate computations

II. Discrete Cosine Transform

One of the key tools used in multimedia systems and specially image/video compression (as well as the surveyed papers) is discrete cosine transform (DCT) which is a data compression scheme. DCT transforms a series of data into frequency domain and represent them by a set of coefficients. The advantage of DCT compared to similar schemes like discrete Fourier transform, lies in the fact that energy of the original signal is mostly concentrated in a few low frequency components of the DCT [3].

First we will have a short summary of each of the surveyed papers and after a comparison, a hybrid scheme will be suggested.

III. "Approximate Signal Processing"

In [3], which is considered a classic paper from time point of view (1997), basics of approximate computations in DSP is covered. It has been alleged that an adequate formal approach to approximate signal processing was not available to that date, while approximate computing was well documented in computer science literature. The author starts by explaining that approximate processing advances are based on two basic assumptions: 1. the desired algorithm can be implemented in a manner which has the incremental refinement property and 2. The performance of the mentioned algorithm is quantifiable to an acceptable level. These two assumptions can be made in many DSP applications since multistage and iterative algorithms are common in DSP. In order to show that DSP has a great potential for approximate processing, several schemes are proposed that either exploit available DSP algorithms with incremental refinement structure for approximation in a context, or suggest structures with this property for ones that do not have it so that they can benefit from approximate processing too. The presented schemes in the paper are for the following applications: Signal Detection using FFT, Spectral Analysis using DFT, Low-Power Frequency-Selective Filtering, and Image Decoding using 2D-Inverse DCT (IDCT). The last scheme is a point of interest to this survey since it can be compared with other papers. In this section, Nawab et al. assert that when a video or image is being broadcasted, receivers' network bandwidth is unknown to the sender while receivers might need to adjust to variable data rates due to local bandwidth limitations which dictates use of an incremental refinement structure.

In video broadcasting context (look at Fig.1), two dimensional DCT is widely used for compression at the broadcaster station while IDCT is used at the receiver, and it is known that a significant proportion of computations required in image/video decompression are made up of IDCT computations. An incremental refinement structure for 2D-IDCT with distributed arithmetic architecture is proposed with a bit-serial ordering in distributed arithmetic. This ordering in

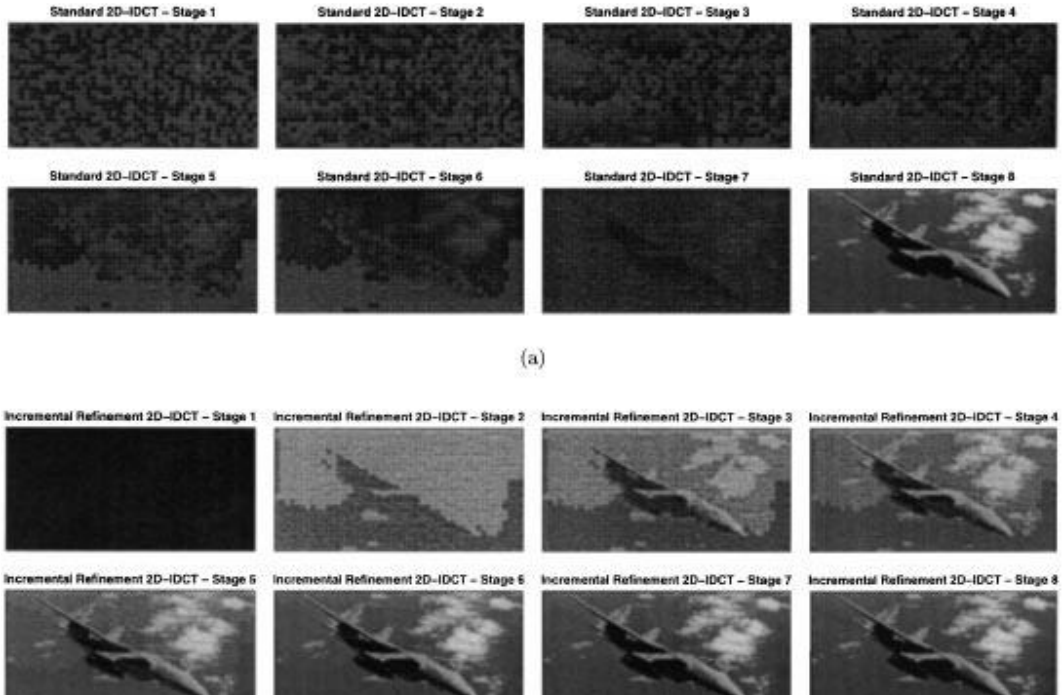

Figure 2. (a) The successive results obtained using a standard distributed arithmetic approach to performing the 2D-IDCT (b) The results obtained during 8 successive stages of 2D-IDCT refinement using the successive refinement architecture described in the text.[1]

architecture allows the processing to begin computations from Most Significant Bit (MSB) of input words which have the most effect on output and advance progressively towards Least Significant Bit (LSB). Another Important aspect of the work done is avoiding break down of the 2D-IDCT into 1D-IDCT of rows and then another 1D-IDCT on columns (which is fairly common in similar implementations), although it will reduce computation's complexity, the intermediate results in the first stage of 1D-IDCT of rows will not represent approximation of the desired output. With the proposed architecture and structure for data processing, even intermediate results of calculations can represent an approximate of the output which makes room for the desired trade-off between precision and resource utilization. This can be observed in Fig. 2(b) while Fig. 2(a) shows intermediate results of an implementation without incremental refinement property. Having this scheme, an appropriate control strategy can be used by each receiver to terminate the decompression process (IDCT) at an intermediate level corresponding to available resources to the device.

IV. Using FPGA-Based Architectures For Low-Power Image Processing

In [2], an architecture for DCT computations with a scalable quality has been proposed. This architecture is FPGA-based and exploits FPGA's dynamic partial reconfiguration in order to trade-off computations accuracy with other desired characteristics. This is done by dedicating computation of each coefficient to one specific region of FPGA. A DCT with 8 coefficients for a one dimensional signal will result in an 8x8 zone for an image compression using 2D-DCT. Quality scalability is achieved by dividing this computations to eight different regions of an FPGA so that 2D-DCT can be done for eight different zones (from 1x1 to 8x8) with eight different levels of precision. The other innovation that provides the architecture with a scalable structure, is its capability to be reconfigured to reduce precision of DCT coefficients. These two properties are achieved by using dynamic partial reconfiguration to either add/remove Processing Elements (PEs) in a zone or change internal logic of PEs to reduce precision of the DCT output. In [2], 2D-DCT is broken down into two 1D-DCT of rows and columns, which reduces computations' complexity. Also, parallel distributed arithmetic is exploited to speedup compression computations. To demonstrate the potential of unused DCT computation modules, exploiting FPGA partial reconfiguration, these modules are reconfigured to do a motion estimation task in a case where a reduced zone of 8x8 DCT is used for compression. Power consumption comparison of different modes of this architecture can be seen in Fig. 3. Note that by using larger zones of DCT coefficients, power consumption increases while precision improves.

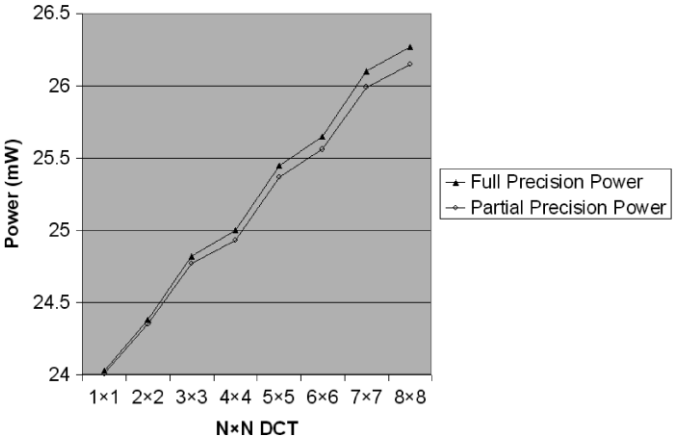

Fig. 3 Comparison of power consumption of different zones of DCT [2]

Another FPGA-based architecture that is proposed for imprecise image processing is described in [10]. This architecture provides an autonomous logic which monitors output health metric and provokes a multi-objective online evolutionary algorithm for self-repair when this health metric goes below a user-given threshold. This evolutionary algorithm also alters configurations to determine the most energy-efficient one while maintaining the desired quality. By looking at results we can see that although main goal in [10] was not lower power-consumption, it still maintains an acceptable level of it even when it is repairing a degraded output in an image processing task.

V. "Dynamic Bit-Width Adaptaion in DCT"

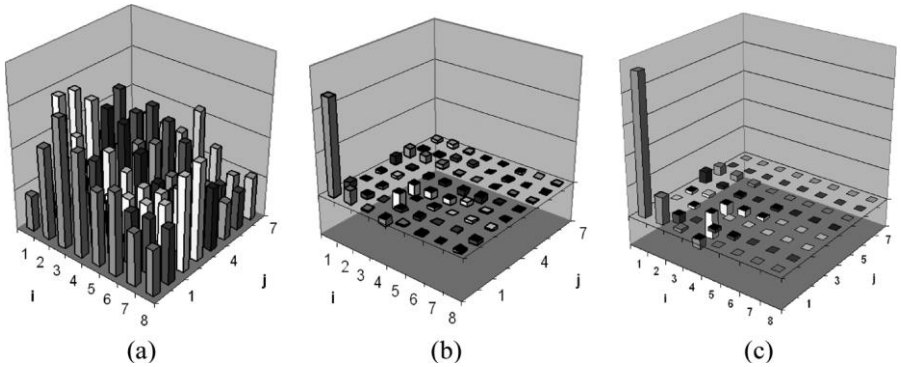

Fig. 4 (a) Normalized 8x8 block of image data. (b) Output of 2-D DCT (64 DCT coefficients). (c) Output of quantization operation

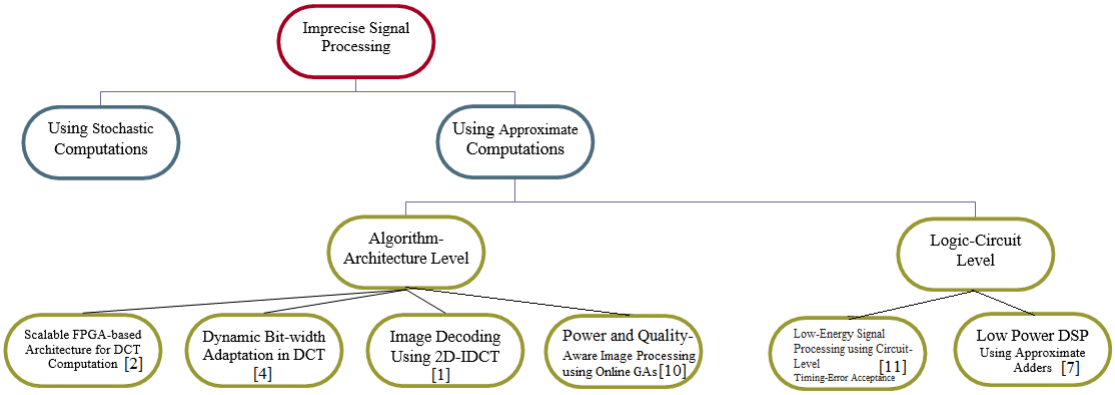

Fig. 5 Taxonomy of the surveyed approaches in Imprecise Signal Processing

There is a series of paper which propose a technique to trade off image quality and DCT computation energy based on bit-width adaptation of DCT coefficients ([4]-[6]). In this series, it is asserted that VLSI implementation of DCT operations prohibits use of floating point arithmetic and in order to satisfy power and system performance constraints, bit-width selection in a fixed point arithmetic is essential. The selected paper among these, which is to the interest of this survey, proposes an algorithm that determines which subset of 64 coefficients of an 8x8 DCT contains the maximum possible energy of the original signal under a desired output precision condition [4]. Having compaction property of DCT in mind, signal energy of the DCT output is concentrated in a few low frequency components (this is demonstrated in Fig. 4). Therefore, precision on higher frequency components can be lowered by reducing size of their bit-width. This will result in an efficient trade-off between image quality and computation complexity. Again 2D-DCT is decomposed into two 1D-DCTs in this work. Peak-signal to noise ratio (PSNR) is selected as measure of signal quality in this work. It is claimed that the proposed reconfigurable DCT architecture can achieve power savings varying from 36% to 75% compared to normal operation with only 0.61 to 5.66 dB reduction in PSNR (normal operation had a PSNR of 36.16 dB).

VI. Using Approximate Adders for DSP

This approach has a fundamental distinction with previous ones: approximate computations are targeted at logic-circuit level while previous introduced ones where mostly targeting approximate computations at algorithm-architecture level. In [7] the relaxation that DSP blocks' resiliency to error brings on numerical exactness, is exploited to carry out approximate computations. Since most of DSP blocks are made up of adders and multipliers (which can be treated as shifters and adders again), different schemes for approximate adders are proposed which reduce logic complexity of them at transistor level. This will lead to power consumption reduction in two different ways. First, smaller hardware results in an inherent reduction in switched capacitance and leakage. Second, this complexity reduction results in shorter critical paths, enabling voltage scaling [7]. In order to come up with an approximate full adder, approximate Mirror Adders (MAs) are used instead of conventional ones. Approximate MAs are results of removing specific transistors from a normal MA. Here it is ensured that removal of the specified transistors (a) will not result in a short circuit under any combination of inputs and (b) introduce the minimal error in the final MA truth-table. In the next step, designs for image/video compression using the proposed approximate adders are presented for DSP blocks and benefits of this designs are demonstrated by using the proposed adders in a 2D-DCT for an image compression task. Again PSNR is used as a measure of output quality and power consumption is examined for different levels of PSNR. Here it is claimed that up to 60% power savings compared to a conventional image compression task can be achieved with a PSNR reduction of only 5.7 dB reduction in output quality (base image PSNR was 31.16 dB), while a truncation algorithm with almost the same power saving would reduce PSNR by 12.12 dBs.

Another logic-circuit level technique is introduced in [11]. Here, quality-energy trade-off is gained through aggressive voltage scaling. The timing errors that arise from voltage scaling, degrade output quality intensively where critical paths are longer, which is the case for computation of high frequency DCT coefficients. On the other hand we saw that high frequency DCT coefficients have less impact on output quality. Therefore, in this framework, a dynamic-width adder architecture is proposed in order to perform computations of higher DCT coefficients in a reduced-width mode. This will minimize timing induced errors due to longer critical paths of these coefficients. Simulation results show that up to 82.63% power saving in an image decompression task can be achieved using 2D-IDCT by reducing output PSNR from 31.6 dBs to 28.3 dBs.

VII. Comparison of The Reviewed Approaches

From approximate computations perspective, last two papers ([7] & [11]) have a basic difference with others in that they target approximate computations at logic level while the other three papers try to find a procedure in a DSP algorithm which can be approximated with efficient arithmetic complexity reduction (look at Fig. 4). In [1] a key technical point that has been made about image/video decompression and is not used in other papers (although they are written many years later) is that when 2D-DCT is being used in the receiving stage (2D-IDCT) it is better not to decompose the algorithm into two 1D-IDCTs as it will impair its incremental refinement

TABLE I. IMAGE QUALITY AND POWER CONSUMPTION COMPARISON OF PROPOSED APPROACHES IN AN IMAGE COMPRESSION/DECOMPRESSION TASK

Approach	Approximation Level	Decomposes 2D-IDCT	Uses Distributed Arithmetic	PSNR(dB) / Base PSNR (dB)	Power Savings (%)	Notable Feature
Scalable FPGA-based Architecture for DCT Computation [2]	Algorithm-Architecutre	Yes	Yes	0.73	9.43%	Can reconfigure unused resources to perform additional tasks
Power and Quality-Aware Image Processing using GAs [10]	Algorithm-Architecture	Yes	No	0.99	16.19%	Can automatically detect quality degradation and awake a GA to improve quality as well as energy saving
Dynamic Bit-width Adaptation in DCT [4]	Algorith-Architecture	Yes	Yes	0.85	74.8%	High efficiency compared to others
Low Power DSP Using Approximate Adders [7]	Logic-Circuit	Yes	N/A	0.82	60%	Reduction in Critical Path
Low Energy Signal Processing using Circuit-level Timing-Error Acceptance [11]	Logic-Circuit	Yes	No	0.89	82.63%	High efficiency compared to others

property. Instead, they have insisted on using this decomposition to reduce arithmetic complexity and used other methods to give the algorithm incremental refinement property. For example in [2] dynamic architecture of FPGAs and partial reconfiguration has been exploited to provide a flexibility between precision and resource utilization while in [4] bit-width adaptation is the tool that enables the system to trade-off accuracy with power consumption. Another reason that 2D-DCT is decomposed in other papers might be the fact that they are focusing on proposing a scheme for the compression task in the broadcaster and not the decompression in receiver. Advantage of [2] is in its ability to utilize the unused resources when a level of approximation in computations is used (of course only if it's desired) while the others only focus on lower power consumption. In [2] and [4] almost the same method for signal approximation is used which is selection of a smaller zone of DCT coefficients, but [4] does the selection process at a finer granularity level (64 possible zones vs. 8) with a specific algorithm that makes the approximation more efficient.

In order to be able to compare the results from these approaches, we normalize the PSNR of the approximated image/video (at maximum power efficiency which means lowest quality) by dividing it by the PSNR of the original image and compare their power savings (compared to a conventional image compressor used in the same task) in Table I. Please note that although the first approach seems far away from the other two in this aspects, its focus is not only on power saving, but also in utilization of unused resources and the power savings related to them is while the unused PEs are performing a motion estimation task.

VIII. A HYPOTHESIS TO BE INVESTIGATED

In the current literature there is no record showing investigation of feasibility of performing image compression using DCT's formula for two dimensional arrays, proposed in [1], instead of decomposing it like the rest of reviewed approaches. Keeping the incremental refinement property will allow much finer granularity levels compared to other proposed approaches, but quality-energy curve of this approach has to be compared with others through some sort of simulation or practical implementation on a platform. If the proposed approach in [1] can keep up with any of the recent methods in terms of energy-efficiency, its finer granularity level, provided at algorithm level which doesn't require much of hardware overhead, can be exploited in implementation of an autonomous controller which simply determines the appropriate stage to terminate the algorithm by monitoring available resources and a pre-specified desired quality level in different circumstances.

IX. A HYBRID APPROACH

What should be noted about approximate image processing and the two distinct levels that they are conducted at, being algorithm-architecture level and logic-circuit, is that the full potential of multimedia applications' error-resiliency is not exploited by exercising ACs at a single level [8]. It is essential to use AC techniques across various layers of design to get to the optimum quality-energy curve in such systems. What we suggest here is a hybrid approach made up of combination of the two most efficient approaches covered so far, one at algorithm level and the other at logic-circuit level. By taking a look at Table I, it can be perceived that among the algorithm level approaches, [4] has presented a more efficient technique by introducing a finer flexibility for choosing DCT coefficients while benefiting from an algorithm that finds the most efficient subset among them. While this approach seems promising, by further examination of their technique for bit-width adaptation, it can be noted what is being done for reducing bit-width is same as truncation of LSBs. On the other hand, in [8] the PSNR of a decoded image using truncation of LSBs was compared with another method which used approximate adders for computation of those LSBs instead of removing them from equations. The results show that PSNR can be greatly improved with slightly more power

consumption. What we suggest here as a hybrid approach, is using the algorithm in [4] for identification of the appropriate numbers of bit-widths for different DCT coefficients, given a desired output quality, and in the next step, instead of changing bit-widths, utilize the approximate adders proposed in [8] for computation of LSBs that would've normally been truncated in [4]. This will enhance output quality, which means adjustments must be made to the algorithm, which determines bit-widths for different DCT coefficients, to account for approximate adders influence on output quality as well. While using approximate adders instead of truncating lower bits slightly increases power consumption, the fact that the improved output quality is taken into account in the algorithm will guarantee an overall improvement in energy efficiency of the system.

X. Conclusion

Most of the mobile devices that are widely being used, benefit from DSP algorithms in their underlying layers. Including a scalable architecture for achieving a balance between low-power consumption and acceptable performance level in these structures can lead to more efficient usage of our portable devices in different circumstances considering their limited energy resource. Here in this work after an introduction to approximate computations' concepts, basics of implementing them in DSP algorithms were explained. Then, some of the recent approaches exercising those basics in a DCT image compression task and innovations used in them were examined. By comparing approximation efficiency among them and the level AC was being using, we suggested a hybrid cross-layer approach which would provide an even more efficient paradigm for trading-off accuracy for power consumption.

References

[1] Nawab, S.; Oppenheim A.; Chandrakasan A.; Winograd J.; Ludwig J., "Approximate Signal Processing," Journal of VLSI Signal Processing vol.15, pp.177,200, Jan. 1997

[2] J. Huang, M. Parris, J. Lee, and R. F. DeMara, "Scalable FPGA-based Architecture for DCT Computation Using Dynamic Partial Reconfiguration," ACM Transactions on Embedded Computing Systems, Vol. 9, No. 1, Art. 9, October, 2009, pp. 1 - 18.

[3] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete Cosine Transform," IEEE Trans. Computer, vol.C-23, no.1, pp.90-93, Jan. 1974

[4] Jongsun Park; Jung-Hwan Choi; Roy, K., "Dynamic Bit-Width Adaptation in DCT: An Approach to Trade Off Image Quality and Computation Energy," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , vol.18, no.5, pp.787,793, May 2010

[5] J. Park, J. Choi, and K. Roy, "Dynamic Bit-width Adaptation in DCT: Image Quality Versus Computation Energy Trade-off," Proc. Des. Autom., Test Eur., pp.520-521, Mar. 2006

[6] T. Xanthopoulos and A. Chandrakasan, "A Low-power DCT Core Using Adaptive Bitwidth and Arithmetic Activity Exploring Signal Correlations and Quantiziation," IEEE J. Solid-State Circuits, vol.35, no.5, pp.740-750, May 2000

[7] Gupta, V.; Mohapatra, D.; Raghunathan, A.; Roy, K., "Low-Power Digital Signal Processing Using Approximate Adders," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on , vol.32, no.1, pp.124,137, Jan. 2013

[8] Roy, K., "Approximate computing for energy-efficient error-resilient multimedia systems," Design and Diagnostics of Electronic Circuits & Systems (DDECS), 2013 IEEE 16th International Symposium on , vol., no., pp.5,6, 8-10 April 2013

[9] N. Levinson, "The Wiener RMS Error Criterion in Filter Design and Prediction," J. Math Phys., vol.25, pp.261-278 Jan 1947

[10] N. Imran, R. A. Ashraf, and R. F. DeMara, "Power and Quality-Aware Image Processing Soft-Resilience using Online Multi-Objective GAs," International Journal of Computational Vision and Robotics, Vol. 5, No. 1, pp. 72 – 98, 2015. DOI: 10.1504/IJCVR.2015.067154

[11] He, Ku, Andreas Gerstlauer, and Michael Orshansky. "Low-energy signal processing using circuit-level timing-error acceptance." In IC Design & Technology (ICICDT), 2012 IEEE International Conference on, pp. 1-4. IEEE, 2012.