SBPG: Secure Better Portable Graphics for Trustworthy Media Communications in the IoT (Invited Paper)

Saraju P. Mohanty, Senior Member, IEEE, Elias Kougianos, Senior Member, IEEE, and Pathrasarathy Guturu, Senior Member, IEEE

***Abstract*—Smart Healthcare is envisioned as the combination of traditional healthcare augmented by smart bio-sensors, wearable devices and a plethora of on-body sensors that communicate with smart hospitals, smart emergency response systems, and ambulances, through advanced information and communication technologies. The vision of smart healthcare as part of a smart city relies on the framework of the Internet of Things (IoT) as the underlying core technology that enables the design and operation of a city, whereby smart technology, energy grids, transportation, buildings, communication, and information technology, are all interconnected. The present paper address some of the challenges faced in the IoT infrastructure, specifically secure communication and user authentication in the context of automated analysis of biomedical images and communication of the analysis results and related metadata in a smart healthcare framework. A hardware architecture for a Secure Digital Camera (SDC) integrated with the Secure Better Portable Graphics (SBPG) compression algorithm, suitable for applications in the IoT, is proposed in this paper. The focus of this work is on patient data protection and authentication. The proposed SBPG architecture offers two layers of protection: concurrent encryption and watermarking which address all issues related to security, privacy, and digital rights management (DRM). The experimental results demonstrate that the new compression technique BPG outperforms JPEG in terms of compression quality and compressed file size while providing increased image quality. High performance requirements of BPG have been met by employing two techniques: (1) insertion of an encrypted signature in the center portion of the image, and (2) frequency domain watermarking using block-wise DCT of size 8×8 pixels. These approaches optimize the proposed architecture by decreasing computational complexity while maintaining strong protection, with concomitant increase of the speed of the watermarking and compression processes. A Simulink® prototype for the proposed architecture has been built and tested. *To the best of the authors' knowledge, the hardware architecture for BPG compression with built-in image authentication capability for integration with a secure digital camera is the first one ever proposed*.**

***Index Terms*—Smart Healthcare, Internet of Things (IoT), Consumer Electronics, Secure Digital Camera, Image Communications, Better Portable Graphics**

I. Introduction

Smart Healthcare (SHC) is envisioned as a vital component of smart cities. The Internet of Things (IoT) is the core technology on which smart cities rely, whereby smart technology, energy grids, transportation, buildings, communication, and information technology, are all interconnected [1], [2]. Smart Healthcare is conceptualized as the combination of traditional healthcare augmented by smart bio-sensors, wearable devices and a plethora of on-body sensors that communicate with smart hospitals, smart emergency response systems, and smart vehicles, including ambulances, through advanced Information and Communication Technologies (ICTs) [3], [2].

Healthcare represents one of the most attractive application areas for the IoT, as depicted in Fig. 1 [2]. To provide timely real-time patient information to health support staff, which may reside in separate offices, hospitals or across several hospitals in the same city or at different cities, a smart hospital may employ cloud computing and big data analytics in conjunction with ICTs and smartphone apps. Similarly, telemedicine uses ICTs to deliver healthcare screening and even a diagnosis in remote and underserved areas, usually rural, as well as in developing countries whereby specialized expertise is not readily available. The IoT infrastructure relies on fast communication among the many different types of sensors, devices, and applications; on network security; and on the integrity of the software, middleware, and firmware algorithms that make the interconnected systems smart [3].

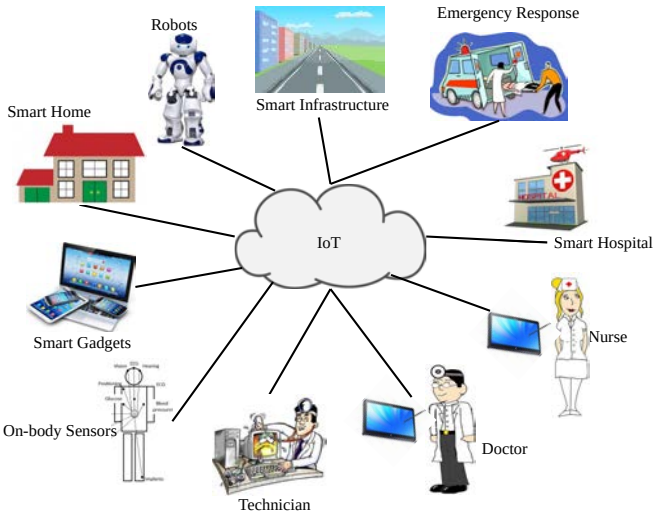

Fig. 1. The IoT is the Backbone of Smart Healthcare.

A useful implementation of the IoT in general, and of smart healthcare in particular, poses several challenges, including

S. P. Mohanty is with the Department of Computer Science and Engineering, University of North Texas, E-mail: saraju.mohanty@unt.edu. E. Kougianos is with the Department of Engineering Technology, University of North Texas, E-mail: elias.kougianos@unt.edu. P. Guturu is with the Department of Electrical Engineering, University of North Texas, E-mail: parthasarathy.guturu@unt.edu

energy efficiency of the components connected to the system; security of communications; unique identification of users that guarantees the privacy of the users information; computational performance, reliability, and flexibility of the system; efficient analysis of large datasets in real time; and processing and storing enormous amounts of data for follow-up studies and for knowledge discovery in meta-analysis stages [4]. These very important requirements of such systems impose significant challenges on the hardware design of electronic circuits and systems in terms of reliability and security. A secure framework should allow the unique identification of the originating site of data, guarantee the integrity of the data, safeguard against corruption, and should also set access privileges automatically to the rightful owners of the data for further offline analysis and patient follow ups.

One of the important requirements of SHC is to develop the capability for automated analyses of biomedical images and secure communication, with user authentication, of the results of those analyses and related metadata over an IoT infrastructure. This article addresses some of these challenges related to the fulfillment of this requirement. In this context, this research focuses on exploration of algorithms and architectures for Better Portable Graphics (BPG) [5], for energy-efficient, low-bitrate, and secure biomedical information exchange within the framework of SHC, and the realization of a Secure BPG (SBPG). More specifically, in this research watermarking methods are explored for use in architectures for trusted biomedical information exchange that is an integral part of an IoT. Additionally, we extend our initial work on BPG [6] and develop advanced energy-efficient real-time architectures using BPG. Medical images can be adapted to validate the security-integrated BPG methods, so that medical data can be securely stored and their integrity can be verified as they are transmitted over insecure Internet connections. Similar approaches can be considered with future expansion to non-image biomedical data.

The rest of the paper is organized as follows: In Section II, contributions of this paper are described. Section III presents related work in the field. In Section IV, a broad application perspective of a Secure Digital Camera (SDC) integrated with SBPG for IoT is presented. In Section V, the architectural overview of the SBPG integrated SDC is discussed. The SBPG algorithm and architecture are illustrated in Section VI, followed by experimental results in Section VII and a power optimization perspective in Section VIII. The conclusions are summarized in section IX.

II. Novel Contributions of this Paper

The IoT infrastructure relies on fast communication among the many different types of sensors, devices, and applications, on network security, and the integrity of the software, middleware, and firmware algorithms that make the systems smart. Thus, an efficient implementation of the IoT in general, and of SHC in particular, poses several challenges including energy efficiency of the components connected to the systems, security of communications, unique identification of users and privacy of the users information, computational performance, system reliability and flexibility, efficient analysis of large datasets in realtime, and handling and storing enormous amounts of data [7], [8]. More importantly, it imposes significant challenges on the hardware design of circuits and systems in terms of reliability and security. A few challenges of IoT-based smart healthcare are presented in Fig. 2 [3], [9].

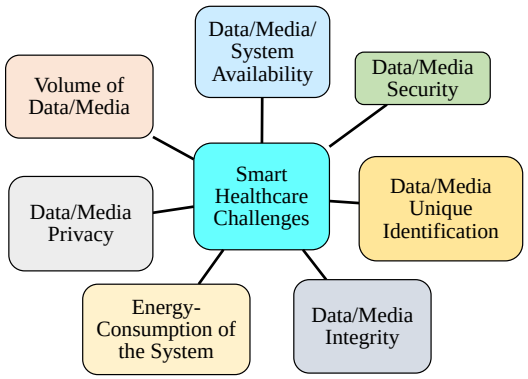

Fig. 2. Challenges in Smart Healthcare.

This paper describes a hardware architecture of the Secure Better Portable Graphics (SBPG) compression encoder that could easily be integrated with a Secure Digital Camera (SDC). The SDC is a secure and novel device for capturing digital images. In contrast to the standard Digital Camera (DC) capable of capturing only still images in a digital form and maintaining a visual record of events, the SDC is equipped with additional security-related features. For example, a DC does not have the capability to track the source, authenticity, and succession of custody for the images. Partial alleviation of these shortcomings can be provided by digital watermarking but it does not provide indisputable image authentication. The unique components of SDC help identification of the user, corroboration of image fidelity, and maintenance of detailed records of time and other pertinent data [10] related to succession of custody of the images.

To meet the rapidly evolving requirements of smaller image size without loss of quality, the proposed architecture uses the recently developed BPG compression [5] algorithm. As far as the authors' knowledge goes, this is the first ever prototype for a hardware architecture for an SDC with an integrated SBPG compression encoder and analysis of the architecture.

The main contributions of this work are:

- An SDC equipped with a novel SBPG compression technique that facilitates real-time, high-quality, and low-size imaging. This is a novel use of the SDC to address the numerous privacy and security issues associated with transmission of biomedical images over the IoT.
- Innovative hardware architecture for SBPG compression that is integrated with the SDC, thus forming an integral system, optimized for security and trust with minimal energy consumption.
- Hardware design space exploration with Simulink®-based prototype of the algorithm, which allows for exhaustive analysis of trade-offs and what-if scenarios, before the design is implemented in silicon.

- Experimentation for a comparative analysis of the architecture with existing standards, which demonstrates the superior performance of SBPG and its suitability for next-generation IoT applications.

III. Related Prior Research

In this section, related prior research in Secure Digital Camera design and a specific application, smart traffic surveillance, are discussed. In [10], the authors have demonstrated a unique approach towards an SDC having two layers of protection with seamless integration of watermarking and data encryption capabilities. Steganography of binary image data and its secure authentication is handled by the proposed architecture. A method for field programmable gate array (FPGA) and System-on-Chip (SoC) implementation is part of this research. In [11], the authors demonstrate a hardware capable of an invisible watermark embedding with the LeGall 5/3 Discrete Wavelet Transform (DWT). The suggested structural design addresses the limitations of standard digital cameras. The algorithm was assessed with JPEG compression and, for the VLSI implementation of the processor, Verilog was recommended. To support ownership rights of pictures and illustrations captured by digital cameras, a novel scheme is introduced in [12]. The proposed two-fold technique combines semi-brittle and vigorous blind type watermarks. Development of the watermark in this work is done using the rate of information recurrence and the camera owner's biometric records. Two digital image watermarking methods have been put into practice by an innovative VLSI based approach presented in [13]. The VLSI approach enables integration of these methods with any available digital camera structure. The proposed design involves a sequence of steps for the watermark construction on a pixel-by-pixel basis based on the signal-to-noise ratio information.

In [14], a camera called TrustCAM is proposed for integrity protection, authenticity and confidentiality of image data. In [15], a Physical Unclonable Function (PUF) based trusted sensor is proposed and the corresponding camera has been prototyped for sensed data attestation. In [16], the design of smart cameras is presented with a concept called Signcryption for resource-efficient data signing and encryption in a single step. In [17], the authors present a novel camera prototype for smart traffic surveillance. The architecture of the camera is based on CMOS sensors, digital signal routing and a complex system CPU. The uniqueness of this approach lies in its ability to detect motionless vehicles automatically. Further, by using the proposed model, detecting traffic jams in real time could become possible. In [18], the authors investigate the use of camera-video-surveillance capabilities for distinguishing moving vehicles in varying and diverse street settings. The authors have considered a functional encoding of OpenCV by integrating several operating systems to assess the outcomes, mainly GNU Linux. The study in [19] presents an original approach to a camera with the capability to transmit and receive images in a multitude of directions when used along with pan-tilt-zoom (PTZ). This distinct capability makes it easier for the tracker to follow irregular activities. The attributes of omni cameras identified included coarse categorization, refined classifications and generation of long-term statistics.

Effectiveness of an embedded smart front-end camera for implementing smart and sophisticated traffic surveillance algorithms has ben studied in [20]. The camera could reach an approximate performance of seventeen frames per second. The proposed system has taken into account major issues faced in traffic networks like the need to reduce system bandwidth for video streaming. It is unique due to its feature of recording unanticipated traffic events autonomously. For real-time monitoring of the traffic from a motionless camera, a novel algorithm called the Scale Adaptive Object Tracking (SAOT) is proposed in [21]. This method uses the correlation between the past and new traffic information for continuously resolving traffic drift issues.

IV. SDC-SBPG Integration for IoT: A Broad Application Perspective

The SDC is a device that augments standard features of a digital camera with additional built-in facilities for real-time performance, and low-cost and low-power operation [22], [23]. In this work, a novel encoder with built-in watermarking and encryption facility called secure Better Portable Graphics (SBPG) is presented. Compared to JPEG, BPG compression offers better compression along with high quality of decompressed images, and this makes it uniquely suitable for real time and bandwidth-restricted environments. The SDC integrated with SBPG is typically implemented as an SoC. Using watermarking and encryption as two layers of protection, the SDC addresses many Digital Rights Management (DRM) issues including rights of ownership, tracking of usage, detection of tampering, and authentication of content. These security features together with real-time performance make the SDC integrated with SBPG immensely suited for smart applications such as the IoT based SHC, as shown in Fig. 3.

The IoT is an ensemble of innumerable smart mobile devices, smart homes, and embedded applications connected to the Internet for integration of greater computational capabilities with the usage of analytics to extract and act on relevant information. The Internet is currently connected to billions of devices, and soon this number is expected to grow to hundreds of billions or even trillions. Related devices communicating with each other via the Internet form an intelligent system of systems or a meta-system. These meta-systems share data over the cloud and analyze it, and thereby transform the world, our lives, and the way we conduct business, in countless ways. Improvement of medical care, faster production of better products with lower development costs, more enjoyable shopping, and optimization energy production and consumption are but a few examples of drastic improvement in our quality life due to this transformation. In the current article, the initial concentration is on image data because of easy access and ease of manipulation as depicted in Fig. 3. At a later part of the research more relevant biomedical information, such as EEG and EKG signals, basal body temperature, pulse and respiration data, etc., will be considered. The basic functions of the framework, namely PUF-based hardware

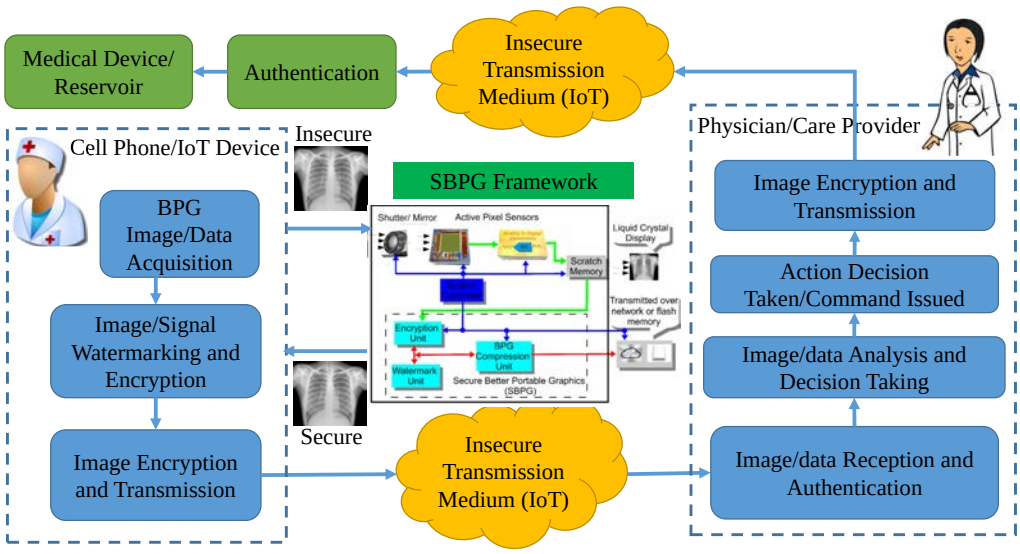

Fig. 3. Benefits of Introducing the IoT in the SHC system.

security/encryption and watermarking, remain the same with minor adaptations to each form of data. A potential typical sequence of events that will trigger the framework function is as follows: (1) An event (epileptic seizure, heart attack, fall, etc.) triggers automatically the IoT-enabled device on the patient. The patient may be incapacitated and not able to direct further actions or provide authentication. (2) The generated data are locally watermarked to evaluate later if they have been tampered. (3) The watermarked data are encrypted with a key uniquely determined by the device (a PUF) to guarantee their point of origin. The key is derived purely from stochastic characteristics of the device and is typically unknown not only to attackers but even the manufacturer of the device. (4) The data travel through the IoT and are received by the responder. (5) The data are decrypted via asymmetric key cryptography, the point of origin is confirmed, and the watermark quality evaluated to ensure that the data have not been tampered with. This step takes place very quickly, typically in fractions of a second. (6) Based on the trustworthy data, the provider makes a decision, and potentially activates remotely a device (e.g. medicine reservoir). (7) Action instructions are locally encrypted with a different unique key and transmitted through the IoT. (8) These instructions are received by the device, decrypted, and the point of origin confirmed, again within fractions of a second. (9) The device takes the appropriate actions and continues transmitting biomedical information to the provider.

V. Architectural Overview of SBPG Integrated SDC

The SDC offers new trends for scrutiny and inspection of real-time image-related patient data. Health providers are therefore able to track the record of any patient easily anytime, anywhere, securely and with complete privacy protection. The digital images and other data captured through the SDC can help in efficient patient monitoring by identifying significant events automatically and triggering appropriate alarms. Upon examination of the basic attributes and significance of the SDC, it can be stated that it would be an effective component for the development of SHC. With its secure and smart features, the SDC can transform existing health monitoring systems to an intelligent and automated system. Such an intelligent system would be capable of processing, analyzing, storing and communicating information over the network autonomously without any human support [24]. Based on the aforementioned facts and findings, it can be presumed that the SDC offers a new dimension to integrate the IoT (Internet of Things) within the existing healthcare system. Its potentials of error minimization and integrity of digital evidence can further help in transforming the traditional healthcare system to a true SHC system.

Fig. 4 presents an architectural overview of the proposed SBPG integrated SDC with a system-level block diagram of the SDC. The proposed architecture enhances a standard digital camera with additional built-in capabilities for watermarking, encryption and BPG compression. the building blocks of this system are an active pixel sensor (APS) unit, a liquid crystal display, analog-to-digital converters, and encryption, watermarking, and compression units. Images captured by an image sensor are converted into digital signals, and stored temporarily in scratch memory. Further processing of the images is done by the SBPG module. The entire sequence of the events is handled by the controller unit. In the system-on-a-chip (SoC) implementation of the proposed architecture for SDC, three independent modules for data encryption, watermarking, and compression work in a coordinated manner.

An invisible robust blind watermarking approach is used in conjunction with the Rijndael advanced encryption standard (AES) [25] in the proposed SBPG module,. The SBPG module performs the BPG compression, which, as mentioned earlier, yields higher compression ratio compared JPEG images without compromising the decompressed image quality. BPG compression is based on High Efficiency Video Coding (HEVC), which is considered a major advance in compression

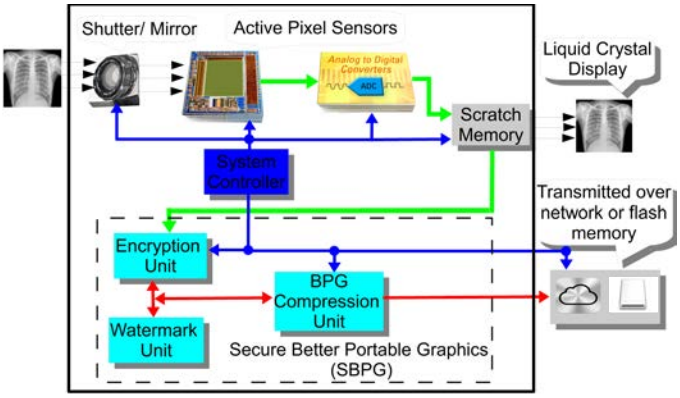

Fig. 4. System-Level Block Diagram of SBPG Integrated With SDC.

techniques. The two layers of protection in the proposed architecture are AES encryption and invisible robust blind watermarking. Two layers of protection are required because all issues related to DRM cannot be addressed by either the encryption or watermarking algorithms alone. For example, an encryption algorithm prevents unauthorized access of the digital content, but not illegal replication of the decrypted content by an unauthorized user. The latter issue can be addressed by digital watermarking. Establishments of ownership rights by protection of the images against false ownership claims based on unauthorized modification and illegal replication is the main strength of the Digital watermarking algorithms. Hence, conjoint use of encryption and watermarking algorithms provides full protection: confidentiality and data integrity. Fig. 5 depicts the various processing stages of the SBPG integrated with SDC. Performing watermarking and data encryption prior to BPG compression is more secure approach compared to the reverse approach because watermarking after compression means watermarking based on changed information of the host image.

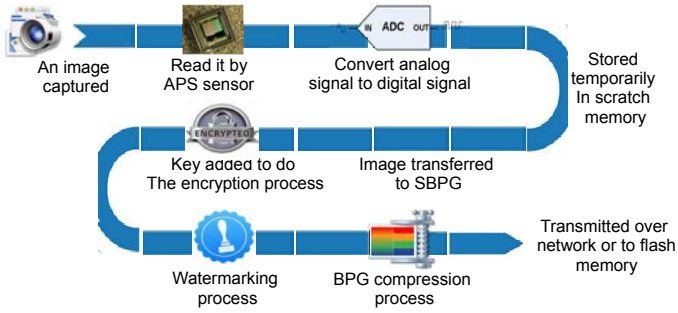

Fig. 5. Processing Stages of the SBPG Integrated With SDC.

VI. SBPG: Algorithm and Architecture

A. Algorithm and Architecture of Encryption and Watermark Unit

AES is considered the most common block cipher, as it supports a variable data block and a variable key length. The proposed encryption and watermarking unit in the context of SBPG operation flow is shown in Fig. 6. The block and key size can be any multiple of 32 bits, between 128 and 256. The main reason for the choice of the AES algorithm is due to the fact that there are no practical known attacks against AES. In addition, AES can be optimized in VLSI architectures to achieve high-performance and high throughput with area efficient implementations [26]. AES consists of four fundamental algebraic function transformations: byte substitution, shift row, mix column, and round key for each standard round with the exception that the last standard round does not perform mix column. Based on the block and key size, 128, 192, or 256, the number of standard rounds will be 10, 12, or 14, respectively.

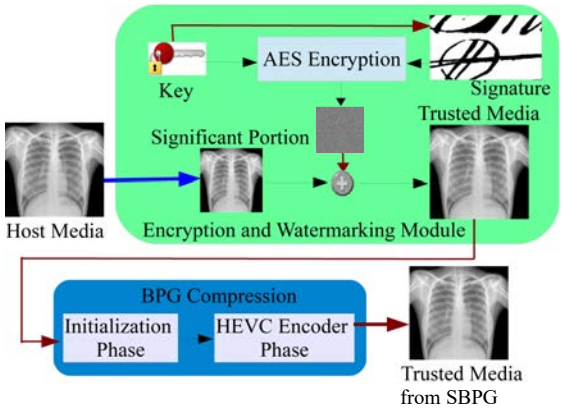

Fig. 6. Block-Level Overview of the Proposed SBPG Module.

In general terms, digital watermarking is a method in which information regarding an image and its owner is embedded in the image itself in such a way that unauthorized users are not able to recover an original copy of the image, while an authorized user can, after suitable processing of the image. Depending upon the application requirements such as speed, cost, and reliability, different types of watermarking can be employed. The invisible-robust-blind watermarking algorithm features chosen for this application are summarized as follows:

1) *Robustness, quality, and computational load* are optimized due to watermark embedding in the center portion of the image containing the main information of the scene. Additionally, encryption and watermark insertion at the quarter- sized sub-image at the center provides robustness against violation of DRM rights because attempts to remove the watermark will result in significant image quality degradation.
2) The *watermark processing* is done in the frequency domain using the Discrete Cosine Transform (DCT) thus increasing the speed of watermark insertion. Additionally, the *computational speed* is increased by selecting an 8×8 DCT block size .
3) *Watermark insertion* is done in the mid-frequency range of the image increases robustness since removal of mid-frequencies rather than the high or low frequency components of the watermarked image significantly affects the watermark as noticed by the human eye.

The watermark insertion is performed on the color image, which is assumed of size $M \times N$ based on the procedure proposed in [27], [28]. As the first step of the insertion process,

the image is transformed from the RBG space to YCbCr color space. From this point on, the Y-color component is considered for further processing. The Y component is divided into 8×8 blocks and DCT is performed on each block. Since the center portion of the image is the focus of attention from the viewer's point of view, the watermark is embedded in the center quarter of the image. By only using 25% of the image area for watermarking, computational efficiency as well as overall image quality are increased. Furthermore, if an attacker attempts to destroy or remove the central watermark, image quality will be substantially degraded. The selection of appropriate DCT coefficients for the watermarking process is of fundamental importance. Selecting low frequency components causes degraded image quality due to their high information content. On the other hand, insertion in the high frequency components makes the watermark susceptible to removal through low-pass filters. Therefore, in this application the mid-frequency blocks are selected for watermark insertion. Four mid frequency coefficients are chosen from each block in the center quarter of the image. From these coefficients, a vector R of size K is generated, as given in equation 1, where K is the number of blocks in the center quarter of the image, and $r_{x,y}$ is the coefficient of the selected block y. A pseudo random sequence is chosen from bits in the encrypted signature, which is used as the watermark represented in equation 2. The watermark is inserted into the DCT coefficients of the image from which vector R is extracted according to equation 3:

$$R = \{r_{1,i}, r_{2,i}, r_{3,i}, r_{4,i}, ..., r_{1,K}, r_{2,K}, r_{3,K}, ..., r_{4,K}\} \quad (1)$$

$$A = \{a_1, a_2, a_3, ..., a_{4\times K}\} \quad (2)$$

$$r_i^{'} = r_i + \alpha \mid r_i \mid a_i, \quad (3)$$

where $i = 1, 2, \cdots, 4 \times K$ and α is a scaling constant determining the watermark strength. Small values of α make the watermark vulnerable to modification and also make its extraction and detection difficult. Large values of α can render the watermark visible. An optimal choice of α is necessary, as discussed in [29]. In this work the focus is on high performance with good quality hence a value of α = 0.5 is chosen. The complete insertion process is shown in Fig. 7. For brevity, only the insertion process is discussed here as the extraction process is its exact reverse.

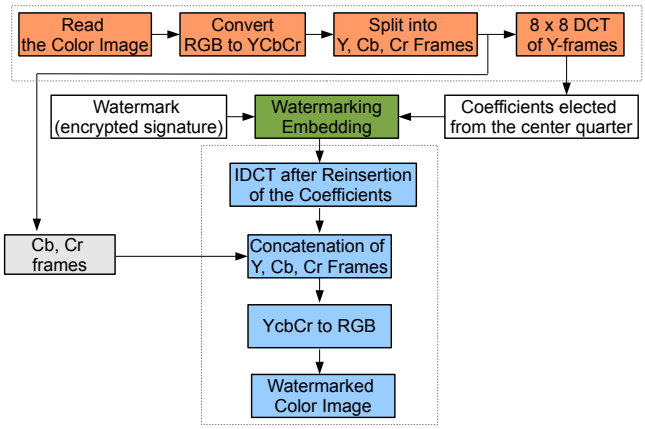

Fig. 7. Watermark Insertion Algorithm.

B. Algorithm and Architecture of the SBPG Compression Unit

Higher compression ratio compared to JPEG without affecting image quality is achieved by BPG. BPG supports, animation, lossless compression, various color spaces (grayscale, YCbCr, RGB, YCqCo), and chroma formats [6], [30], [5]. The success of SBPG relies on the capability of BPG to generated high-quality, small-size images. Preliminary results by the authors of this work [6], [30] demonstrate that the algorithm is very promising in terms of quality and compression ratio. Fig. 8 presents an overview of the simplified BPG processing flow to be investigated. The reference BPG image library and utilities (libbpg) can be divided into four functional modules: BPG encoder, BPG decoder, Javascript decoder, and BPG viewer. With JPEG or PNG images as input, the BPG encoder performs BPG compression to produce corresponding BPG-compressed images. The BPG decoder performs the decompression. With an embedded small Javascript decoder, most web browsers support the BPG format. Any program can view a BPG image with the help of a BGP viewer. The BPG encoder is based on HEVC encoding [31]. HEVC, also known as H.265, is considered as the prime candidate to replace the H.264 standard due to its compression efficiency [32]. HEVC aims at reducing the bitrate used in H.264/AVC because it is more parallel-friendly [33], [34].

Fig. 9 presents the BPG compression encoder flow. BPG encoding involves two main stages: the pre-encoding (initialization) stage and the encoding stage. In the initialization stage, meta data such as color space and bit depth is read. Bit depth is the number of bits used to indicate the color of each pixel [10]. Common bit depths are: 8, 10, 12, $\cdots$. Images with high bit depths have stricter data storage and transmission bandwidth requirements. Moreover, some displays are not capable of reproducing all these colors. The BPG compression encoder strictly considers images with bit depth of 8. In the second (encoding) stage, HEVC first reduces temporal and spatial redundancy of video data by inter and intra frame prediction and then, using an 8×8 block as the basic coding unit, transforms the image data to frequency domain with the help either the Discrete Cosine Transform (DCT) or the Discrete Sine Transform (DST). Standard sizes of the transform, which is usually the DCT, are 4×4, 8×8, 16×16, and 32×32. The transformed image is quantized and additionally compressed using entropy coding to eliminate redundancies not removed by the aforementioned prediction mechanisms. The reconstruction core constructs a decoded reference frame as seen by a decoder every time a frame is encoded [35] during the processing of a video. In this BPG application, it remains idle. The bitstream core further compresses the image via entropy coding by the use of context-adaptive binary arithmetic coding (CABAC). A hardware architecture for energy-efficient encoding is shown in Fig. 10 [6]

The controller unit is responsible for coordination of the entire process; it is modeled as a finite state machine (FSM) with fifteen states: `init, S0 ··· S13`, as shown in Fig. 11. The `init` state checks the watermark signal: if it is 0, it infers

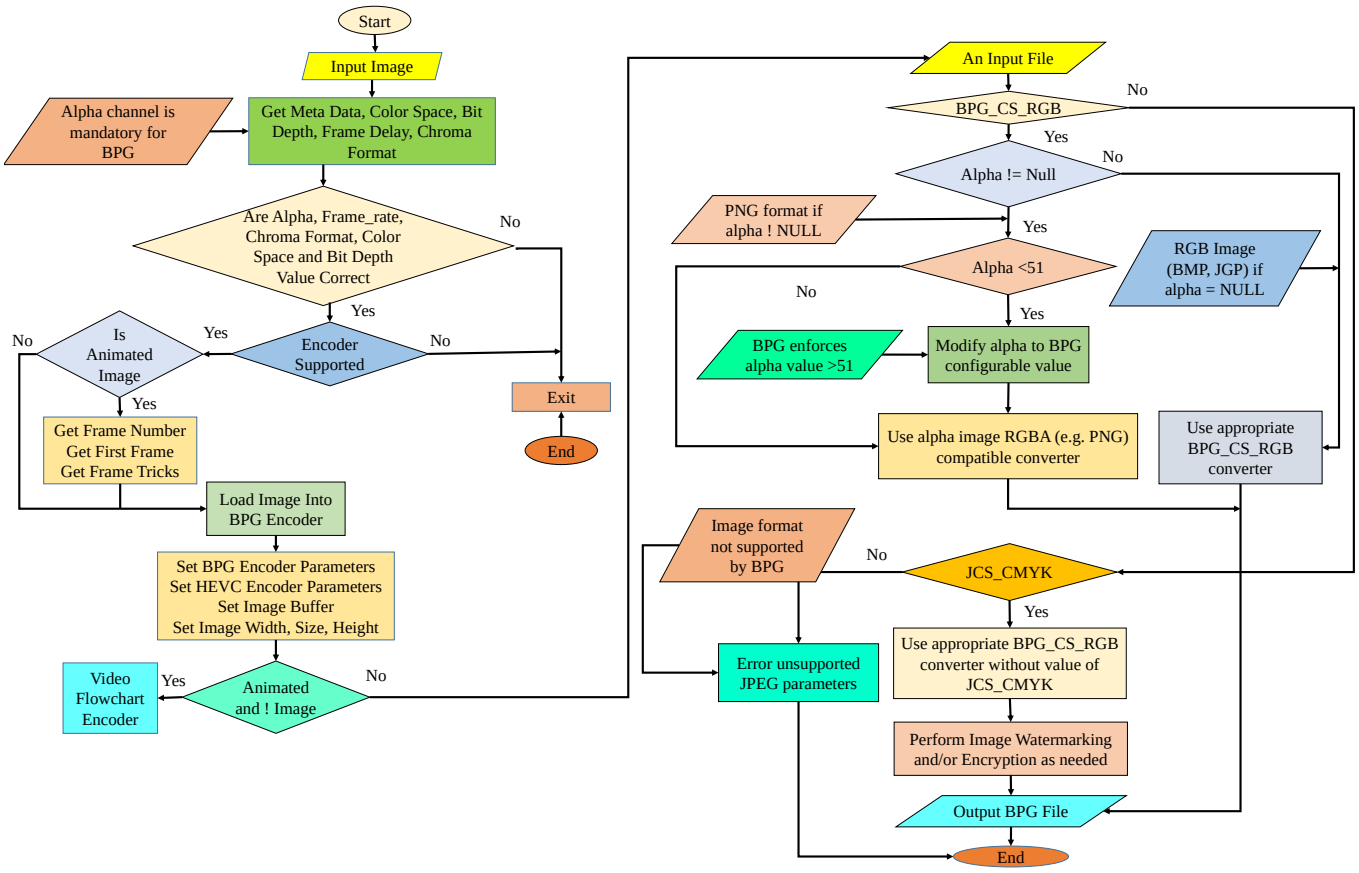

Fig. 8. Flowchart for Secure Better Portable Graphics (SBPG) compression.

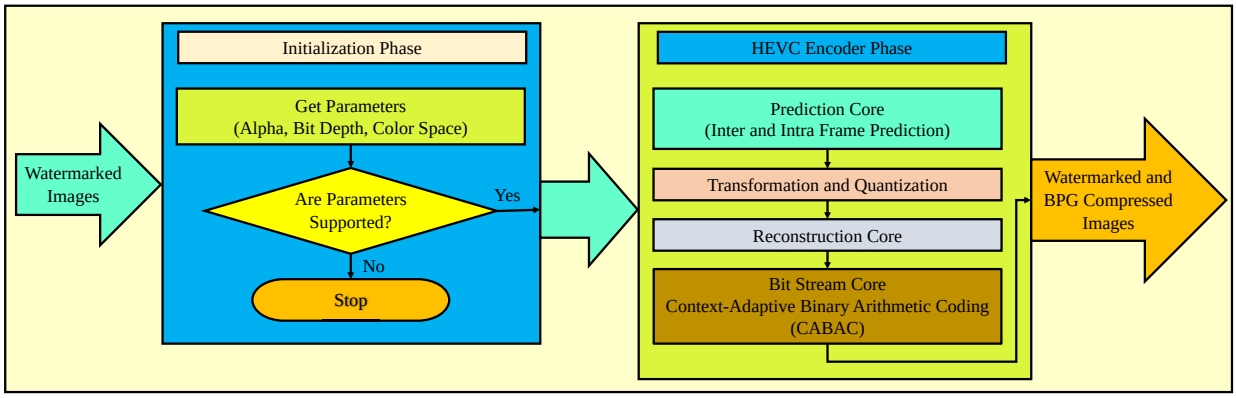

Fig. 9. BPG Compression Encoder Flow.

that the watermarking process has not yet been performed. In this case the transition `init` → `S0` takes place. After reading of the pixel data in `S0` state, the controller transitions to state `S2`, where DCT is performed. If all DCT coefficients of a block is have not been calculated, the transition `S2` → `S0` takes place. After the completion of all blocks, transition `S2` → `S3` occurs to encrypt the signature. The determination of the perceptually important center quarter of the image is done in state `S4`. The actual watermarking takes place in states `S4` and `S5`. Subsequently, the watermark is written to RAM. When all the coefficients of the blocks are watermarked, a transition occurs to the `init` state and the watermark signal is changed to 1 which causes the transition `init` → `S8` where the image parameters are checked. If they are compatible for BPG encoding, image splitting is performed in state `S9`, otherwise the system resets back to the `init` state. In state `S10`, temporal and spatial redundancy are reduced. In state `S11` quantization of the coefficients takes place. For video, reference frames are constructed in state `S12`. Finally, CABAC is performed in state `S13`, which completes the BPG compression of the watermarked image.

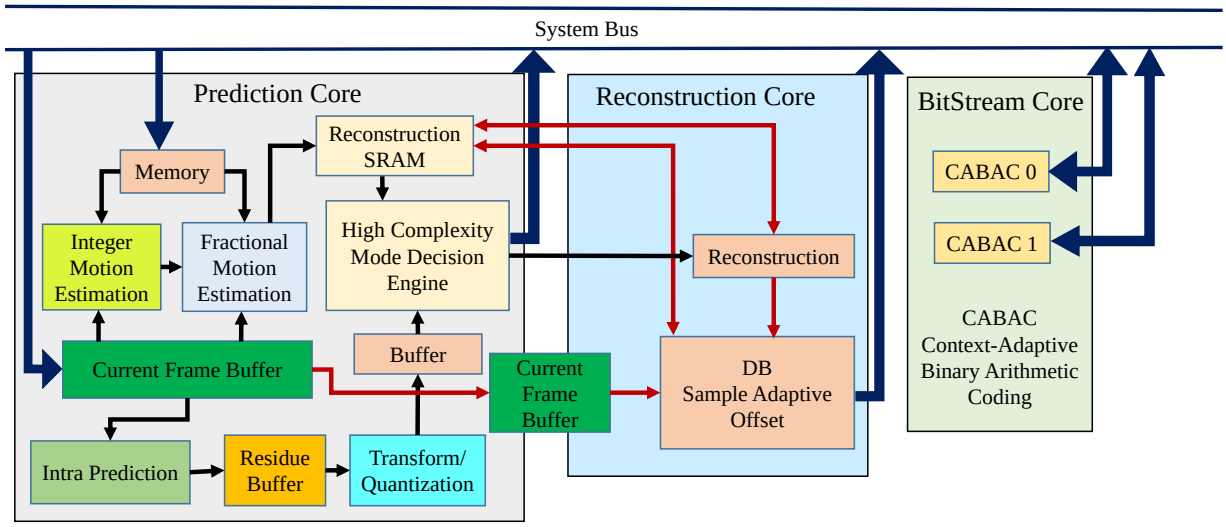

Fig. 10. HEVC Encoder Architecture used in BPG.

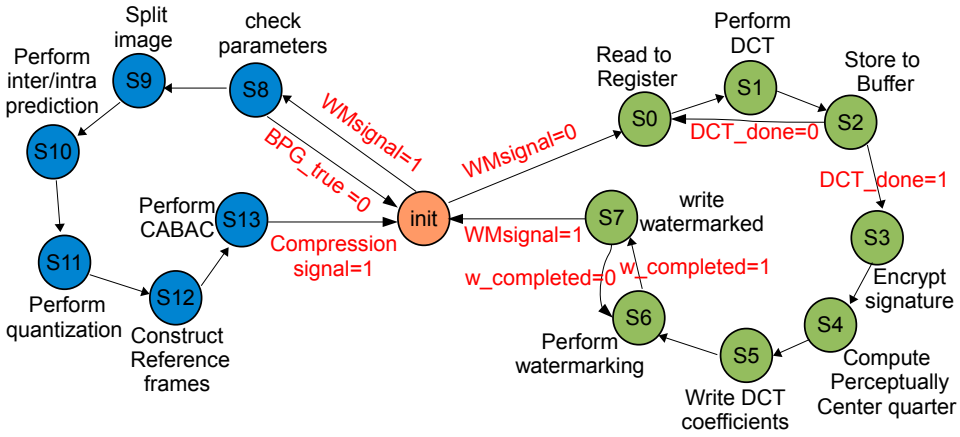

Fig. 11. Finite State Machine for the SBPG Controller

VII. Experimental Results

Implementation of the SBPG architecture is done in MATLAB®/Simulink® with the Computer Vision Toolbox [36]. Use of MATLAB® provides flexibility for the low-level implementation whereas Simulink® model provides a top-level functional and dataflow visualization.

A. Simulink Modeling of the SBPG Encoder

The SBPG encoder model developed using a bottom-up modeling methodology is shown in Fig. 12. Initially the encryption, watermarking, and BPG units are modeled and then integrated into sub-systems. Verification and testing of the overall system functionality is performed on this composite SBPG encoder model. Experimental results of the extensive testing phase for several test images are shown in this section.

B. Watermark Insertion and Image Compression using the SBPG Encoder

Out of a large data set from the Joint Picture Expert Graphics (JPEG) Group, five standard images with different spatial and frequency characteristics are selected for experimentation with SBPG. The cover image, watermarked image, and corresponding BPG image are shown in Fig. 13, Fig. 14, Fig. 15, Fig. 16, and Fig. 17.

C. Graphs of RMSE and PSNR to Measure Quality Assurance

The Root Mean Square Error (RMSE) [37] given in equation 4, and the Peak Signal-to-Noise Ratio (PSNR) [38] given in equation 5 are the two figures of merit calculated to measure the robustness and strength of the watermarked images. The RMSE is used twice: first to compare the watermarked image $O^{'}$ with the original image O of size $M \times N$, and a second time to compare the watermarked compressed image $O^{'}$ with the watermarked image O. The same process is applied to PSNR as well.

$$RMSE = \frac{1}{\sqrt{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N=1} ||(O(i,j) - O'(i,j)||^2 \quad (4)$$

$$PSNR = 20 \log \left(\frac{Emax}{MSE} \right) \quad (5)$$

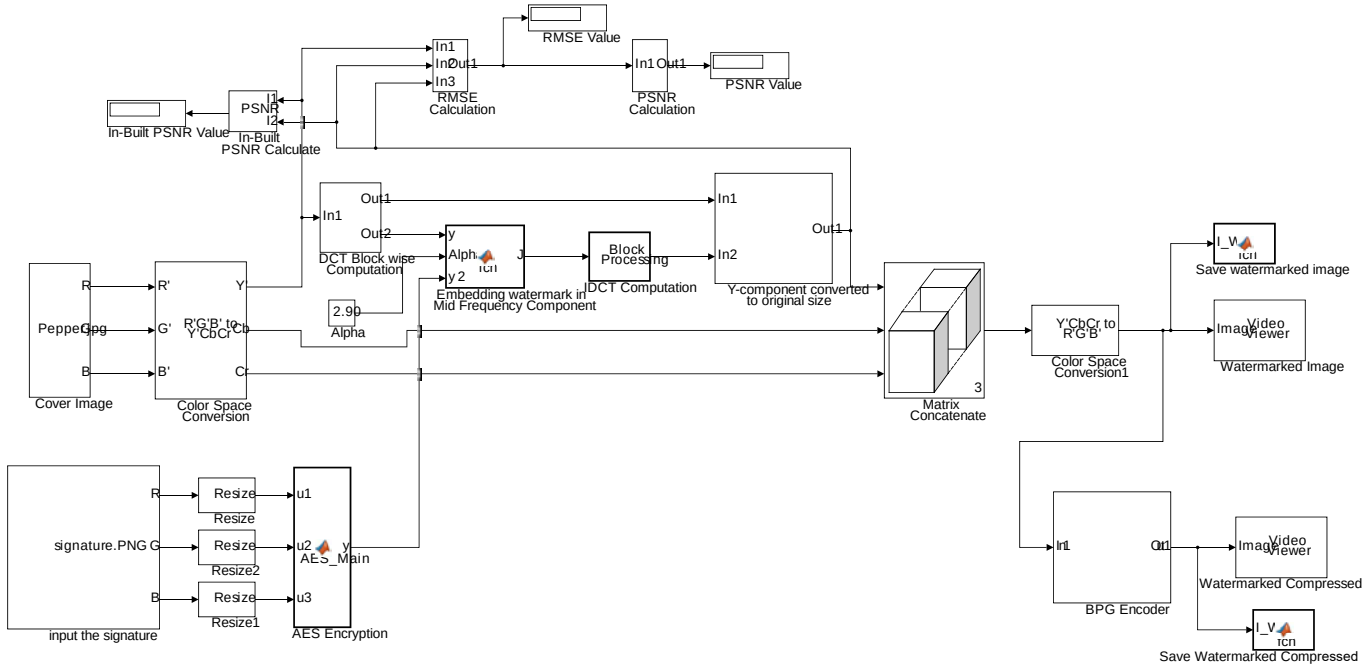

Fig. 12. SBPG Compression Encoder in Simulink®.

(a) cover Image. (b) Watermarked Image (c) Watermarked Compressed Image

Fig. 13. Secure BPG Compression of Baboon Image (256×256).

(a) Cover Image. (b) Watermarked Image (c) Watermarked Compressed Image

Fig. 14. Secure BPG Compression of Forest Image (256×256).

(a) Cover Image. (b) Watermarked Image (c) Watermarked Compressed Image

Fig. 15. Secure BPG Compression of IClimbImage (512×512).

(a) Cover Image. (b) Watermarked Image (c) Watermarked Compressed Image

Fig. 16. Secure BPG Compression of Lena Image (512×512).

(a) Cover Image. (b) Watermarked Image (c) Watermarked Compressed Image

Fig. 17. Secure BPG Compression of PepperImage (512×512).

Relevant metrics for each test image are presented in Table I. In all the cases, a PSNR value above 47.9 dB is maintained. Improvement after decompression in the visual quality of the watermarked and compressed images with larger PSNR values confirms the strength of the proposed SBPG encoder in maintaining the quality of the watermarked images and making it impossible for the human eye to detect the signatures of the watermarks in the images. Higher values of PSNR demonstrate the quality of the images compressed with SBPG as well as the attack resilience and consequent robustness of the

algorithm. Fig. 18 and Fig. 19 show the graphs of PSNR and RMSE versus the sizes of the corresponding watermarked, and watermarked and compressed images for all the test images. Table I indicates that the PSNR value for the "Pepper" image is the highest at 55.4 dB, whereas it is the lowest for the "Forest" image . Fig. 20 and Fig. 21 show that the trend is reversed in the case of RMSE.

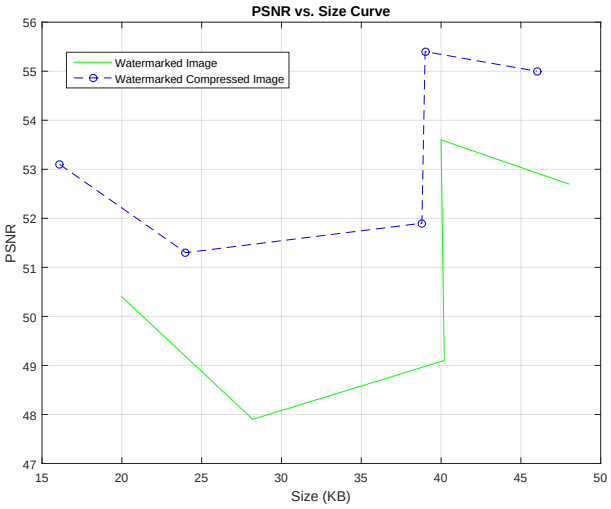

Fig. 18. PSNR vs Size Curve.

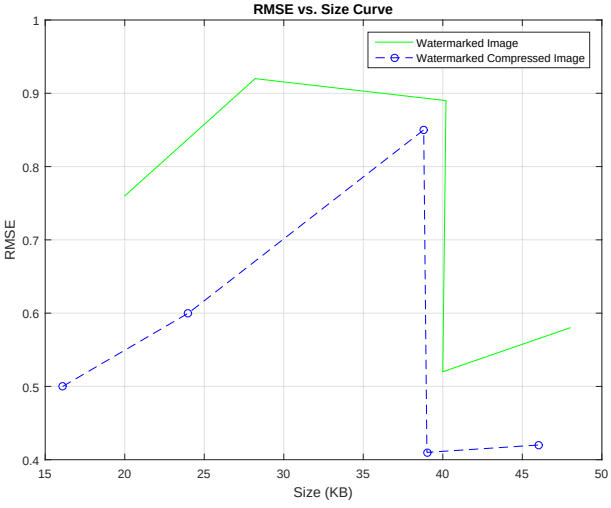

Fig. 19. RMSE vs Size Curve.

A comparison of this architecture vis-a-vis some prior works on secure digital camera is presented in Table IV. In this context, the superiority of the proposed SBPG should be emphasized.

D. Testing for High performance

Trade-offs between watermark quality and compression performance need to be compared. The results presented in Table I show a consistent maintenance of PSNR value above 47.9 dB in all test cases and thereby verify that the proposed SBPG architecture provides high quality watermarking along

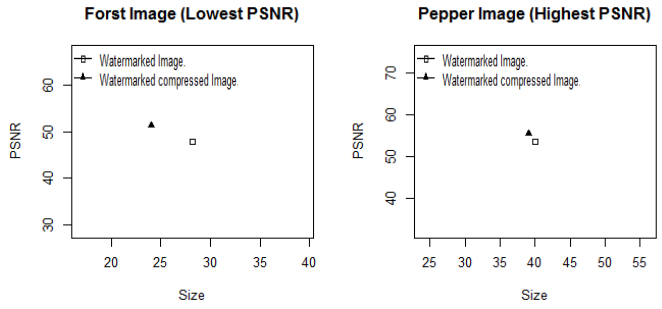

Fig. 20. Highest and Lowest value of PSNR.

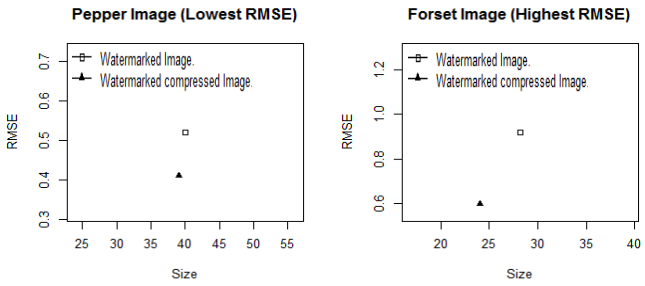

Fig. 21. Highest and Lowest value of RMSE.

with better compression compared to JPEG. The high performance architecture has been developed based on the following considerations:

1) Optimization of robustness, quality, and computational load by inserting watermarks exclusively at the center portion of the image, which typically is the repository of main information content. Consideration of a portion of the image rather than the whole image results in increased speed. Any attempt to tamper with the watermark co-located with the main image content at the central portion of the image will result in serious degradation of image quality. Thus this choice for watermark location provides attack resilience and hence robustness.
2) Watermark insertion speed is increased by performing the watermarking in the frequency domain using a block-wise Discrete Cosine Transform (DCT) of size 8×8.
3) With due consideration for extension of this architecture to video processing in the future, inter and intra frame prediction algorithms to reduce the temporal and spatial redundancy and improve thereby the computational speed, have been incorporated the proposed architecture.

When tested with 30 random images to obtain a rough estimate of the frame-rate for video applications, the Simulink® model produced the 30 watermarked compressed images in a time of 1.27 s. This amounts to a maximum throughput of 25 frames/sec at a CPU clock speed of 2400 MHz for the proposed SBPG . It can be argued that the frame rate is acceptable due to the fact that modules in the proposed SBPG run sequentially: the output of the watermarking module is considered as an input to the BPG compression encoder.

TABLE I
QUALITY METRICS FOR THE WATERMARKING AND COMPRESSION TECHNIQUES AND TEST IMAGES.

Test Image	Code	Size (KB)	RMSE	PSNR
Cover Baboon Image (16.7KB)	Watermarked Image	20.0	0.76	50.4
	Watermarked Compressed Image	16.1	0.50	53.1
Cover Forset Image (25.1KB)	Watermarked Image	28.2	0.92	47.9
	Watermarked Compressed Image	24.0	0.60	51.3
Cover IceClimb Image (83.3KB)	Watermarked Image	48.0	0.58	52.7
	Watermarked Compressed Image	46.0	0.42	55.0
Cover Lena Image (32.0KB)	Watermarked Image	40.2	0.89	49.1
	Watermarked Compressed Image	38.8	0.85	51.9
Cover Pepper Image (39.3KB)	Watermarked Image	40.0	0.52	53.6
	Watermarked Compressed Image	39.0	0.41	55.4

TABLE II
COMPARATIVE PERSPECTIVE WITH EXISTING SECURE DIGITAL CAMERA ARCHITECTURES.

Prior Research	Built-in Security Function		Domain	Built-in Compression	object
	Watermarking	Encryption			
Mohanty *et al.* [10]	Invisible Robust	AES	DCT	None	Image
Anand *et al.* [39]	Invisible Feasible	None	DWT	None	Image
Mohanty *et al.* [40]	Visible	None	DCT	JPEG Encoder	Image
Lei *et al.* [41]	Semi-Fragile and Robust	None	DWT	None	Image
Mohanty *et al.* [36]	Visible	None	DCT	MPEG-4 Compression	Video
This paper	Invisible Robust Blind	AES	DCT	BPG Encoder	Image

Overall the proposed SBPG yields a better quality compression at a higher speed along with a double layer of protection with watermarking and encryption by using a state-of-the-art image compression technique (BPG) for achieving high compression ratio.

E. Testing with Different Attacks

A watermark algorithm is considered to be robust if the embedded watermark cannot be impaired without deterioration of the attacked data [42]. Malicious attackers come up with different schemes to impair the watermarks beyond acceptable limits while maintaining the perceptual quality of the attacked data. Therefore it is important to test a watermarking scheme for robustness against various kinds of attacks.

In the current experiment, state-of-the-art attacks have been used for testing the robustness of our watermarking scheme, as follows: Gaussian filter (standard devision = 0.25), Salt noise (density = 0.05), and Gaussian noise (variance = 0.0001). Table III and Figs. 22 and 23 present the PSNR results and the normalized cross correlation coefficients obtained while testing with the aforementioned attacks.

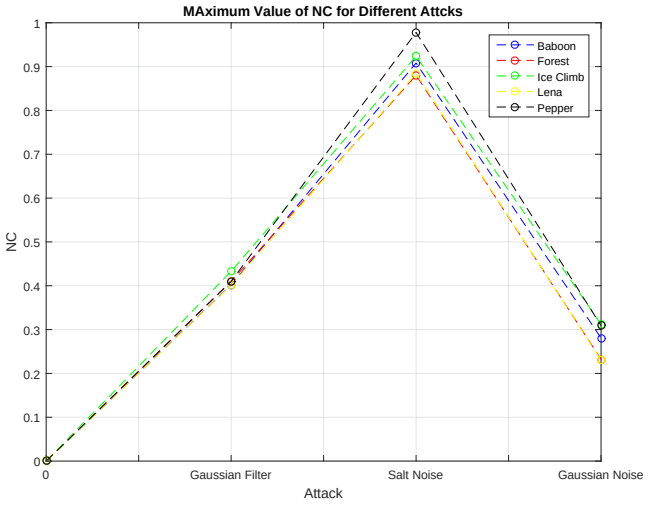

Fig. 22. Maximum Value of NC for Different Attacks.

VIII. POWER OPTIMIZATION PERSPECTIVE OF THE PROPOSED SBPG

Power consumption is one of the most important aspects of any portable application [43] [44]. Low power consumption

TABLE III
THE PERFORMANCE OF SBPG AGAINST VARIOUS ATTACKS.

Watermarked Compressed images	Attack	Maximum Value of NC	Corresponding PSNR
Baboon (PSNR=53.1 dB)	Gaussian fillter	0.4019	51.80
	Salt pepper noise	0.9081	51.57
	Gaussian noise	0.2807	51.43
Forset (PSNR=51.3 dB)	Gaussian fillter	0.4098	48.93
	Salt pepper noise	0.8803	49.15
	Gaussian noise	0.2319	49.58
IceClimb (PSNR=55.0 dB)	Gaussian fillter	0.4330	53.19
	Salt pepper noise	0.9239	53.28
	Gaussian noise	0.3120	53.21
Lena (PSNR=51.9 dB)	Gaussian fillter	0.4012	49.58
	Salt pepper noise	0.8841	49.73
	Gaussian noise	0.2301	49.60
Pepper (PSNR=55.4 dB)	Gaussian fillter	0.4100	53.11
	Salt pepper noise	0.9780	53.34
	Gaussian noise	0.3091	53.15

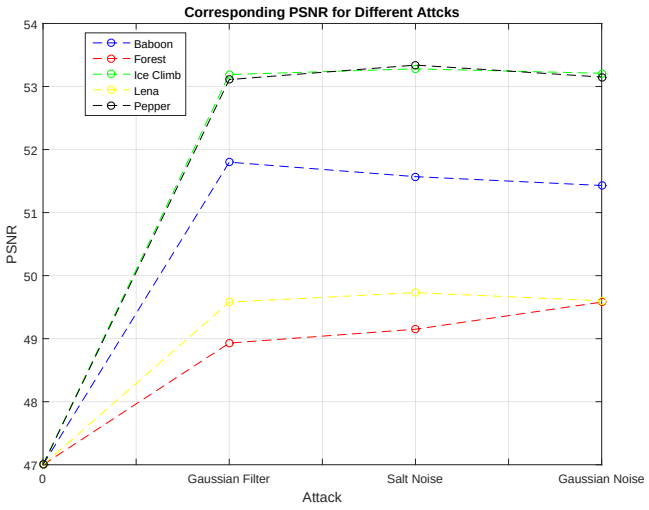

Fig. 23. PSNR of Different Attacks.

results in extended battery life and thereby increases portability. Moreover, low power consumption reduces packaging cost, and is beneficial for cooling in both portable and non-portable applications.

A. DCT Optimization

The DCT optimization aims at minimizing the number of arithmetic operations involved in the compression process. The lower frequency coefficients obtained from the DCT processing of an image block contain most of the visual information whereas the higher frequency coefficients contain noise and some information about edges. The first row and first column in the block represent the DC component while the remaining blocks represent AC components. Usually, the DC component along with few AC components of lower frequencies are enough to reconstruct the image with perceptually the same quality as the original in a very computationally efficient manner. Discarding the high frequency coefficients results in high image compression as well.

B. Sub-Sample Interpolation

The necessity for subsample (or fractional sample) interpolation arises in case of video coding because the positions of objects that change from frame to frame need not necessarily align with the camera grid positions. To capture smooth and continuous motion of objects accurately, the image intensity values at sub-pixel positions are estimated (interpolated) from the samples at integer positions. Motion vectors with quarter-pixel accuracy for the luma component and one-eighth pixel accuracy for chroma components are supported by the HEVC/H.265 standard. To increase the level of precision in estimating video samples, HEVC generates motion compensated prediction signal by employing a number of strategies including subsample interpolation filtering, dynamic range change, and horizontal or vertical shifting.

C. Mechanism of Power Measurement

Power estimation is very crucial to the design of energy-efficient systems. Speed and power traded off are important considerations in real-time operation of devices. Two broad categories of power estimation methods are pattern-dependent and pattern-independent methods. In the pattern-dependent method, power dissipation is estimated using a selected set of input patterns whereas in the pattern independent method, random set of input pattern vectors are used to estimate the average power dissipation through a simulation experiment. In this design, a pattern-independent method is adopted, i.e., average power dissipation is determined after many simulation runs with different inputs. The system is considered as a black box and the current and voltage values are measured to calculate power from them with the help of sensors and power blocks available in Simulink®. The simulations are carried out using the `ode45` solver configuration, as shown in Fig. 24.

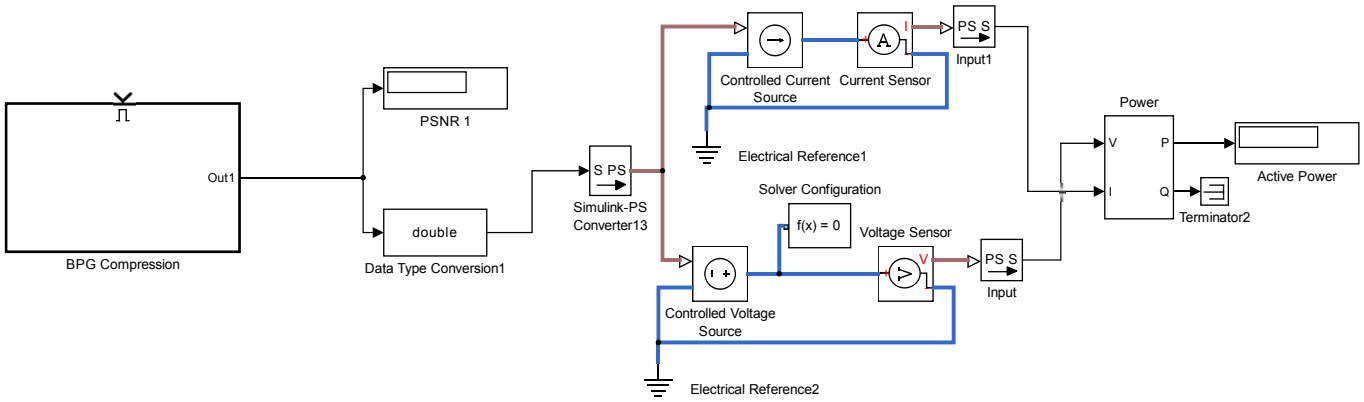

Fig. 24. Experimental Setup for Power Measurement.

Table IV and Fig. 25 presents a comparative view of the "dissipated power" metrics with different test objects for the baseline non-optimal and the proposed optimal designs. These experimental results demonstrate that the proposed SBPG architecture offers energy-efficiency without compromising on quality, compared to the baseline design unoptimized for energy-efficiency.

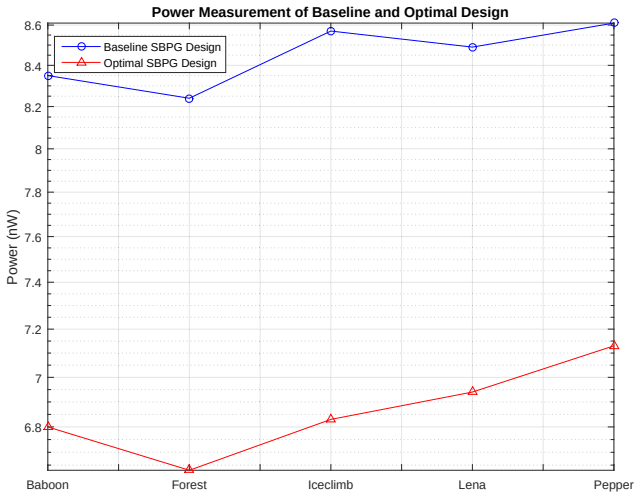

Fig. 25. Power Measurement of Baseline and Optimal Design.

IX. Conclusions And Future Directions of Research

This paper presents a hardware architecture for a secure BPG compressional unit built into a secure digital camera (SDC), which is very useful for image communications in the Internet of Things (IoT). A prototype of the proposed architecture is developed in Simulink®. Experimental results establish the superiority of the new compression technique BPG over the traditional JPEG with respect to compression quality and size of the compressed file. Improvement in the visual quality of the watermarked and BPG compressed images with larger PSNR values, indicates that the proposed SBPG maintains the quality of the watermarked and compressed images intact, and thereby makes it impossible for the human eye to detect the signature of any watermarks in them. Better performance is achieved by choosing to insert the encrypted signature in the central portion of the image containing the crucial image information. This design consideration of using a subimage rather than the whole image results in an architecture optimized with respect to speed and robustness due to reduced computational load and consequent increase speed of operation. Insertion of the watermark in the subimage containing main image information also increases system robustness because any attempt to tamper with the watermark results in image quality degradation. Moreover, watermarking in the frequency domain using DCT increases watermark insertion speed. Finally, the computational speed is further increased with a block-wise DCT of size 8×8. To the best of the authors' knowledge, this is the first ever proposed hardware architecture for SBPG compression integrated with SDC. One possible shortcoming of the proposed architecture is its still-image reliance. Future research could include modifications for other types of medical data as well as development of highly energy-efficient architectures for SBPG and novel IoT based smart applications (other than SHC) of the SBPG for disaster relief, emergency management, crime detection and prevention, etc.

X. Acknowledgment

Preliminary ideas of this paper were presented in the conferences [45], [46]. The authors thank UNT graduate Dr. Umar Albalawi for all his help on this paper. The authors would like to acknowledge inputs of Dr. George Zouridakis, University of Houston.

References

[1] A. Gharaibeh, M. A. Salahuddin, S. J. Hussini, A. Khreishah, I. Khalil, M. Guizani, and A. Al-Fuqaha, "Smart Cities: A Survey on Data Management, Security and Enabling Technologies," *IEEE Communications Surveys Tutorials*, vol. PP, no. 99, pp. 1–1, 2017.

[2] S. P. Mohanty, U. Choppali, and E. Kougianos, "Everything You wanted to Know about Smart Cities," *IEEE Consumer Electronics Magazine*, vol. 5, no. 3, pp. 60–70, July 2016.

TABLE IV
QUALITY METRICS FOR THE BASELINE DESIGN AND THE PROPOSED OPTIMAL DESIGN.

Test Image	SBPG Baseline Design		SBPG Optimal Design		Power Reduction
	PSNR	Power (nW)	PSNR	Power (nW)	
Baboon 128×128	53.1	8.35	52.6	6.80	18%
Forest 256×156	51.3	8.24	50.8	6.63	19%
Iclimb 512×512	55.0	8.57	54.5	6.83	20%
Lena 512×512	51.9	8.49	51.2	6.94	18%
Pepper 512×512	55.4	8.61	54.7	7.13	17%

[3] P. Sundaravadivel, E. Kougianos, S. P. Mohanty, and M. K. Ganapathiraju, "Everything You Wanted to Know about Smart Health Care," *IEEE Consumer Electronics Magazine*, vol. 8, no. 1, pp. x–y, Jan 2018.

[4] L. Catarinucci, D. de Donno, L. Mainetti, L. Palano, L. Patrono, M. L. Stefanizzi, and L. Tarricone, "An IoT-Aware Architecture for Smart Healthcare Systems," *IEEE Internet of Things Journal*, vol. 2, no. 6, pp. 515–526, Dec 2015.

[5] F. Bellard, "The BPG Image Format," http://bellard.org/bpg/, last Accessed on 09/20/2015.

[6] E. Kougianos, S. P. Mohanty, G. Coelho, U. Albalawi, and P. Sundaravadivel, "Design of a High-Performance System for Secure Image Communication in the Internet of Things," *IEEE Access*, vol. 4, pp. 1222–1242, 2016.

[7] M. Zhang, A. Raghunathan, and N. K. Jha, "Trustworthiness of Medical Devices and Body Area Networks," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1174–1188, Aug 2014.

[8] C. Li, A. Raghunathan, and N. K. Jha, "Hijacking an Insulin Pump: Security Attacks and Defenses for a Diabetes Therapy System," in *Proceedings of the 13th IEEE International Conference on e-Health Networking, Applications and Services*, 2011, pp. 150–156.

[9] R. Khatoun and S. Zeadally, "Cybersecurity and Privacy Solutions in Smart Cities," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 51–59, March 2017.

[10] S. P. Mohanty, "A Secure Digital Camera Architecture for Integrated Real-Time Digital Rights Management," *Elsevier Journal of Systems Architecture (JSA)*, vol. 55, pp. 468–480, 2009.

[11] A. Darji, A.N.Chandorkar, S.N.Merchant, and V. Mistry, "VLSI Architecture of DWT Based Watermark Encoder for Secure Still Digital Camera Design," in *Proceedings 3rd International Conference on Emerging Trends in Engineering and Technology (ICETET)*, 2010, pp. 760 – 764.

[12] L. Tian and H. M. Tai, " Secure Images Captured by Digital Camera," in *Proceedings International Conference Consumer Electronics*, 2006, pp. 341 – 342.

[13] S.C.Ramesh and M. M. I. Majeed, "Implementation of a visible watermarking in a secure still digital camera using VLSI design," in *Proceedings AFRICON*, 2009, pp. 1 – 4.

[14] T. Winkler and B. Rinner, "TrustCAM: Security and Privacy-Protection for an Embedded Smart Camera Based on Trusted Computing," in *Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2010, pp. 593–600.

[15] I. Haider, M. Hberl, and B. Rinner, "Trusted Sensors for Participatory Sensing and IoT Applications based on Physically Unclonable Functions," in *Proceedings of the 2nd ACM International Workshop on IoT Privacy, Trust, and Security*, 2016, pp. 14–21.

[16] S. Ullah, B. Rinner, and L. Marcenaro, "Smart Cameras with Onboard Signcryption for Securing IoT Applications," in *Proceedings of the Global Internet of Things Summit (GIoTS)*, 2017, pp. 1–6.

[17] M. Bramberger, J. Brunner, B. Rinner, and H. Schwabach, "Real-time video analysis on an embedded smart camera for traffic surveillance," in *Proceedings 10th IEEE Real-Time and Embedded Technology and Applications Symposium*, 2004, pp. 174 – 181.

[18] Z. N. K. Wafi, R. Ahmad, and P. M.P, "Highways Traffic Surveillance System (HTSS) using OpenCV," in *Proceedings IEEE Control and System Graduate Research Colloquium (ICSGRC)*, 2010, pp. 44 – 48.

[19] R. Khoshabeh, T. Gandhi, and M. Trivedi, "Multi-camera Based Traffic Flow Characterization & Classification," in *Proceedings IEEE Intelligent Transportation Systems Conference ITSC*, 2007, pp. 259 – 264.

[20] X. Lu, C. Ye, and J. Y. andYaying Zhang, "A Real-Time Distributed Intelligent Traffic Video-Surveillance System on Embedded Smart Cameras," in *Proceedings Fourth International Conference on Networking and Distributed Computing (ICNDC)*, 2013, pp. 51 – 55.

[21] S. ElKerdawy, A. Salaheldin, and M. ElHelw, "Vision-based scale-adaptive vehicle detection and tracking for intelligent traffic monitoring," in *Proceedings IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2014, pp. 1044 – 1049.

[22] S. P. Mohanty, A. Sengupta, P. Guturu, and E. Kougianos, "Everything You Want to Know About Watermarking," *IEEE Consumer Electronics Magazine*, vol. 6, no. 3, pp. 83–91, July 2017.

[23] S. P. Mohanty, *Nanoelectronic Mixed-Signal System Design*. McGraw-Hill Education, 2015, no. 9780071825719.

[24] C. Folk, D. C. Hurley, W. K. Kaplow, and J. F. X. Payne, "The Security Implications of The Internet of Things," AFCEA International Cyber Committee, Tech. Rep., 2015. [Online]. Available: http://www.afcea.org/mission/intel/documents/InternetofThingsFINAL.pdf

[25] "Advanced Encryption Standard (AES)," Federal information processing standards publication 197, Tech. Rep., 2001. [Online]. Available: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[26] N. M. Kosaraju, M. Varanasi, and S. P. Mohanty, "A High-Performance VLSI Architecture For Advanced Encryption Standard (AES) Algorithm," in *Proceedings of the 19th International Conference on VLSI Design*, 2006, pp. 481–484.

[27] U. Albalawi, S. P. Mohanty, and E. Kougianos, "A New Region Aware Invisible Robust Blind Watermarking Approach," *Springer Multimedia Tools and Applications Journal*, vol. 76, no. 20, pp. 21 303–21 337, Oct 2017.

[28] S. P. Mohanty, N. Pati, and E. Kougianos, " A Watermarking Co-Processor for New Generation Graphics Processing Units," in *Proceedings of the 25th IEEE International Conference on Consumer Electronics (ICCE)*, 2007, pp. 303–304.

[29] B. C. Choi and D. I. Seo, "A statistical approach for optimal watermark coefficients extraction in HVS-based blind watermarking system," in *Proceding The 7th International Conference on Advanced Communication Technology*, vol. 2, 2005, pp. 1085 – 1088.

[30] U. Albalawi, S. P. Mohanty, and E. Kougianos, "A Hardware Architecture for Better Portable Graphics (BPG) Compression Encoder," in *Proceedings of the 1st IEEE International Symposium on Nanoelectronic and Information Systems*, 2015, pp. 291–296.

[31] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1649–1668, Dec 2012.

[32] M. U. K. Khan, M. Shafique, and J. Henkel, "Software Architecture of High Efficiency Video Coding for Many-Core System with Power-Efficient Workload Balancing," in *Proceedings Automation and Test Design in Europe Conference and Exhibition (DATE)*, 2014, pp. 1–6.

[33] C. C. Chi, M. Alvarez-Mesa, B. Juurlink, G. Clare, F. Henry, S. Pateux, and T. Schierl, "Parallel Scalability and Efficiency of HEVC Parallelization Approaches," *IEEE Transactions on Circuits and Systems for Video Technology.*, vol. 22, no. 12, pp. 1827 – 1838, 2012.

[34] M. Shafiquea, M. U. K. Khan, and J. Henkel, "Power Efficient and Workload Balanced Tiling For Parallelized High Efficiency Video Coding," in *Proceedings IEEE International Conference on Image Processing*, 2014, pp. 1253 – 1257.

[35] V. Sze, M. Budagavi, and G. J. Sullivan, Eds., *High Efficiency Video Coding (HEVC)*. Springer International Publishing, 2014.

[36] S. P. Mohanty and E. Kougianos, "Real-Time Perceptual Watermarking Architectures For Video Broadcasting," *Elsevier Journal of Systems and Software (JSS)*, vol. 19, no. 12, pp. 724 – 738, 2011.

[37] C. J. Willmott and K. Matsuura, "Advantages of the Mean Absolute Error (MAE) over the Root Mean Square error (RMSE) in assessing average model performance," in *Proceedings Climate Research*, vol. 30, 2005, p. 79 82.

[38] Q. Huynh-Thu and M. Ghanbari, "Scope of Validity of PSNR in Image/Video Quality Assessment," in *Electronics Letters*, vol. 44, 2008, pp. 800 – 801.

[39] A. Darji, A.N.Chandorkar, S.N.Merchant, and V. Mistry, "VLSI Architecture of DWT Based Watermark Encoder for Secure Still Digital Camera Design," in *Proceeding 3rd International Conference on Emerging Trends in Engineering and Technology (ICETET)*, 2010.

[40] S. P. Mohanty, N. Ranganathan, and R. K. Namballa, "A VLSI architecture for visible watermarking in a secure still digital camera (S^2DC) design," in *Proceeding IEEE Transactions on Very Large Scale Integration (VLSI) System*, vol. 13, 2005, pp. 1002 – 1012.

[41] L. Tian and H.-M. Tai, "Secure images captured by digital camera," in *Proceeding International Conference on Consumer Electronics (ICCE)*, 2006.

[42] H. Agarwal, B. Raman, and P. K. Atrey, "Watermarking schemes to secure the face database and test images in a biometric system," in *Proceedings IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, 2013, pp. 128–133.

[43] S. P. Mohanty, N. Ranganathan, and K. Balakrishnan, "Design of a low power image watermarking encoder using dual voltage and frequency," in *Proceedings of the 18th International Conference on VLSI Design*, 2005, pp. 153–158.

[44] S. P. Mohanty, N. Ranganathan, and V. Krishna, "Datapath scheduling using dynamic frequency clocking," in *Proceedings of the IEEE Computer Society Annual Symposium on VLSI*. IEEE, 2002, pp. 58–63.

[45] U. Albalawi, S. P. Mohanty, and E. Kougianos, "SBPG: A Secure Better Portable Graphics Compression Architecture for High Speed Trusted Image Communication in IoT," in *Proceedings of the 17th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE)*, 2016, pp. 1–5.

[46] ——, "Energy-Efficient Design of the Secure Better Portable Graphics Compression Architecture for Trusted Image Communication in the IoT," in *Proceedings of the IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, 2016, pp. 302–307.