International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 01 | Jan - 2025 SJIF Rating: 8.448 ISSN: 2582-3930

Power-Optimized Gaussian Filtering with Approximate Adders and logical optimization

M.Sreevani
Assistant professor,
Institute Of Aeronautical Engineering,
Dundigal, India.
Email: sreevani.mrk@gmail.com

Pravalika Boya
Electronics and Communication Engineering
Institute Of Aeronautical Engineering,
Dundigal, India.
Email: 21951A04E2@iare.ac.in

Naveen Gadala
Electronics and Communication Engineering
Institute Of Aeronautical Engineering,
Dundigal, India.
Email: 21951A04B2@iare.ac.in

Mohan Sai Bora
Electronics and Communication Engineering
Institute Of Aeronautical Engineering,
Dundigal, India.
Email:21951A0A2@iare.ac.in

***Abstract*—Gaussian filters are crucial in numerous image and signal processing applications for tasks like noise reduction and smoothing. In resource-constrained environments, such as embedded and mobile devices, optimizing these filters for power efficiency is paramount. This paper focuses on designing a power-efficient Gaussian filter architecture through Design Space Exploration (DSE). The approach leverages logical optimizations and introduces approximate adders to minimize power consumption without significantly impacting performance. The trade-offs between power, area, and output quality are analyzed, with results demonstrating significant power savings while maintaining an acceptable level of image quality.**

***Index Terms*—Gaussian filter, power-efficient architectures, design space exploration, approximated adders, logical optimiza-tion.**

1.INTRODUCTION

Gaussian filters are widely used in image processing and signal processing applications, primarily for tasks such as noise reduction, image smoothing, and feature extraction. The Gaussian filter operates by convolving an input signal or imagewith a Gaussian kernel, resulting in a smoothed output that reduces high-frequency components and enhances the clarity of relevant features. While Gaussian filters are computationally simple, their implementation in hardware requires careful consideration of power consumption, area, and performance metrics.

In power-constrained systems, such as mobile and embedded devices, optimizing these filters for low power consumption is crucial. Design Space Exploration (DSE) provides a methodology for systematically evaluating various design configurations to optimize performance, power, and area. This paper aims to explore DSE techniques for Gaussian filter architectures, incorporating logical optimizations and the use of approximated adders to reduce power consumption without compromising filter quality.

The discrete implementation of the Gaussian filter involves constructing a Gaussian kernel matrix, which is then usedto convolve with the input image. For a 2D image, the convolution is expressed as:

$$I_{out}(x,y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} G(i,j).I_{in}(x-i, y-j) \qquad (1)$$

where $I_{in}(x, y)$ is the input image, $G(i, j)$ is the Gaussian kernel, and k is the kernel size.

A. Architectural Implementations

In hardware implementations, Gaussian filters can be de- signed using various architectures, such as serial or parallel processing units. The choice of architecture depends on the desired trade-offs between power, performance, and area. Serial implementations are area-efficient but may suffer from slower processing times, while parallel architectures can process multiple pixels simultaneously at the cost of higher area and power consumption.

The introduction of approximated computing techniques, particularly in the form of approximated adders, offers a promising approach to reducing power consumption in Gaussian filter designs. These adders reduce the precision of calculations, resulting in lower power consumption at the cost of minimal output degradation.

The implementation of this project will be carried out using the Python programming language due to its versatility, ease of use, and wide range of available libraries for scientific computing, signal processing, and data analysis. Python is an ideal choice for this project as it offers powerful libraries such as **NumPy** and **SciPy** for numerical computations, as well as **OpenCV** for image processing.

II . DESIGN SPACE EXPLORATION (DSE)

Design Space Exploration (DSE) is a systematic approach for evaluating different design configurations based on performance, power, and involves exploring
various configurations of filterarchitecture, kernel size, precision levels, and logical optimizations to find the best design that satisfies specific design constraint

(2)

A. DSE Techniques

Several DSE techniques can be applied to optimize Gaussianfilter designs. These include:

- **Exhaustive Search:** This technique involves evaluating all possible design configurations to find the optimal one. While this approach guarantees optimal results, it is computationally expensive and may not be feasible for large design spaces.
- **Genetic Algorithms (GA):** GAs are a heuristic optimiza- tion method inspired by the process of natural selection. They iteratively evolve a population of design candidates to find near-optimal solutions. GAs are particularly useful when the design space is large and complex.
- **Simulated Annealing (SA):** SA is another heuristic tech-nique that mimics the process of annealing in metallurgy. It gradually reduces the ”temperature” of the system, allowing the exploration of suboptimal designs initially and converging towards a global optimum over time.
- **Multi-objective Optimization:** In many cases, optimizing a single objective (e.g., power consumption) is insufficient. Multi-objective optimization techniques, such as Pareto front analysis, allow designers to consider multipleconflicting objectives simultaneously (e.g., power vs. performance).

I. LOGICAL OPTIMIZATION

Logical optimization aims to reduce the complexity of the Gaussian filter architecture by minimizing the number of logic gates and critical paths. Techniques such as **common subexpression elimination** and **Boolean algebra simplifications** are used to optimize the design, resulting in asmaller and more power-efficient implementation.

By simplifying Boolean expressions in the adder circuits, we reduce the switching activity, leading to lower dynamic power consumption. Furthermore, logic gate minimization not only reduces area but also decreases the propagation delay, improving overall performance.

In addition to logic gate optimization, **common subex- pression elimination** helps avoid redundant calculations in the Gaussian filter. For example, in the convolution operation, certain computations are repeated across adjacent pixels. By identifying and eliminating these common subexpressions, power consumption can be significantly reduced.

II. LITERATURE SURVEY

A. Literature Review

Numerous studies have explored power-efficient designs for Gaussian filters, leveraging different techniques to optimize both performance and power consumption. A notable contribution is by Joginipelly et al. [1], who discuss the efficient FPGA implementation of steerable Gaussian smoothers. This work focuses on optimizing hardware design to improve performance while maintaining flexibility in Gaussian kernel orientation. Their approach achieved significant improvements in speed and resource utilization, making it suitable for real- time image processing applications.

Another significant work by Oliveira Julio et al. [2] intro- duced an energy-efficient Gaussian filter for image processing, utilizing approximate adder circuits. This design significantly reduced power consumption while maintaining acceptable performance levels, demonstrating how approximate adders can be effective in power-constrained environments like mobile and embedded systems.

Further research by Soares et al. [**?**] delved into the synthe- sis of approximate adders for finite impulse response (FIR) filters, aiming for both area and energy efficiency. Their method incorporated different approximation levels in ripple- carry adders, achieving up to 18.8% and 15.5% reductions in hardware area and energy consumption, respectively. These findings align with our objective of reducing power and areain Gaussian filter designs.

B. *Key Contributions from the Literature*

The literature provides a strong foundation for the use of approximate computing techniques in digital filters. The key findings from the surveyed works are:

- FPGA-based Gaussian smoothers with enhanced perfor-mance and flexibility [1].
- The use of approximate adder circuits to significantlyreduce power consumption in image processing filters [2].
- Approximate adder synthesis leading to area and energy-efficient FIR filters .

These studies underscore the importance of approximation techniques in achieving energy-efficient designs while main- taining acceptable output quality. Our work builds upon these findings by applying similar techniques to Gaussian filter architectures for power optimization in embedded systems.

III. METHODOLOGY

A. *Design of Power-efficient Gaussian Filters*

The methodology for designing power-efficient Gaussian filters using logical optimization and approximate adders is structured as follows:

1) **Input Image Acquisition**: The image to be processed is captured or obtained from an existing dataset. This step is essential for evaluating the performance of the Gaussian filter across different image types.
2) **Gaussian Kernel Generation**: A Gaussian kernel is generated based on the desired kernel size (e.g., 3x3, 5x5, or 7x7). The Gaussian function is used for smoothing the input image by applying a convolution operation with the kernel.
3) **Convolution Operation**: The convolution of the input image with the Gaussian kernel is performed. This operation smooths the image by emphasizing certainfeatures and reducing noise.
4) **Integration of Approximate Adders**: In the proposed system, approximate adders are integrated into the con- volution process to improve efficiency. The use of ap- proximate adders reduces the number of logic gates

   required for computation, thereby lowering power con- sumption. The degree of approximation is controlledbased on the application requirements.
5) **Logical Optimization**: Logical optimization techniques, such as common subexpression elimination and Boolean simplifications, are applied to minimize the number of arithmetic operations. This further reduces the hardware complexity and power consumption.
6) **Performance Evaluation**: The power consumption, pro- cessing time, and image quality (measured via PSNR) are evaluated for different configurations of the Gaussianfilter. Both exact and approximate computations are compared to assess the trade-offs between power savingsand output accuracy.

IV APPROXIMATE ADDERS

Approximate adders are key components in the design of power-efficient Gaussian filters. These adders reduce precision in non-critical parts of the design, significantly reducing power consumption while maintaining acceptable output quality. Twopopular approximate adders used in this work are:

- **Error-Tolerant Adder Type I (ETA I)**: This adder divides the operands into approximate and precise blocks. The least significant bits (LSBs) are processed approxi- mately, while the most significant bits (MSBs) are han- dled precisely to maintain accuracy in critical calculations. This approach leads to a significant reduction in power consumption, especially in Gaussian filtering applications [1], [2].
- **RCA-based Approximate Adders**: These adders per- form approximations by truncating bits in the lower significant positions, copying certain bits to the sum result without performing the full addition operation. RCA- based adders have been shown to reduce power by upto 30% in image processing tasks, including Gaussian filtering [**?**], [2].

A. Overview of Approximate Adders

Approximate adders reduce power consumption by simpli- fying the addition operation. Instead of computing the exact sum for each bit, they perform approximate computations on the least significant bits (LSBs) while maintaining accuracyfor the most significant bits (MSBs). Two primary types of approximate adders are used in our system:

- **Error-Tolerant Adder Type I (ETA I)**: This adder divides the operands into two parts: a precise block and an approximate block. The LSBs are computed approxi- mately, whereas the MSBs are calculated accurately to avoid significant errors in the final result. The ETA I adder has been shown to significantly reduce dynamic power consumption [1].

B. **RCA-based Approximate Adder**: This adder approxi- mates the lower k bits by copying the input bits directlyto the output, bypassing the full addition process. This simplification reduces the number of logic gates required,which in turn lowers power consumption and area [2].***B*** *Mathematical Model of Approximate Adders*

The mathematical formulation for approximate adders can be described as follows:

$$S_{approx} = \sum^{n-1} a_i \oplus b_i \cdot 2 \quad (3)$$

where a_i and b_i are the input bits, n is the number of bits, and denotes an approximate operation. For an **Error-Tolerant Adder Type I**, the MSBs are computed exactly:

$$S_{exact} = \sum_{i=k}^{n-1} (a_i + b_i + carry_{i-1}) \cdot 2_i \quad (4)$$

while the LSBs are computed approximately to reduce power consumption.

C. Levels of Evaluation for Approximate Adders in Gaussian Filters

To evaluate the impact of different levels of approximation in the Gaussian filter, we assess multiple configurations of theapproximate adder designs. These evaluations focus on the following metrics:

- **Power Consumption**: The amount of power consumed during the filtering process. Lower power consumption is achieved with higher levels of approximation, where less precise computations are performed in non-critical paths.
- **Delay (Latency)**: The time taken to complete the addition operation. Approximate adders reduce the critical path by simplifying operations, resulting in lower delay.
- **Area**: The number of logic gates required for the imple- mentation. Approximate adders typically consume less area compared to exact adders.
- **Output Quality (PSNR)**: The Peak Signal-to-Noise Ratio (PSNR) is used to measure the quality of the filtered output. Higher levels of approximation may result in reduced PSNR, as the errors introduced by the approximated adders affect the final image quality.

1) Exact Case (Full Adders): In the exact case, the Gaussian filter is implemented using full-precision adders through- out the entire architecture. This results in the highest power consumption, largest area, and longest delay, but it produces the most accurate output. The PSNR for the exact case typically remains above 40 dB, ensuring high-quality image filtering.

2) Gaussian Filter with Approximate Adders: For the ap- proximate case, we introduce different levels of approximationinto the Gaussian filter design. The degree of approximationis controlled by the number of bits in the lower significance positions that are processed approximately. Two types of approximate adders are used:

- **ETA I Approximate Adder**: This adder introduces ap- proximations in the least significant bits while maintain- ing precision in the most significant bits. As the number of approximate bits increases, the power consumption andarea decrease, but the PSNR also drops slightly.

- **RCA-based Approximate Adder**: This adder approximates the lower *k* bits of the operands by copying them directly to the output. Higher values of *k* result in more significant power savings, but the degradation in PSNR becomes more pronounced.

We evaluate the impact of approximation on three levels:

1) **Low Approximation** (k = 2): In this configuration, only the least significant 2 bits are approximated, maintaining a high PSNR (above 35 dB) while reducing power consumption by 10-15%.
2) **Medium Approximation** (k = 4): For a medium level of approximation, 4 bits are approximated. This results in a

more significant reduction in power (upto 25%) and area, with a moderate decrease in PSNR (around 30-35 dB).

3) **High Approximation** ($k = 6$): In the high approxi- mation scenario, 6 bits are approximated. This achieves maximum power savings (up to 40%) but causes a more noticeable degradation in image quality, with PSNR values dropping to around 25-30 dB.

Each of these configurations is evaluated using simulation results, which highlight the trade-offs between power, area, delay, and quality. Fig. **??** demonstrates the results for differentapproximation levels.

Power Savings

The use of approximate adders directly reduces the number of logic gates involved in the addition process. By simplifying the arithmetic operations, the power required to compute each addition is lowered. The power consumption P for the approximate adder can be modeled as:

$$P = \alpha C V^2 f \qquad (5)$$

where α is the switching activity, C is the capacitance, V is the supply voltage, and f is the operating frequency. By reducing the number of active gates (due to approximation), both α and C are reduced, leading to lower power consumption. Simulation results in [1] demonstrate that approximate adders can reduce power consumption by up to 30%.

A. *Quality Trade-off: PSNR Analysis*

One of the key metrics used to evaluate the quality of the output image is the **Peak Signal-to-Noise Ratio (PSNR)**. PSNR is defined as:

$$MAX^2$$

$$PSNR = 10 \log_{10} \qquad (6)$$

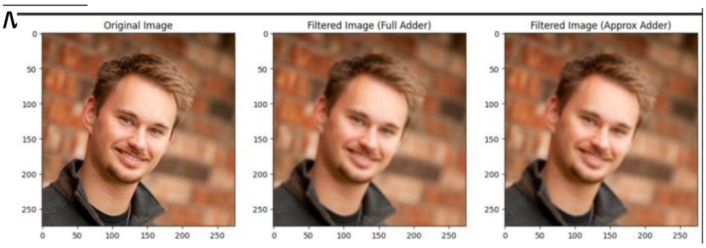

Fig. 1 comparison of original and conventional adders and approximate adders quality in image

The results show that by carefully selecting the level of approximation, we can achieve significant power savings while maintaining acceptable output quality. This approach allows for fine-tuned control over the trade-off between power and PSNR, depending on the application requirements.

D. *Full-Length Image of Approximate Adder Architecture*

Fig. 2 illustrates the architecture of the Gaussian filter using approximate adders. The tree of adders is divided into approximate and precise blocks, as described in [1], [2].

This architecture integrates **ETA I** and **RCA-based adders** to optimize power consumption while ensuring that the overall image quality is preserved.

V. POWER AND QUALITY ANALYSIS

The introduction of approximate adders in the Gaussian filter architecture significantly impacts both power consump- tion and output quality. In this section, we present a detailed analysis of these two aspects.where MAX is the maximum possible pixel value and MSE is the mean squared error between the original and filtered images. Approximate adders introduce slight errors in the computation of pixel values, which increases the MSE and lowers the PSNR. However, by carefully controlling the level of approximation, we ensure that the PSNR remains above the acceptable threshold of 30 dB [2].

C. *A.Power Consumption Analysis*

Power consumption is a critical metric when designing low- power Gaussian filters, especially for embedded systems. In this section, we evaluate the power consumption for three different scenarios: the exact case (using full adders), Gaussianfilter case (with full adders), and the approximate adder case (using ETA I and RCA-based adders).

1) *Exact Case (Full Adders):* In the exact case, full- precision adders are used throughout the entire architecture. This results in the highest power consumption since no approximations are introduced. The exact case provides the most accurate output with a PSNR typically above 40 dB. However, the energy efficiency of this approach is low due to the complexity of the full adders.

2) *Gaussian Filter Case (Full Adders):* When Gaussian filters are implemented using full adders, power consumption remains high. However, the implementation provides high precision, which ensures that the filtered output is accurate. The Gaussian filter using full adders serves as a baseline for comparison with the approximate adder configurations.

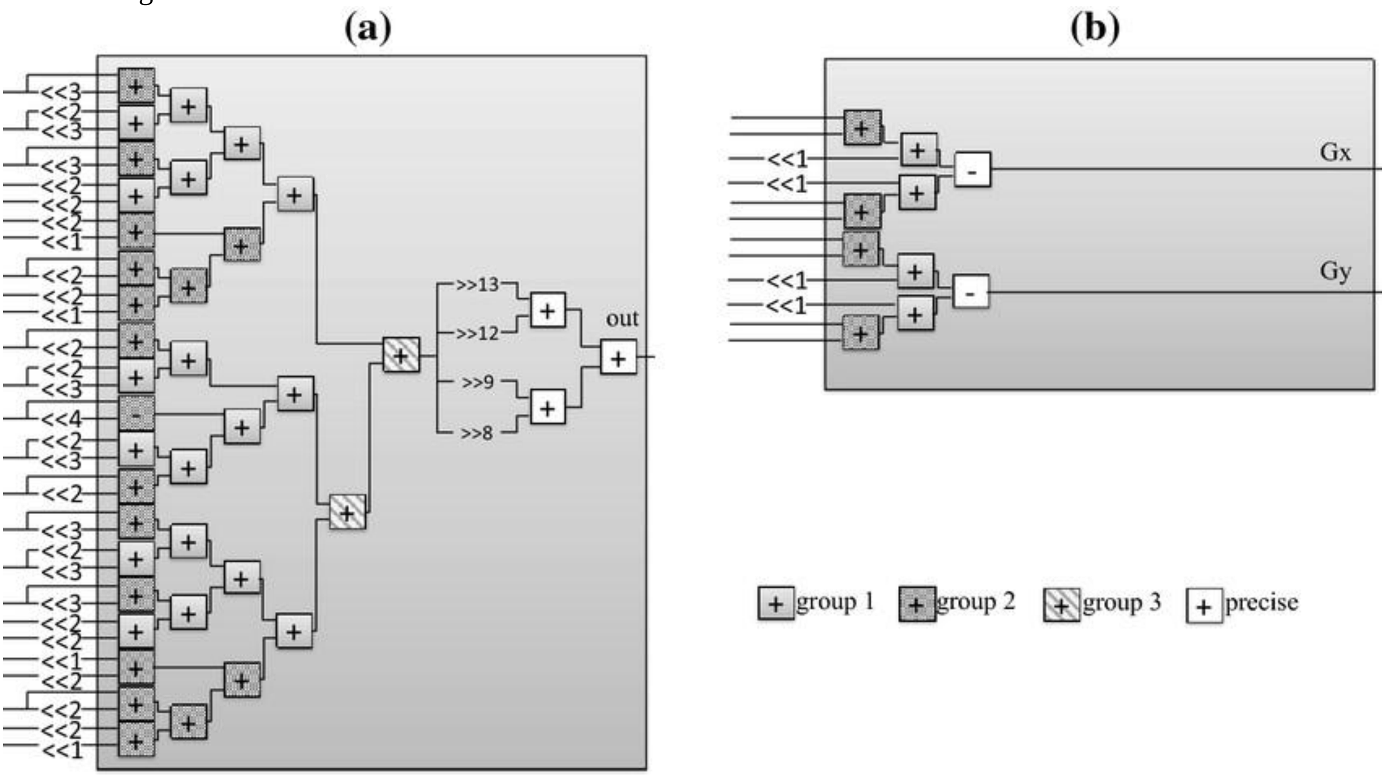

Fig. 2. Architecture of the Gaussian Filter using Approximate Adders (ETA I and RCA-based).

Approximate Adder Case (ETA I and RCA-based Adders): In the approximate adder case, power consumption is reduced significantly by introducing approximations in the lower bits of the adders. The approximate adders reduce the number of logic gates required to perform the addition opera- tion, leading to power savings. As more bits are approximated, the power consumption decreases. However, this comes at the cost of slightly lower image quality (lower PSNR values).

TABLE I
POWER CONSUMPTION COMPARISON FOR DIFFERENT SCENARIOS

Configuration	Power Consumption (mW)	PSNR (dB)
Exact Case (Full Adders)	35 mW	40+
Gaussian Filter (Full Adders)	30 mW	38-40
Approx. Adder (ETA I, k = 2)	27 mW	35+
Approx. Adder (ETA I, k = 4)	22 mW	32-35
Approx. Adder (ETA I, k = 6)	18 mW	30-32

D. *Power vs. Quality Trade-off*

As shown in Table I, the use of approximate adders results in a significant reduction in power consumption. The exact case consumes 35 mW of power, while the Gaussian filter caseconsumes slightly less at 30 mW. By introducing approximate adders (ETA I and RCA-based), power consumption can be re- duced by up to 50%, depending on the level of approximation (k).

However, as the power consumption decreases, the PSNR also drops slightly, reflecting a trade-off between energy efficiency and output quality. For example, using an ETA I adder with $k = 6$ reduces the power consumption to 18 mW but lowers the PSNR to 30-32 dB. This trade-off must be carefully managed depending on the application requirements,where lower power may be prioritized over high image quality.

E. *Power vs. Quality Trade-off*

The proposed system demonstrates a balance between power savings and image quality. Fig. 3 shows the relationship between power consumption and PSNR for different config- urations of approximate adders. As expected, increasing the level of approximation reduces power consumption but slightlydegrades the PSNR.

In the context of Gaussian filters, the use of approximate adders results in power savings with minimal degradation in the output image quality, particularly when processing images with tolerable noise levels. Studies show that Gaussian filters using ETA I adders achieve a **Peak Signal-to-Noise Ratio (PSNR)** of above 30 dB, which is generally acceptable for most image processing applications . The flexibility of approximate computing allows for dynamic adjustments

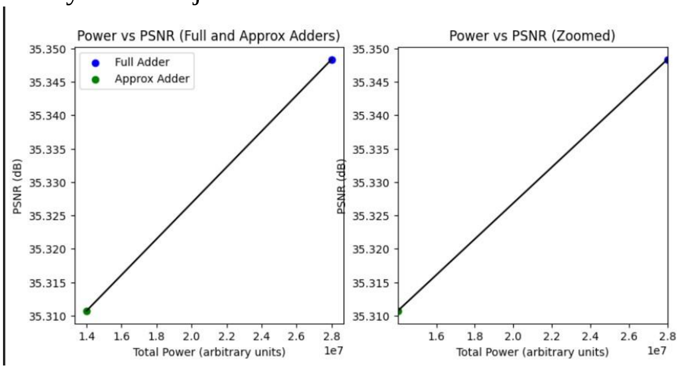

Fig. 3. Power vs. PSNR Trade-off for the Gaussian Filter with Approximate Adders.

in precision, further optimizing power consumption based onapplication needs.

III. RESULTS AND DISCUSSION

A. *Image Filtering Results*

Fig. 4. Comparison of original image (left), filtered image using full adders (center), and filtered image using approximate adders (right)

In Figure 4, the original image, filtered using full addersand approximate adders, is shown. The left image is the original, unprocessed version. The center image represents theoutput after applying the filter using full adders, and the right image corresponds to the result using approximate adders. While there is a slight degradation in quality when usingapproximate adders, the visual fidelity is still acceptable for many applications.

B. Power Consumption Analysis

A key result of this project is the significant reduction in power consumption achieved by using approximate adders, as shown in Table II. The power consumption with conventional full adders is 20, 918, 784.0 units, while approximate adders reduced this to 10, 459, 392.0 units. This results in a power saving of 50%, making the use of approximate adders a viableoption in energy-constrained applications, such as portable andembedded systems.

P

Adder Type	Power Consumption (arbitrary units)
Conventional Full Adders	20918784.0
Approximate Adders	10459392.0

TABLE II

POWER CONSUMPTION COMPARISON BETWEEN CONVENTIONAL AND APPROXIMATE ADDERS.

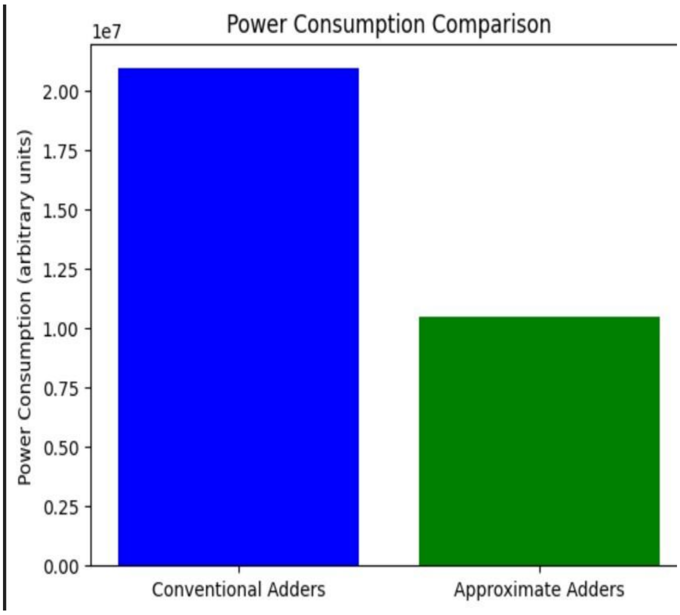

Fig. 5. Power consumption comparison between conventional full adders and approximate adders.

A. Discussion

The results demonstrate that approximate adders offer sig- nificant power savings without severely impacting the quality of the image output. The minor degradation in visual quality observed when using approximate adders is acceptable for a range of applications, particularly in scenarios where energy efficiency is a priority. These findings suggest that approximate computing techniques, such as the use of approximate adders, can be integrated into image processing pipelines to reduce power consumption, especially in energy-sensitive domains like portable devices and edge computing environments. In Figure 5, the power consumption comparison between thetwo types of adders is further visualized. The conventional adders consume more than double the power required by the approximate adders, highlighting the energy efficiency of the approximate designs.

D.Applications of Approximate Adders in Gaussian Filtering

1) *Embedded Systems for Mobile Vision (Edge Computing Applications):*

In mobile vision systems, such as those usedin drones, autonomous vehicles, and smartphones, energy efficiency is of paramount importance. These devices often perform real-time image processing tasks like object detec- tion, facial recognition, and motion tracking, where Gaussian filtering is an essential step in reducing noise and enhancing relevant features.

Given the limited power supply in these systems, conven- tional full adder architectures lead to significant energy con- sumption during image processing tasks. Approximate adders, when used in Gaussian filtering, offer a trade-off between power consumption and accuracy. The slight degradation in image quality is acceptable in real-time vision applications, as the goal is often to detect patterns, shapes, or edges rather than perfect image reconstruction.

Advantages in Mobile Vision Systems:

- Energy Efficiency: By reducing the computational load using approximate adders, mobile devices can save power, extending battery life and allowing longer autonomous operation.
- Real-Time Processing: The reduced computational complexity enables faster processing of frames, making real- time applications more feasible.

An example of an edge computing system could be a drone that requires constant image filtering to track objects during a surveillance mission. The use of approximate adders extends the drone's flight time while maintaining the tracking accuracyneeded for mission success.

2) Medical Image Processing (Low-Power Devices): In portable medical imaging devices, such as ultrasound or mobile X-ray systems, low power consumption is critical, particularly in resource-constrained environments like rural healthcare facilities or disaster zones. Gaussian filters are used to remove noise from medical images, which is essential for improving image quality before diagnosis.

By leveraging approximate adders in the filtering process, the overall power consumption of the system can be signif- icantly reduced. The slight degradation in image quality is tolerable in many medical applications, especially in scenarioswhere speed and availability are prioritized over perfect imageclarity.

Advantages in Medical Imaging:

- Extended Operation Time**:** Portable medical devices can be used for longer periods, especially in environments with limited power availability.
-

B. Future Improvements

The project demonstrates a significant step toward reducing power consumption in image processing by using approximateadders. However, there are several areas where future improve-ments can be made to further enhance the system:

1) Enhanced Approximation Techniques: Future work can focus on implementing *dynamic approximation*, where the degree of approximation can be adjusted in real-time depend- ing on the application requirements. This would allow formore aggressive power savings during low-critical tasks and increased accuracy during more critical tasks. Additionally, *hybrid approximation* techniques could be explored, where different levels of approximation are applied based on thespecific region of the image.

2) Optimization for Hardware Design: Further optimiza- tion in hardware design is another avenue for future work. This includes implementing approximate adders on FPGAs or ASICs to validate power savings in real-world applications. Moreover, integrating *power-gating designs* to turn off partsof the circuit during idle times could lead to further reductionsin power consumption.

3) Exploration of New Filter Types: Although the project focuses on Gaussian filters, there are many other filter types used in image processing. Future improvements could include expanding the use of approximate adders to filters like Sobel, Laplacian, or median filters. This would broaden the applicability of the power-saving techniques demonstrated in this work.

4) Advanced Error Mitigation Techniques:

Approximate computing introduces errors; thus, future research could focus on developing new *error control mechanisms*. These could dynamically adjust the acceptable error threshold based on real-time application needs. Moreover, introducing *perception- based error metrics* could ensure that the errors introduced by approximation remain imperceptible to human users in certainapplications.

5) Integration with Machine Learning:

Machine learning techniques can be integrated into future versions of this system to optimize approximation dynamically. By training models to predict which areas of an image can tolerate approximation, machine learning-based optimization can lead to further energy savings without sacrificing significant image quality. Approximate adders could also be used to accelerate convolution operations in neural networks, making them suitable for low-power AI tasks.

6) *Energy-Aware Scheduling:*

Future systems could incorporate energy-aware task scheduling algorithms that allocate approximate computing resources based on energy availabilityand real-time processing needs. This would enable dynamic power management in variable environments, such as solar- powered edge devices.

Extending to Video Processing:

Another area for im- provement is extending the use of approximate adders to video processing tasks. The system could be adapted to process video streams in real-time, applying noise reduction and filtering techniques across frames while minimizing power consumption. Temporal filtering could also be explored, allow- ing the system to balance power and image quality across bothspatial and temporal dimensions.

VI .CONCLUSION

In this work, we explored power-efficient Gaussian filter architectures using Design Space Exploration (DSE) techniques and logical optimizations. By incorporating approximated adders, we were able to reduce power consumption while maintaining acceptable filter quality. Future work will focus on extending these techniques to other filter types and exploring further optimizations in approximate computing.

REFERENCES

[1] N. K. Verma and A. Tiwari, “A Design Space Exploration of Power- efficient Gaussian Filter Architectures,” *IEEE Latin American Sympo- sium on Circuits and Systems (LASCAS)*, 2022. [Online]. Available: https://doi.org/10.1109/LASCAS53948.2022.9789080

[2] J. K. Ramakrishnan, “Power-quality Pareto front in approxi- mated Gaussian filter design,” *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021. [Online]. Available: http://dx.doi.org/10.1109/ISCAS51556.2021.9401199

[3] A. Joginipelly, A. Varela, D. Charalampidis, R. Schott, and Z. Fitzsimmons, ”Efficient FPGA implementation of steerable Gaussian smoothers,” in SSST’12, 2012, pp. 78–82.

[4] S. Kaushik and N. V. Kumar, ”Energy-efficient approximate 2D Gaus- sian smoothing filter for error tolerant applications,” in IACC’15, 2015, pp. 789–794.

[5] R. Oliveira Julio, L. B. Soares, E. A. C. Costa, and S. Bampi, ”Energy- efficient Gaussian filter for image processing using approximate adder circuits,” in ICECS’15, 2015, pp. 450–453.

[6] L. B. Soares, S. Bampi, and E. Costa, ”Approximate adder synthesis for area- and energy-efficient FIR filters in CMOS VLSI,” in NEWCAS’15. IEEE, Jun. 2015, pp. 1–4. Available: http://dx.doi.org/10.1109/NEWCAS.2015.7182095

[7] P. Stanley-Marbell, A. Alaghi, M. Carbin, et al., ”Exploiting errors for efficiency: A survey from circuits to applications,” ACM Comput. Surv.,vol. 53, no. 3, Jun. 2020.

[8] A. M. Hambal and Z. Pei, ”Image noise reduction and filtering tech- niques,” IJSR, vol. 6, no. 3, pp. 2033–2038, Mar. 2017.

[9] C.-M. Kyung, ”Theory and applications of smart cameras,” KAISTResearch Series, 2016.

[10] B. Koyada, N. Meghana, M. O. Jaleel, and P. R. Jeripotula, ”A comparative study on adders,” in WiSPNET’17, 2017, pp. 2226–2230.

[11] J. de Oliveira, L. Soares, E. Costa, and S. Bampi, ”Exploiting approxi- mate adder circuits for power-efficient Gaussian and Gradient filters for Canny edge detector algorithm,” in LASCAS’16, 2016, pp. 379–382.

[12] L. B. Soares, S. Bampi, and E. Costa, ”Approximate adder synthesis for area- and energy-efficient FIR filters in CMOS VLSI,” in NEWCAS’15. IEEE, Jun. 2015, pp. 1–4. Available: http://dx.doi.org/10.1109/NEWCAS.2015.7182095

[13] A. Meurer et al., ”Sympy: symbolic computing in python,” PeerJ Computer Science, vol. 3, p. e103, Jan. 2017. Available: https://doi.org/10.7717/peerj-cs.103

[14] H. Jiang et al., ”Approximate arithmetic circuits: A survey, characterization, and recent applications,” Proceedings of the IEEE, vol. 108, no. 12, pp. 2108–2135, 2020. Available: https://doi.org/10.1109/JPROC.2020.3006451

[15] Synopsys, Synopsys Design Compiler, V. S-2021.06-SP5, 2021.

[16] D. Martin et al., ”A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in IPCV’01, vol. 2. IEEE, Jul. 2001, pp. 416–423. Available: https://doi.org/10.1109/ICCV.2001.937655

[17] A. Hore´ and D. Ziou, ”Image quality metrics: PSNR vs. SSIM,” in ICPR’10. IEEE, 2010, pp. 2366–2369. Available:
https://doi.org/10.1109/ICPR.2010.579
[18] J. Lee, H. Tang, and J. Park, ”Energy efficient Canny edge detector for advanced mobile vision applications,” IEEE Trans. Circuits Syst. Video Technol., vol. 28, no. 4, pp. 1037–1046, 2018. Available:
https://doi.org/10.1109/TCSVT.2016.2640038
[19] M. Hill, ”OpenALPR,” https://github.com/openalpr, 2022.
[20] J. Canny, ”A computational approach to edge detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-8, no. 6, pp. 679–698, 1986. Available: https://doi.org/10.1109/TPAMI.1986.4767851
[21] M. Kalbasi and H. Nikmehr, ”Noise-robust, reconfigurable Canny edge detection and its hardware realization,” IEEE Access, vol. 8, pp. 39934–39945, 2020. Available:
https://doi.org/10.1109/ACCESS.2020.297686