

8-11-2015

Performance Comparison of Data Reduction Techniques for Wireless Multimedia Sensor Network Applications

Pinar Sarisaray-Boluk
Bahcesehir University

Kemal Akkaya
Department of Electrical and Computer Engineering, Florida International University, kakkaya@fiu.edu

Follow this and additional works at: https://digitalcommons.fiu.edu/ece_fac



Part of the [Electrical and Computer Engineering Commons](#)

Recommended Citation

Sarisaray-Boluk, Pinar and Akkaya, Kemal, "Performance Comparison of Data Reduction Techniques for Wireless Multimedia Sensor Network Applications" (2015). *Electrical and Computer Engineering Faculty Publications*. 11.
https://digitalcommons.fiu.edu/ece_fac/11

This work is brought to you for free and open access by the College of Engineering and Computing at FIU Digital Commons. It has been accepted for inclusion in Electrical and Computer Engineering Faculty Publications by an authorized administrator of FIU Digital Commons. For more information, please contact dcc@fiu.edu.

Research Article

Performance Comparison of Data Reduction Techniques for Wireless Multimedia Sensor Network Applications

Pinar Sarisaray-Boluk¹ and Kemal Akkaya²

¹Bahcesehir University, 34349 Istanbul, Turkey

²Florida International University, Miami, FL 33174, USA

Correspondence should be addressed to Pinar Sarisaray-Boluk; pinar.sarisaray@eng.bahcesehir.edu.tr

Received 28 March 2015; Revised 20 June 2015; Accepted 11 August 2015

Academic Editor: Haiping Huang

Copyright © 2015 P. Sarisaray-Boluk and K. Akkaya. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the increased use of smart phones, Wireless Multimedia Sensor Networks (WMSNs) will have opportunities to deploy such devices in several contexts for data collection and processing. While smart phones come with richer resources and can do complex processing, their battery is still limited. Background subtraction (BS) and compression techniques are common data reduction schemes, which have been used for camera sensors to reduce energy consumption in WMSNs. In this paper, we investigate the performance of various BS algorithms and compression techniques in terms of computation and communication energy, time, and quality. We have picked five different BS algorithms and two compression techniques and implemented them in an Android platform. Considering the fact that these BS algorithms will be run within the context of WMSNs where the data is subject to packet losses and errors, we also investigated the performance in terms of packet loss ratio in the network under various packet sizes. The experiment results indicated that the most energy-efficient BS algorithm could also provide the best quality in terms of the foreground detected. The results also indicate that data reduction techniques including BS algorithms and compression techniques can provide significant energy savings in terms of transmission energy costs.

1. Introduction

The increasing availability of battery-operated low- or high-resolution wireless cameras [1, 2] provides opportunities to improve the quality, scalability, and efficiency of current remote surveillance and monitoring systems via smart networking and processing techniques. Referred to as WMSNs, such networks can be used in several outdoor applications such as border surveillance, habitat monitoring, and critical infrastructure monitoring in the large scale due to convenient deployment of wireless cameras with cheaper costs [3, 4]. The main advantage of WMSNs compared to the traditional multicamera network is that they can operate in environments where there is no electricity and Internet access [5].

In addition to these battery-operated cameras in WMSNs, the rapid adoption of smart phones provides a growing capability to collect, identify, and transport a wide variety of multimedia data in a convenient manner. Possible use of smart phones in WMSNs for monitoring purposes brings

additional gains into picture due to their processing power and ubiquity. WMSNs typically suffer from poor resources and thus the current research has focused on lightweight mechanisms when dealing with multimedia data processing and communication. The use of smart phones may alleviate some of these issues in several contexts. In particular, the use of smart phones during emergency situations is invaluable. The smart phones can be used as backup mechanisms for the collection and transmission of multimedia data. For instance, during an earthquake, when there is no Internet connection via the communication infrastructure, smart phones can be used to receive multimedia data (i.e., acting as a gateway) from existing low-resolution/fixed wireless cameras and relay it to a center via 4G connections. Similarly, survivors can be recorded by smart phones and this data can be relayed via the smart phones' wireless connections. Smart phones can be used as temporary relays/gateways in national parks when the surveillance cameras' network connection is lost/damaged due to some disasters such as fire or storms [6, 7].

In all of these cases, the smart phones can process the multimedia data but the use is temporary due to the limited battery power they have. Therefore, given the criticality of the situations, the battery power should be used wisely. This necessitates some smart data processing techniques to reduce the energy consumption of the smart phones. While existing research in WMSNs has already focused on such smart techniques, there was always the problem of limited CPU and memory, which restricted the use of sophisticated multimedia processing techniques on these cameras. As a result, only primitive approaches have been preferred or proposed for these nodes. Such challenges, however, are not the case for smart phones as they come with more sophisticated CPUs and larger memories. Therefore, most of the existing techniques for traditional multicamera networks can be applied to smart phones for reducing multimedia data size. However, the testing of these techniques on smart phones has been underexplored.

In this paper, we focus on this aspect and investigate the computation and communication energy consumption of data reduction techniques including background subtraction algorithms and compression techniques for Android-based smart phones. BS is a common processing approach to discover region of interest or moving parts of a frame (i.e., foreground) by performing subtraction between each video frame and a background model. Compression techniques also help decrease the amount of the transmitted multimedia data. The main motivation is to transmit the extracted data rather than the whole frame for saving energy. The background can be sent initially and the exact video can be constructed at the receiver side easily when the foreground frames arrive. While there exist many other studies which compare BS algorithms for different challenges such as illumination changes, dynamic background, and shadows [8–11], to the best of our knowledge the comparison in terms of computational energy overhead, complexity, and quality has not been done before. In addition, when Android phones are used in a WMSN under harsh environments such as earthquakes, where packet losses are inevitable during communications, the performance of BS algorithm under distorted frames needs to be investigated.

This paper is an extended version of our work in [12]. The prior work presented the BS algorithms and their performance with limited experimentation and did not provide any details of the encoding techniques for multimedia transmission in WMSNs. In this extended version, we have implemented and compared 5 BS algorithms on an Android platform in terms of energy, time, and BS quality through extensive experimentation. The quality assessment is done under distorted images in cases when the multimedia data is generated by other cameras and sent over a wireless link. When transmitting multimedia data over a wireless link, transmitted data is exposed to losses or errors due to channel impairments [5], creating distortion. For assessing the quality, we have used recall, precision, and harmonic mean (F -measure) as the metrics.

We have also compared two common compression techniques, namely, H.264 and MPEG-4, on an Android platform in terms of energy and storage when used in conjunction

with BS algorithms in a WMSN setup. These compression techniques are also evaluated in terms of multimedia quality over a single hop WMSN. The experiment results showed that AMF approach [13] performs best in terms of energy savings and time. There are also several interesting outcomes regarding the quality for both regular and distorted images.

The rest of the paper is organized as follows: Section 2 introduces related work. Section 3 provides the background and motivation. Section 4 presents the experimental setup. In Section 5 experimental results are given including the evaluation of the BS algorithms and compression techniques. Finally, conclusion is given in Section 6.

2. Related Work

Several performance evaluation studies have been published to examine the weakness and strengths of BS algorithms [8, 9, 14–17]. In [8], a comparative evaluation of classical approaches has been conducted on background subtraction algorithms for exposing static foreground objects. The previous solutions have been categorized into several classes. Then, representative solutions have been compared using both quantitative and qualitative metrics. The paper concludes that subsampling based solutions give the best results at the expense of a low computational cost for general-purpose static object detection. The authors in [9] have evaluated BS algorithms for a wide group of challenges in video surveillance. Nine different object segmentation algorithms are performed for surveillance settings where each of them covers different challenge such as gradual illumination changes, dynamic background, and shadows. The effect of the used postprocessing algorithms on several popular BS algorithms has been examined in [15]. Their comparison is conducted with a varied set of 7 outdoor and 6 indoor video sequences for different threshold parameters. In [18], the authors provided a comparative study of several BS algorithms in terms of robustness, memory requirement, and computational efficiency. In [17], several background subtraction algorithms are applied to the images with the ground truth in order to count the number of cars. The performance of the algorithms for the processing of million images is assessed based on some important metrics including scalability, accuracy, and processing time. The authors in [14] devote efforts to a certain application: intrusion detection in video surveillance. A multilevel technique is utilized for evaluating and comparing background subtraction algorithms. Moreover, a new similarity measure, called D -Score, is also proposed which is adapted to the context of intrusion detection.

The authors in [19] have introduced a low complexity background subtraction algorithm which needs low-memory requirement. In this technique, a scaled background image is stored in internal memory of hardware platform, which leads to faster access. In subtraction operation, a lightweight image upscaling technique is employed to obtain the scaled down image. Background pixel is obtained from this scaled image. In [20], authors proposed a novel technique for background subtraction on videos encoded in the baseline profile of H.264/AVC. The performance of their method is shown by

using diverse set of real-world surveillance sequences for real-time network streaming applications.

In [21], several BS techniques are compared for different set of scenarios. Each of them is analyzed for different parameter configuration where precise ground data is missing. A sample of traffic snapshot captured from various intersections and highways is recorded in these images. The study aims to count an approximate number of cars in these images. Several approaches ranging from simple BS with global thresholding to more complex statistical ones are implemented and evaluated on various videos with ground truth. The aim of this study is to present an analytic ground to identify the strengths and weaknesses of the mostly used motion detection techniques. The study shows that simple modeling methods present roughly as well as complex methods. Additionally, postprocessing techniques greatly increase the performance of the BS algorithms when the parameters are chosen properly. Performance comparisons of these test scenarios are performed with fabricated datasets that present the fact that multimodal BS techniques outperform unimodal ones.

The authors in [22] show an energy consumption analysis of an Android smart phone for multimedia data transmission over UDP and TCP on an IEEE 802.11g network. This study reveals the relation between the wireless parameters, such as channel quality and network load, and battery consumption of the mobile phones for multimedia delivery. The results indicate that the network load and the channel quality have a significant impact on the energy consumption. In [23], predictive and distributed coding techniques are studied empirically to present their energy efficiency in wireless sensor networks. The results for predictive video coding present the idea that intervideo coding shows higher compression efficiency. However, it consumes much more energy than intracoding. Hence, the authors propose utilizing image compression based intracoding in order to increase energy efficiency in the predictive video coding technique. Additionally, the results for distributed video coding reveal that Wyner-Ziv encoder is more energy efficient than PRISM encoder. The paper also proposes some modifications on PRISM and Wyner-Ziv encoders to decrease the energy consumption of these encoders. Eventually, their results reveal that the major cause of energy wastage is local processing implemented for video compression and not video transportation.

The authors in [24] survey some recent studies which improve the energy efficiency of wireless multimedia transmission in mobile devices. They classify the previous work conforming to different layers of the Internet protocol stack they use. Then, these works are regrouped based on their traffic scheduling and multimedia content adaptation techniques and compared in terms of their energy efficiency. The traffic scheduling category includes several solutions which optimize receiving energy without altering the real multimedia content. The second category contains the solutions which changes the real multimedia content in order to reduce energy consumption on the receiver and to decode and view the content. The work also considers some other research work which deals with energy aware multimedia data transmission among mobile devices. The performances of the MPEG-4 and H.264 video codes are analyzed by

EvalVid framework and toolset in [25] for wireless ad hoc and sensor networks. In this work, the authors have performed simulations in NS2 for AODV routing protocol and constant bit rate traffic. The performance of these codes is evaluated in terms of Peak Signal to Noise Ratio (PSNR) [12]. The study shows that H.264 is more efficient for low bandwidth networks because it produces less amount of the data and has better quality compared to the MPEG-4 ones. Although this study shows the performance of two encoding methods for multimedia data transmission, it does not consider the computation energy and delay cost at the source.

Different from the above works, our study in this paper investigates the performance of various BS algorithms and compression techniques on Android devices in terms of various criteria. First, our main goal is to investigate the energy efficiency of these approaches, which has not been done before. Second, we investigate the quality of the BS algorithms under different network channel conditions and packet sizes. Third, we show the gain of energy efficiency in terms of video transmission when compared to traditional approach of sending the whole video. It is important to note that the study is conducted on Android devices integrated with WMSN, which has not been done before. Finally, we investigate the use of different compression techniques along with BS algorithms.

3. Background Subtraction for Android Devices

3.1. Preliminaries on BS. Background subtraction is a crucial task for many vital applications where the aim is to identify moving objects in the multimedia data for applications such as surveillance [21]. Determining moving objects from a video sequence is an essential and important task in many computer-vision applications [26]. The purpose of background subtraction is to separate the static part of the scene from the dynamic one in the raw video frames. This technique has three components: foreground detection; background maintenance; and postprocessing.

Foreground detection indicates that pixels are classified as background or foreground. An example frame and its foreground obtained from the considered techniques are shown in Figure 1. Background maintenance shows how the background is maintained over time. Different techniques are used for background subtraction process such as frame differencing, mean filtering, and mixture of Gaussian distributions [21]. Background maintenance shows how the background is maintained over time. It decides how the background is adapted to consider some important situations, which may take place. There are two common maintenance methods: the blind and the selective. Postprocessing specifies how the segmented object areas are postprocessed to refuse false positives. Postprocessing goal is to enhance the results of BS algorithms. There are several kinds of postprocessing techniques: noise removal, blob processing, and object-level feedback [15].

A good background subtraction algorithm has many rigors. First, the algorithm must be robust against changes

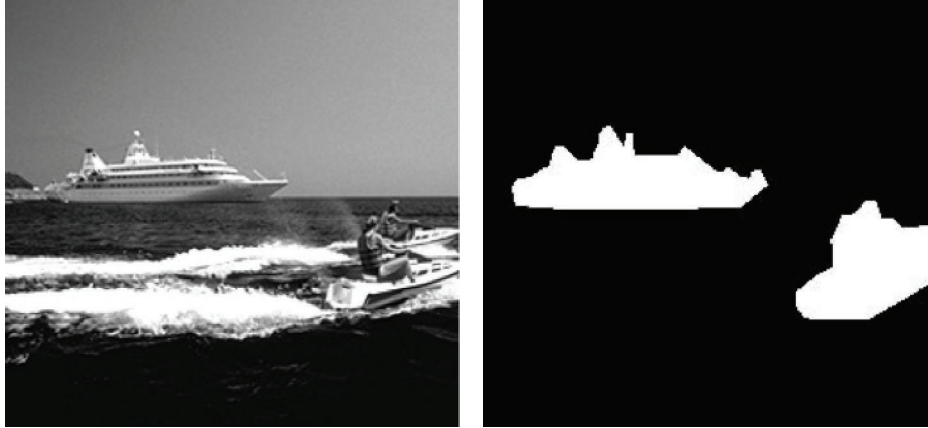


FIGURE 1: Original image and its region of interest area.

in illumination. Second, it should prevent detecting nonstationary background objects, namely, swinging leaves, rain, snow, and shadow cast by moving objects. Finally, its internal background model should respond promptly to variation in background.

3.2. Motivation. In order to decrease the amount of the multimedia data, BS algorithms are utilized for many multimedia applications. BS algorithms have been used in WMSNs for the purpose of multimedia data reduction to reduce energy consumption of camera sensors [27]. The main research challenge in this context was to be able to design lightweight BS algorithms that will require limited memory and CPU usage due to severe resource constraints of camera sensors. The quality of the BS algorithm was not the main concern. Instead, the focus was to reduce the computational complexity. For instance, in [28, 29], the authors work with image blocks (i.e., 5×5 pixels) rather than the whole frame. Hence pixel blocks rather than individual pixels are obtained for processing. Average color of each block is then computed and used for reducing computation costs. Similarly, the work in [30] is based on compressive sensing (CS), as opposed to getting the average color of each block. The idea of CS is simple and different compared to the process of traditional compressing for images. It decreases the dimensionality of the data while protecting most of the information. BS is applied in the reduced data.

In this paper, we still consider a scenario in the context of WMSNs but battery-operated Android devices are also considered as part of this network. Such a device (e.g., an Android phone) can act as a relay to transmit multimedia data from cameras using its WiFi capability assuming that the WMSN runs a protocol stack based on IEEE 802.11s which is the new mesh networking standard [31]. Another possibility is to employ the Android devices as a data collector that relays the data to a remote station using 4G/LTE connections. A sample system architecture is shown in Figure 2.

For such situations, BS algorithms can still be applied to save energy. Nonetheless, due to resource characteristics of Android devices, the selected BS algorithms do not have to be computationally efficient ones. This is because Android

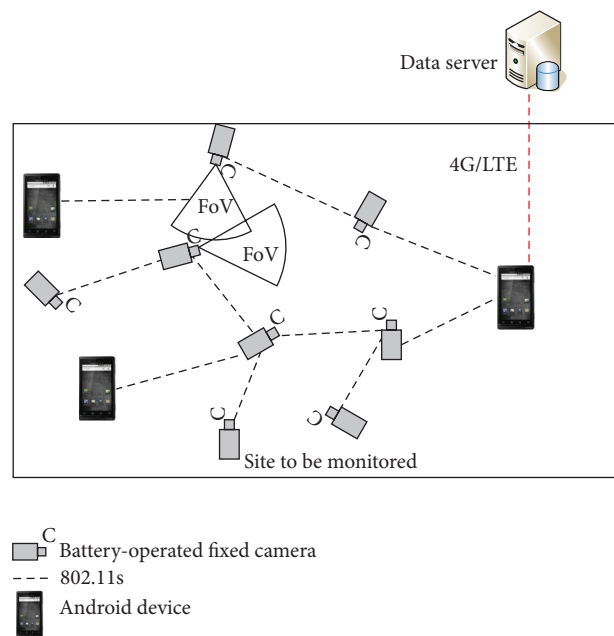


FIGURE 2: A WMSN with Android devices involved.

devices have sophisticated hardware (e.g., CPU and memory), which makes them available to run computationally heavy algorithms. The only limitation, though, is the limited battery. These characteristics make smart phones a separate category in terms of resource availability. At the bottom of the hierarchy are the camera sensors. The next level includes Android devices and finally at the top level there will be devices without any computational and power limitations such as multimedia gateways.

Considering the role of Android devices in WMSNs, we argue that the performance of BS algorithms that will be run on Android devices needs to be assessed under a variety of conditions. This paper will be the first to assess the performance of BS algorithms for Android environments. We mainly focus on two aspects. First, we would like to assess energy consumption for both the computation and



FIGURE 3: An example of a distorted image.

communication. Second, we would like to consider various network conditions assuming harsh environments such as earthquakes. When transmitting multimedia data over wireless channels in WMSNs, transmitted data is subject to packet losses or errors due to channel impairments. In particular, wireless link quality fluctuates dramatically over time due to the distance between nodes, multipath propagation, and interference [5]. The problems can be compounded with the effects from the environments (e.g., line of sight, obstacles). These problems may affect the quality of the received multimedia data which makes the selection of BS algorithm a more challenging task. Therefore, we will test the quality of the BS algorithms under various network conditions. As will be elaborated in the next section, we will consider different packet loss ratios and packet sizes when sending video data to the Android devices as seen in Figure 3 which makes the selection of BS algorithm a more challenging task.

3.3. Considered BS Algorithms. We have picked 5 different BS algorithms to be compared by considering common methods of BS algorithms in the literature. These algorithms are shown in Table 1. We have selected a set of current methods and more common approaches. These techniques are shown in Table 1. MBG algorithm is the simplest technique selected as a baseline for performance improvement. It is explained in Section 3.1. Many BS techniques have been proposed. In order to reveal future directions of research, we have selected a moderate set of these involving current methods and more common approaches. We consider the Gaussian Mixture Model (AGM) algorithm for the background subtraction in [32]. The algorithm is implemented to generate the foreground objects utilizing a Gaussian mixture probability density with the elimination of shadows. The arguments for each Gaussian distribution are modified in a recurrent method. Besides, the proper number of Gaussian distributions is chosen during pixel processing for adjusting to the viewed scene. Then, a k -Nearest Neighbor (k -NN) classifier is employed for recognizing persons. The classifier frequently avails two attributes used for object classification: area and

TABLE 1: Selected BS techniques.

Method	Naming
Zivkovic and van der Heijden [32]	AGM
McFarlane and Schofield [13]	AMF
Oliver et al. [34]	EGB
McIvor [56]	MBG
Wren et al. [35]	RGA

the rate of the bounding box related to each detected object. This approach is easy but effective, and it leads remarkably to the tracking and prevents a complicated procedure for training data [33]. The nonperson objects and image noise can be efficiently deleted.

The second considered technique called AMF is a segmentation and tracking technique which has proposed to track piglets in [13]. The method is based on an approximate median filter. The technique utilizes one background estimate, which is compared pixel-by-pixel with the current frame and updated. As long as the considered pixel in the frame is greater than (brighter) the one in the background, the background value is increased by one or vice versa. Finding the difference between the current frame and background identifies the foreground. If it is greater than predefined threshold, the pixel is a foreground pixel. The approximate median filter provides low-memory consumption, fast computation, and robustness. However, this technique responds with slower update to big changes on illumination.

A real-time computer-vision and machine learning system is proposed for recognizing and modeling human behaviors for a surveillance application in [34]. A nonpixel technique (EGB) is proposed by using an eigenspace to construct background. This work also takes into account neighboring statistics. The algorithm has an ability to learn the background model from video sequences even if the video is composed of the moving foreground objects. Supervised statistical machine learning technique is used in order to identify normal single person behaviors and to model for person-to-person interactions.

Wren et al. [35] have proposed a technique named RGA for tracking people and interpreting their behavior called in a real-time system. The system is called Pfinder that includes a background estimation module. This technique is composed of a single Gaussian distribution for the background model at each pixel, and a variable number of Gaussian distributions coincide with several foreground object models. Pixels are divided into the background and foreground by discovering the model with the least Mahalanobis distance. Pfinder implements a maximum a posteriori probability approach for detecting and tracking of human body using simple two-dimensional models. It combines a priori knowledge about people so as to preinstall itself and to recover from errors. Pfinder indicates the utility of stochastic, region-based features for real-time image understanding. This approach is important for interactive-rate interpretation of the human form without custom hardware [35].

3.4. Considered Compression Techniques. In addition to BS algorithms, in this work, we have compared the performances of two well-known compression techniques, namely, MPEG-4 and H.264 [36–39], for video transmission when BS is in use. MPEG-4 and H.264 are used standards for the coding of the visual data. These two techniques include two main features: a coded presentation naming syntax which represents visual information on a compressed form and a decoding method to reconstruct the original visual data. MPEG-4 has been developed by Moving Picture Experts Group (MPEG) [40]. H.264 coding standard is developed by both the Joint Video Team (JVT) of the International Telecommunication Union-Telecommunication Standardization Sector (ITU-T) [41] and the Moving Picture Experts Group (MPEG). This standard has gained a significant advance on the compression performance over the existing standards. Although MPEG-4 and H.264 both engage with compressing of video data, these standards have different stress in their techniques. While MPEG-4 cares about flexibility, H.264 emphasizes efficiency and reliability. Hence, MPEG-4 can deal with video data including many different kinds of shapes by using a highly flexible toolkit of coding methods and resources. Contrary to MPEG-4, H.264 is really focused on efficient compression of video data [37].

MPEG-4 employs discrete cosine transform (DCT) and predictive coding to decrease casual redundancy [42, 43]. MPEG-4 describes a group of profiles: simple, object based, scalable, still texture, and studio profiles. As MPEG-4 is object based, video objects (VOs) in each scene are decoded separately. If individual decoding of the separate objects is not useful, the entire scene can be decoded as one VO. Each VO may include some scalability layers (one base layer and one or more enhancement layers), which are named as video object layers (VOLs). Each VOL contains video object planes which refer to an ordered sequence of snapshots in time. The encoder works on the shape, motion, and texture properties for each VOP.

H.264 specifies three kinds of profiles: baseline, extended, and main. All of these profiles are employed for different applications such as video conferencing, video streaming, video broadcast, and storage [25]. It uses block based motion compensated video coding and other features. H.264 includes two different layers, namely, Video Coding Layer (VCL) and Network Abstraction Layer (NAL). VCL consists of the specification of the main video compression components which carry out basic functions such as entropy coding, motion compensation, and transform coding of coefficients [41, 44].

4. Experiment Setup

To examine the performance of the data reduction techniques, we have conducted a group of simulation experiments in Eclipse IDE for Android [45] including Java, C++, JNI [46], and OpenCV libraries [47]. Assuming that an HTC Inspire 4G Android phone with 1GHz Snapdragon processor and 4GB memory on-board is part of the WMSN architecture, video data is processed on this phone. Once the phone obtains the video, BS algorithms are executed on this video in order to

find the foreground of the considered frames. In this part of the experiments, we assumed that only one source generates multimedia data and this data is transmitted to the smart phone. We have not considered any distributed collaboration between the existing nodes and the smart phone. In the energy analysis, we have used the PowerTutor tool which allows measuring the battery consumption of each running application on the phone [48].

For evaluating the performance of the compression techniques in terms of quality and energy over WMSN, we have used Network Simulator (NS-2) EvalVid framework [49].

4.1. Performance Metrics. We used three metrics for the overhead of the algorithm:

- (i) *Energy*: this metric indicates the consumed energy from the battery for computation or communication.
- (ii) *Time*: this metric indicates the computational complexity of the considered algorithm.
- (iii) *Storage*: this metric indicates the number of bytes to store the compressed data after the BS algorithm is executed.

In order to compare BS performance qualitatively, we adopted the recall, precision, and *F*-measure metrics used by Brutzer et al. [9]. Before describing these metrics, the following quantities should be known.

- (i) True positives (NTP): the number of foreground pixels correctly detected.
- (ii) False positives (NFP): the number of background pixels incorrectly detected as foreground (false alarms).
- (iii) True negatives (NTN): the number of background pixels correctly detected.
- (iv) False negatives (NFN): the number of foreground pixels incorrectly detected as background (misses).

Based on these definitions, we define the metrics as follows.

- (i) *Recall*: this metric refers to detection rate or sensitivity. Additionally, we also use FP as metric, which is the percentage of background pixels incorrectly detected as foreground. It is computed as follows:

$$\text{Recall} = \frac{\text{NTP}}{\text{NTP} + \text{NFN}}. \quad (1)$$

- (ii) *Precision*: it refers to specificity and can be computed as follows:

$$\text{Precision} = \frac{\text{NTP}}{\text{NTP} + \text{NFP}}. \quad (2)$$

- (iii) *F-measure*: this metric is the weighted harmonic mean of recall and precision and computed as follows:

$$F = 2 * \frac{\text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}. \quad (3)$$

- (iv) *Peak Signal to Noise Ratio (PSNR)*: this metric is used as the multimedia quality metric to show the performance of the BS techniques on the transmitted multimedia quality.

A good background algorithm should attain as high recall value as possible without losing precision. Although high values of recall and precision mean high performance, these metrics should be examined together. There is a trade-off between these two metrics. To achieve high recall value, one may need to sacrifice precision or vice versa. Hence, *F*-measure [50] is also used to express the performance when considering both the precision and recall results simultaneously.

For example, a simple algorithm that assigns every pixel to foreground will have a perfect recall of 100% but an unacceptably low score in terms of precision. Conversely, if a system assigns most of the pixels to background, it will have a high score in terms of precision but will sacrifice recall to a significant degree. We used the *F*-measure as another performance measure in order to exactly compare the performance when considering both the precision and recall results simultaneously. The *F*-measure is maximized when the values of recall and precision are equally high or close. If it is set to 1, it is denoted as *F1* [51]. In this paper, the *F1* measure was used to compare the performance of the proposed method with that of the other methods.

4.2. Multimedia Quality Metric. PSNR is calculated with the mean squared error (MSE), computed by averaging the squared intensity differences of distorted and reference frame pixels, along with the related quantity of PSNR. In this metric, we measure quality distortion comparing the input frame of the source's encoder against the impaired frame of the destination's (sink) decoder.

The PSNR of a grayscale frame of size 256×256 is measured as the MSE value which is defined as

$$\text{MSE} = \frac{1}{256 \times 256} \sum_i \sum_j [I(i, j) - \hat{I}(i, j)]^2, \quad (4)$$

where $I(i, j)$ and $\hat{I}(i, j)$ are the pixel value of the reconstructed frame from the input of the source code's frame encoder and that at the destination's decoder, respectively. We use the following PSNR metric:

$$\text{PSNR (dB)} = 20 \log_{10} \frac{255}{\sqrt{\text{MSE}}}, \quad (5)$$

where MSE is the mean square error between the two frames.

5. Performance Analysis

In order to compare the performance of data reduction techniques, we have performed two sets of experiments. First energy and time cost of the techniques are investigated. The second tests include the comparison of the techniques in terms of quality such as recall, precision, and *F*-measure. In this section, we present energy and transmission cost of

TABLE 2: A complexity comparison of the BS algorithms for a particular pixel in terms of time complexity.

Algorithm	Time complexity
AGM	$O(M)$
AMF	$O(1)$
EGB	$O(K)$
MBG	$O(N)$
RGA	$O(1)$

the BS techniques in order to present their efficiency. In computational analysis section, we present the computational energy result of BS techniques in Android. Then, we show the transmission cost of BS techniques to reveal their benefit for Android and WMSN transmission. Additionally, we have investigated the performance of use of MPEG-4 and H.264 compression techniques in terms of power, storage, and multimedia quality.

5.1. Computational Analysis. Energy consumption, time, and storage are critical issue for resource-constrained environment in a surveillance application. Therefore, it is crucial to analyze considered data reduction technique in terms of its power consumption, storage, and time.

In this section, two different sets of performance comparisons are done. Firstly, BS techniques are analyzed in terms of energy and time and secondly the performances of compression techniques are investigated in terms of energy and storage. We would like to note that our goal is not to analyze the cost for the whole WMSNs. Instead, we focused on a single smart phone's energy consumption rates.

5.1.1. BS Techniques. BS methods generate a background model by using different techniques and algorithms. As AMF uses basic methods with an adaptive median, RGA is based on statistical methods using one Gaussian distribution. AGM utilizes statistical methods based on multiple Gaussian distributions. Eventually, EGB employs some methods which are based on eigenvalues and eigenvectors. Table 2 gives a comparison of the BS algorithms in terms of algorithm complexity (excluding any initial processing costs) [52, 53]. In this table, N is the number of frames, K is the number of best eigenvectors, and M is the number of Gaussian distributions used in the algorithm.

MBG requires N frames in order to obtain the background. Its complexity is defined as $O(N)$. AMF uses a very simple technique which compares the frame and the background model for a certain pixel. It is fast and effective with low time complexity $O(1)$. RGA utilizes statistical methods based on shape and color properties. It has fast response time with $O(1)$ complexity. AGM uses a probability density function for each pixel. It presents $O(M)$ time complexity where M is the number of Gaussian distributions per pixel. EGB method employs N images in order to calculate best eigenvectors. Finally, it shows $O(K)$ complexity per pixel where K is the number of best eigenvectors.

TABLE 3: A comparison of both energy and time results for the detection of the first foreground object in the frames.

Algorithm	Energy (J)	Time (ms)
AGM	1.17	1.93
AMF	0.65	1.00
EGB	5.7	4.26
MBG	1.59	2.76
RGA	1.30	2.19

TABLE 4: Information about the test video file.

Length	Format	Res.	Bit rate (kb/s)	Frame rate
597.22 M	mov	1920 × 1080	795466	23.98 fps

In order to measure the performance of each BS approach in terms of the processing time and energy consumption, we conducted experiments by using 100 frames. The results are provided in Table 3. This table shows that, due to its computational efficiency and simplicity, AMF method gives better performance in terms of time and energy compared to EGB. EGB algorithm requires huge memory and time in its learning phase. The classification phase of EGB requires less resource compared to the learning phase. As this learning cost leads to the additional delay and memory requirement, EGB is not an attractive method for Android devices in terms of delay and storage. AMF employs a lightweight algorithm which uses a simple approximate median filter to obtain the median. It does not need to store any frames in a buffer and modifies the obtained background model online. As AMF works much faster and requires less memory, it is suitable for Android devices. AGM algorithm gives also good performance in terms of energy and time requirement. It is improved by adapting the number of Gaussian distributions being used to model a given pixel. By means of this improvement, AGM consumes less energy and memory requirement. Considering their performances in terms of energy consumption and time, the order from the best to the worst is AMF, AGM, RGA, MBG, and then EGB.

5.1.2. Compression Techniques. In this section, we have evaluated the overhead of MPEG-4 and H.264 coding techniques in terms of power and storage by means of the Video Converter Android program. Our goal was to observe which one would be most appropriate to use when BS is in use. Specifically, we compared the required energy of these techniques for compression in Android smartphone by using PowerTutor. In this test, we use a raw test file whose basic properties are given in Table 4. This test file is compressed to Mp4 and 3gp, obtained from codecs MPEG-4 and H.264, respectively. Table 5 shows energy consumption values of the compression operation and storage values of the compressed data for different resolutions of data. In this table, energy consumption of the compression operation $E(\text{process})$ and total energy consumption $E(\text{total})$ including computational and LCD are given in the separate columns in terms of joule. The result shows that H.264 gives the smallest size along with significantly low bit rate for high-resolution compressed data.

While the resolution of the compressed data is decreasing from 640×480 Kb/s to 640×360 Kb/s, the difference between these two techniques becomes smaller in terms of storage. However, it is known that H.264 presents better video quality at significantly lower bit rates [54]. For all of the benefits of H.264, our results show that it requires more processing power compared to MPEG-4. The reason behind this is that H.264 has more sophisticated computational techniques. Although MPEG-4 looks more efficient than H.264 in terms of computational power, H.264 has several benefits such as higher video quality at lower bit rates. In order to show their suitability for WMSN applications, the performances of these two techniques are also analyzed in terms of PSNR in Section 5.3.2.

5.2. Transmission Energy Analysis. In order to decrease energy and bandwidth consumption, the size of the transmitting data should be decreased in embedded camera networks. Hence, the performance of BS techniques in terms of energy consumption is another vital issue. In this section, we present transmission energy analysis of the BS techniques so as to show their transmission energy consumption over WMSN.

As mentioned, the use of the BS algorithms leads to reductions in the size of video data, which eventually lowers the energy consumption (and bandwidth). To assess the energy gains, we conducted an experiment based on the transmission costs. Specifically, we measured the energy cost of BS algorithm and the transmission energy cost of extracted foreground in order to compare it with the transmission cost of the whole frame. Note that once the initial background of a frame is sent, there is no need to send the whole frames thereafter. Only, the foreground is sent to reconstruct the video at the receiver side. In order to measure transmission energy, we uploaded each frame to a web page. Again, the energy consumption of the battery is measured by using PowerTutor.

We picked a frame whose original size was 468 KB and the actual foreground is 2.68 KB. We run the 5 BS algorithms and obtained the foregrounds for the AGM, AMF, EGB, MBG, and RGA as 2.15 KB, 3.38 KB, 4.11 KB, 3.87 KB, and 4.62 KB, respectively. Each of these foregrounds is sent via wireless connection and the energy consumption is measured. The results shown in Table 6 show the transmission energy consumption of these foreground frames. Note that, due to similar size of the frames, the results returned from the PowerTutor are the same. While there would be minor differences if we had a more accurate way of getting energy consumption results, we argue that the energy consumption would not change significantly. We then added the cost of processing to this transmission energy to compute the total cost. These are also shown in Table 6. All of these results are compared with a baseline where the cost of sending a whole frame without any processing is considered.

Table 6 indicates that the used BS methods are at least three times energy efficient in terms of transmission. Looking at the results, we can see that AGM, AMF, and RGA are reducing the cost compared to the baseline. MBG is not efficient to be used and EGB has a lot of training overheads. Note that these results are for a single frame only. When the gain is computed for hundreds of frames, the efficiency of

TABLE 5: A comparison of both energy and storage for the considered compression techniques.

Length	Format	Res.	Bit rate	E (process)	E (total)
5.21 M	Mp4	640 × 480	6938	51.9	109.4
3.65 M	3gp	640 × 480	4858	202.7	350.3
3.18 M	Mp4	640 × 360	4236	43.7	102.3
3.10 M	3gp	640 × 360	4122	157.6	268.8
619.77 K	Mp4	176 × 144	804	33	72.5
612.04 K	3gp	176 × 144	793	47.9	103.9

TABLE 6: A comparison of energy results (in joules) for transmission of a foreground frame.

Algorithm	Energy (802.11s)	Energy (process)	Energy (total)
Whole frame	2.2	0	2.2
AGM	0.72	1.17	1.89
AMF	0.72	0.66	1.33
EGB	0.72	5.7	6.42
MBG	0.72	1.59	2.31
RGA	0.72	1.3	2.02

TABLE 7: Energy gains in joules from BS application.

Algorithm	Gain (100 frames)	Gain (1000 frames)
AGM	31	310 J
AMF	87	870
RGA	18	180

the BS algorithms will be much dramatic as seen in Table 7. Such gains are for one hop transmission only and they can be further increased when multihop transmissions are used in a WMSN.

5.3. Quality Analysis. In this section, we present the performance of BS methods and compression techniques in terms of quality, while the performances of these techniques are analyzed by precision, recall, and F -measure as well as FP.

5.3.1. BS Techniques. In addition to energy and computational efficiency, the quality of the foreground is also an important metric to assess. In this section, we conducted experiments to assess the quality of BS algorithms by considering precision, recall, and F -measure as well as FP. For this experiment, we applied the BS algorithms to 500 frames. The video data is taken from Background Models Challenge (BMC) dataset [55]. We used parameter values based on the recommendations in [15]. These values are chosen by considering the best result of the algorithms. We use the BS results of the techniques in order to compare their performance. An example of foreground objects obtained from the considered techniques is also shown in Figure 4. During evaluation of the five different algorithms introduced in Section 2, all the frames are used and average recall and precision percentages are found for each technique separately.

TABLE 8: Quality measure values of BS methods on the frames.

Algorithm	Recall	FP	Precision	F -measure
AGM	0.93	0.0037	0.45	0.60
AMF	0.84	0.0003	0.88	0.86
EGB	0.91	0.0022	0.57	0.70
MBG	0.48	0.0007	0.69	0.56
RGA	0.89	0.0018	0.62	0.73

To examine the performance of each BS algorithm, we have found the average of recall, precision, and F -measure of the obtained results for three different packet sizes (16, 64, and 256) and two different frame error rates (0.01, 0.1). While this study uses video data with 500 frames, we only depicted the frame number 250 as a background frame as shown in Figure 4(a). Its mask or ground truth is shown in Figure 4(b). Figure 4 shows a background image, its mask (ground truth), and extracted objects (foreground) using the considered algorithms.

Table 8 shows a comparison of BS algorithms in terms of precision, recall, F -measure, and FP. The results indicate that AMF achieves the best F -measure, precision, and FP results compared to the others. RGA is the second, followed by EGB, AGM, and MBG in terms of F -measure. While EGB has a higher precision, its F -measure is significantly worse than others. Moreover, AGM method shows significantly high recall because FNs are substantially decreased while TPs are increased. Nonetheless, the average F -measure of the AGM method is 26% less than that of the AMF method. Among all, MBG performs the worst in terms of F -measure, recall, and FP. Considering that AMF also provides the best performance in terms of energy consumption and time, it is the most suitable BS algorithm to be used on Android devices.

5.3.2. Compression Techniques. In this section, we have evaluated the performances of two common video encoders MPEG-4 and H.264 in terms of video quality over a single hop WMSN. In this scenario, the base station is at one hop distance away from the WMSN nodes. In the simulations, one standard video sequence named Akiyo in the YUV format of QCIF resolution (176 × 144) is transmitted in the single hop WMSN.

Figure 5 presents PSNR values of the received video frames which are decoded in MPEG-4 and H.264, respectively. MPEG-4-ref or H.264-ref PSNR values present reference PSNR values of the coded and then decoded videos

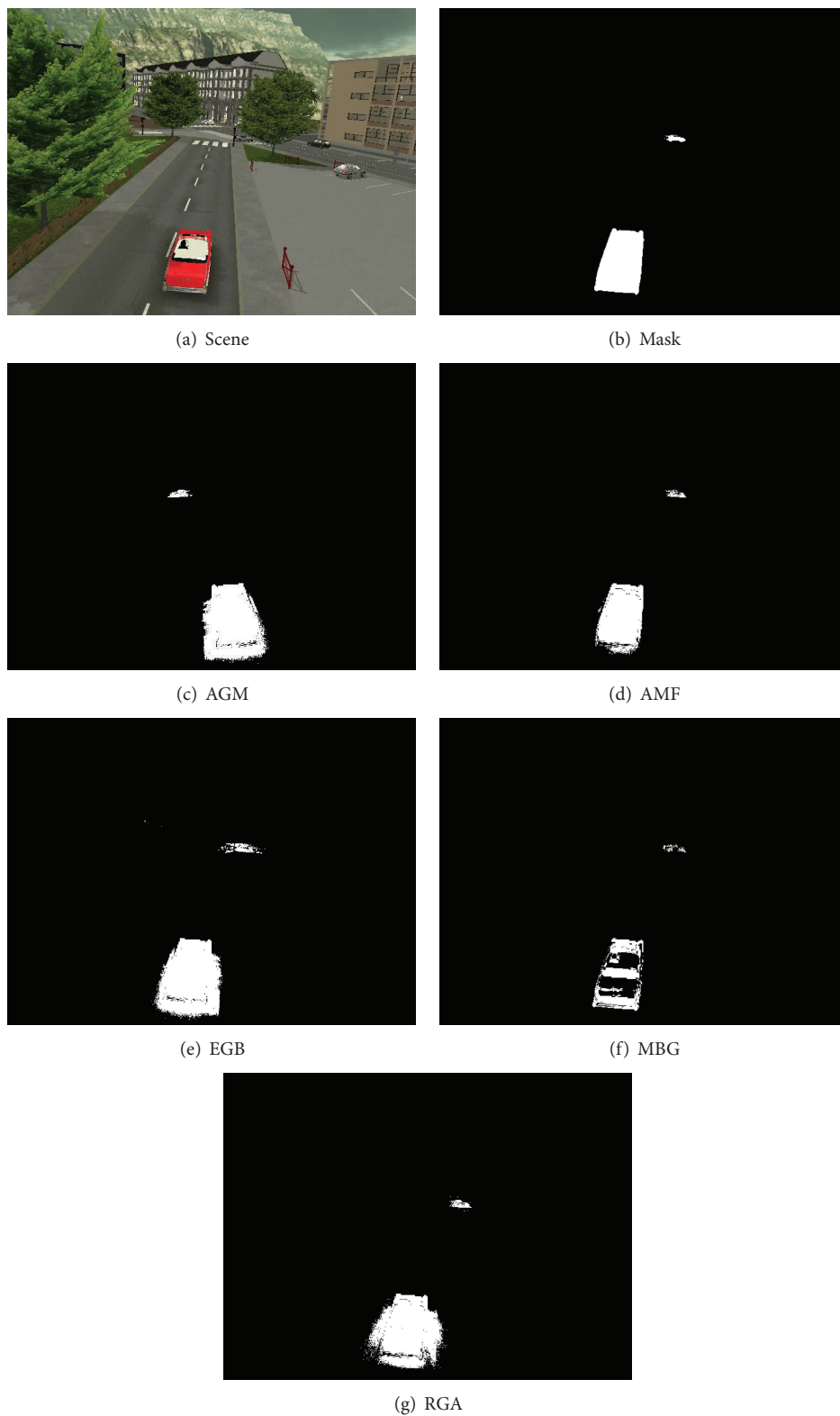


FIGURE 4: Foreground detected by five techniques compared to the actual result.

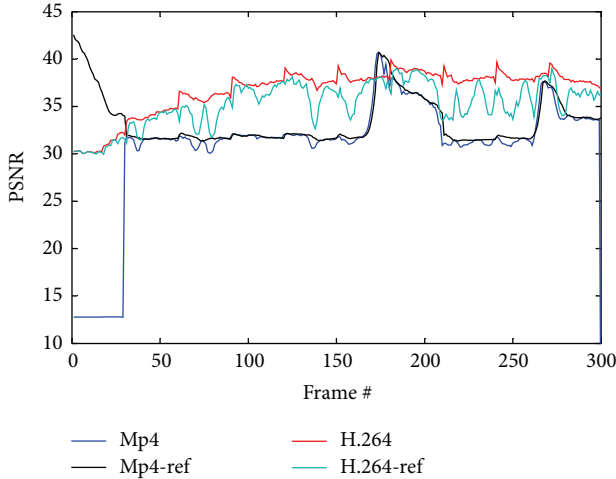


FIGURE 5: Multimedia quality comparison of the compression techniques in terms of PSNR.

without transmission errors/losses in relation to the uncoded raw video source.

The figure shows H.264 only slightly different than the reference PSNR at the beginning of the frames sent. However, the PSNR value of MPEG-4 named Mp4 is less than its reference one because of the frame loss. MPEG-4 tends to have bigger I frame than H.264 due to its compression level. As 10% of the I frames for Mp4 have been lost, there is no loss in the transmission of the H.264 frames. This has a significant effect on PSNR results.

The figure reveals that H.264 encoding techniques present better multimedia quality in single hop WMSN. H.264 includes several techniques for coding multimedia that leads to the compression efficiency. It also intends to satisfy the requirements of the multimedia applications. Hence, H.264's PSNR values are nearly 5 dB greater than MPEG-4's ones.

5.4. Network Condition Analysis. When transporting multimedia data over a wireless channel, transmitted data are exposed to losses or errors due to channel impairments. This dynamic nature of the wireless communication causes packet loss during communication. Due to such packet losses, the performance of BS algorithm can be impacted since the BS will be run on distorted frames. The investigation that is made in this section is related to the quality of the received video data at the Android phone since such quality may affect the performance of background subtraction algorithm that will be applied. For instance, if the received video frames are distorted significantly, then the quality of background subtraction may degrade.

In this section, we simulated packet losses in MATLAB by introducing various frame error rates in the channel in order to assess this impact. The frame error rate is applied on each video frame. The detailed information of distortion operation can be found in [5]. Each BS algorithm is applied on these distorted frames to evaluate the effect of the distortion on the algorithms. Additionally, the effects of the size of the packets

TABLE 9: F -measure values of BS algorithms for frame error rate = 0.01.

Algorithm	ps = 16	ps = 64	ps = 256
AGM	0.24	0.25	0.22
AMF	0.25	0.27	0.23
EGB	0.36	0.36	0.34
MBG	0.14	0.14	0.14
RGA	0.29	0.30	0.28

are considered on the performance for the multimedia data. The same MATLAB settings in [5] are also used in this paper.

5.4.1. Impact of Packet Loss on the BS Performance. We have produced these precision-recall points for different frame error rates with predefined parameters. For brevity, we do not present precision-recall curves for each test sequence but instead provide averaged results.

Figure 6(a) and Table 8 present some representatives of the performance results for nondistorted (original) frames. We conducted experiments with frame error rates of 0.1 and 0.01. We looked at the precision versus recall results as shown in Figure 6. Figure 6(a) shows that AMF is better than other methods in terms of F -measure and precision.

Figure 6(b) shows precision versus recall points for BS methods when the frame error rate and packet size are 0.01 and 16, respectively. Figure 6(b) presents the idea that while the recall values are nearly the same for all methods, the F -measure values are decreased. In this case, FPs of the algorithms increase, resulting in poorer precision. The decreasing percentages of the AGM, AMF, EGB, MBG, and RGA algorithms are 32%, 76%, 35%, 42%, and 45%, respectively.

Figure 6(c) shows precision versus recall points for BS methods when the frame error rate and packet size are 0.1 and 16, respectively. Figure 6(b) presents the idea that the high frame error rate generally leads to an increase of recall rate and decrease of the precision in the performance results. While the recall values of AMF, MBG, and EGB are slightly increasing, the recall values of RGA and AGM are nearly the same compared to Figure 6(b). However, the precision values are generally decreased due to the high frame error rate. As a result, F -measures of the algorithms are extremely decreasing depending on the distortion of the frames. The results indicate that the error rate highly affects the performance results. It implies that some preprocessing and postprocessing algorithms are required for detection of the region of interest parts of the frames.

5.4.2. Impact of Packet Size on BS Performance. In addition to frame error rate, another factor that may impact the performance is the size of the packets created for the video data. To this end, we considered three different block sizes, 4×4 , 8×8 , and 16×16 bytes, to be read from each frame for transmission. Such block size created 16-, 64-, and 256-byte packets for transmission. We conducted the same experiment as in the previous subsection. Tables 9 and 10 provide the F -measure performance of BS algorithms for different

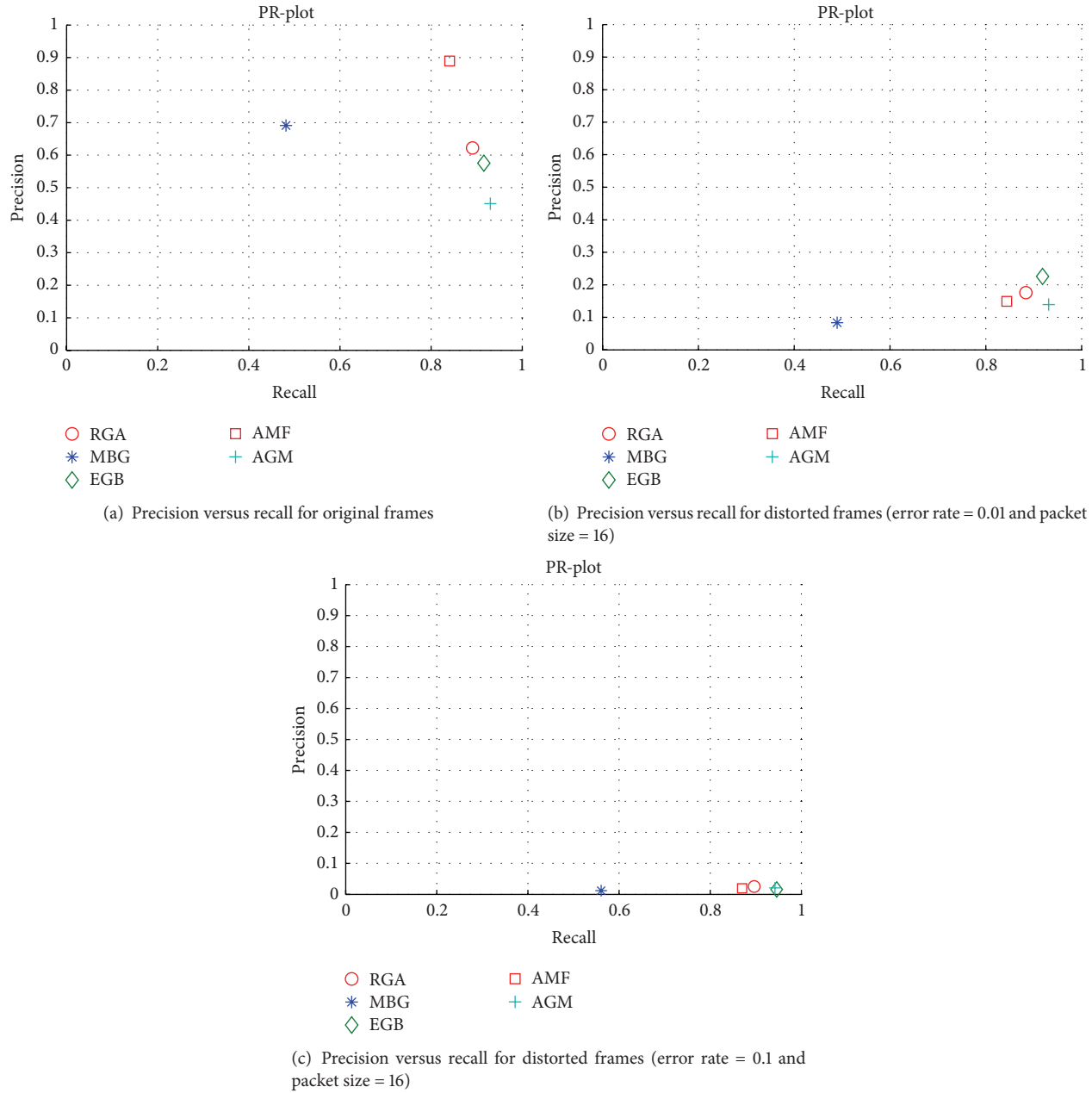


FIGURE 6: Background subtraction performances in terms of PR plots.

TABLE 10: F -measure values of BS algorithms for frame error rate = 0.1.

Algorithm	ps = 16	ps = 64	ps = 256
AGM	0.04	0.03	0.04
AMF	0.036	0.035	0.037
EGB	0.035	0.033	0.031
MBG	0.023	0.022	0.023
RGA	0.04	0.04	0.04

packets sizes where the frame error rate is 0.01 and 0.1, respectively. These results show that the packet size effect on

the F -measure performance of BS algorithms is minimal (i.e., less than 10%). When the error rate is higher, the results suggest using small packet size (e.g., 16 or 64). With the reduced error rate, the bigger packet size (e.g., 256) is better.

6. Conclusion

This paper presented a comparative evaluation of a representative group of data reduction techniques including background subtraction algorithms and compression techniques for Android devices used in WMSNs. These techniques are compared based on their energy requirements for computation and communication as well as their capability of

correctly detecting objects on both original and distorted frames. Since the frames in the database come with ground truth, precision, recall, and F -measure are used to compare the relative accuracy of the algorithms. The study investigates the energy and time requirement of five background subtraction (BS) algorithms for Android-based smart phones in WMSNs. Additionally, two well-known compression algorithms are examined in terms of energy and storage for the same Android platform. Firstly, the existing approaches have been analyzed in terms of energy, delay, and storage in an Android application. Then, the algorithms have been studied in terms of several metrics including recall, precision, and F -measure. Additionally, the impact of the packet size and frame error rate on performance of BS algorithms is evaluated with respect to wireless transmission errors. All performance results of BS are given to understand their suitability for wireless communication for video data.

The performance results of BS techniques showed that, compared to the other algorithms, AMF has been the best in terms of all the metrics compared and thus fits the Android applications the best. AMF has the least energy consumption and it is the fastest algorithm. AMF is also better in terms of F -measure and precision for nondistorted video data. The simulation results also show that the EGB, AMF, RGA, and AGM algorithms are also capable of identifying region of interest part of the distorted frames in lossy networks. However, increased frame error rate results in poorer performance in terms of F -measure. Additionally, packet length does not have an impact on performance results for varying frame error rates.

The computational evaluation results of compression techniques showed that the encoding energy consumption of MPEG-4 at the source is less than that of H.264 one at the expense of storage and multimedia quality over WMSN. The performance of MPEG-4 in terms of transmission energy also outperforms H.264 for a single hop multimedia transmission over WMSN.

Disclosure

This paper is an expanded version of a paper entitled “Performance Evaluation of Background Subtraction Algorithms for Android Devices Deployed in Wireless Multimedia Sensor Networks” presented at Wireless Communications and Mobile Computing Conference (IWCMC), 2014 International, 4–8 August, Cyprus.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

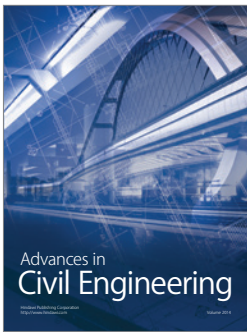
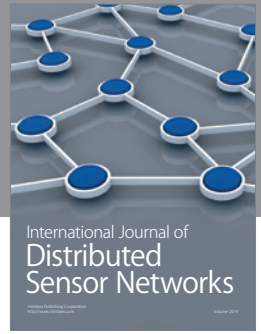
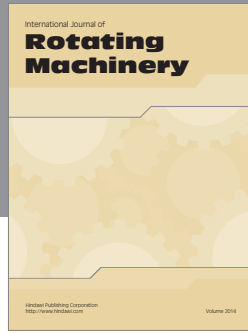
This work is supported in part by Turkish National Science Foundation (TUBITAK) under the Grant no. BIDEB-2219.

References

- [1] A. Seema and M. Reisslein, “Towards efficient wireless video sensor networks: a survey of existing node architectures and proposal for a flexi-WVSNP design,” *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 462–486, 2011.
- [2] B. Girod, V. Chandrasekhar, D. M. Chen et al., “Mobile visual search,” *IEEE Signal Processing Magazine*, vol. 28, no. 4, pp. 61–76, 2011.
- [3] P. S. Boluk, S. Baydere, and A. E. Harmanci, “Robust image transmission over wireless sensor networks,” *Mobile Networks and Applications*, vol. 16, no. 2, pp. 149–170, 2011.
- [4] S. Pudlewski and T. Melodia, “Low-complexity video streaming for wireless multimedia sensor networks,” in *The Art of Wireless Sensor Networks*, Signals and Communications Technology, pp. 529–565, Springer, Berlin, Germany, 2014.
- [5] P. Sarisaray-Boluk, “Performance comparisons of the image quality evaluation techniques in Wireless Multimedia Sensor Networks,” *Wireless Networks*, vol. 19, no. 4, pp. 443–460, 2013.
- [6] K. Akkaya and F. Senel, “Detecting and connecting disjoint sub-networks in wireless sensor and actor networks,” *Ad Hoc Networks*, vol. 7, no. 7, pp. 1330–1346, 2009.
- [7] M. Di Felice, L. Bedogni, A. Trotta et al., “Smartphones like stem cells: cooperation and evolution for emergency communication in post-disaster scenarios,” in *Proceedings of the 1st International Black Sea Conference on Communications and Networking (BlackSea-Com '13)*, pp. 28–33, IEEE, 2013.
- [8] Á. Bayona, J. C. SanMiguel, and J. M. Martínez, “Comparative evaluation of stationary foreground object detection algorithms based on background subtraction techniques,” in *Proceedings of the 6th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS '09)*, pp. 25–30, IEEE, September 2009.
- [9] S. Brutzer, B. Höferlin, and G. Heidemann, “Evaluation of background subtraction techniques for video surveillance,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1937–1944, IEEE, Providence, RI, USA, June 2011.
- [10] J. Pilet, C. Strecha, and P. Fua, “Making background subtraction robust to sudden illumination changes,” in *Computer Vision—ECCV 2008*, vol. 5305 of *Lecture Notes in Computer Science*, pp. 567–580, Springer, Berlin, Germany, 2008.
- [11] S.-C. S. Cheung and C. Kamath, “Robust techniques for background subtraction in urban traffic video,” in *Visual Communications and Image Processing*, vol. 5308 of *Proceedings of SPIE*, pp. 881–892, International Society for Optics and Photonics, San Jose, Calif, USA, January 2004.
- [12] P. Sarisaray-Boluk and K. Akkaya, “Performance evaluation of background subtraction algorithms for android devices deployed in Wireless Multimedia Sensor Networks,” in *Proceedings of the 10th International Wireless Communications and Mobile Computing Conference (IWCMC '14)*, pp. 779–784, IEEE, Nicosia, Cyprus, August 2014.
- [13] N. J. B. McFarlane and C. P. Schofield, “Segmentation and tracking of piglets in images,” *Machine Vision and Applications*, vol. 8, no. 3, pp. 187–193, 1995.
- [14] C. Lallier, E. Reynaud, L. Robinault, and L. Tougne, “A testing framework for background subtraction algorithms comparison in intrusion detection context,” in *Proceedings of the 8th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS '11)*, pp. 314–319, IEEE, Klagenfurt, Austria, September 2011.

- [15] D. H. Parks and S. S. Fels, "Evaluation of background subtraction algorithms with post-processing," in *Proceedings of the IEEE 5th International Conference on Advanced Video and Signal Based Surveillance (AVSS '08)*, pp. 192–199, IEEE, September 2008.
- [16] A. Sobral and A. Vacavant, "A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos," *Computer Vision and Image Understanding*, vol. 122, pp. 4–21, 2014.
- [17] G. S. Thakur, M. Ali, P. Hui, and A. Helmy, "Comparing background subtraction algorithms and method of car counting," <http://arxiv.org/abs/1202.0549>.
- [18] Y. Benezeth, P.-M. Jodoin, B. Emile, and H. Laurent, "Comparative study of background subtraction algorithms," *Journal of Electronic Imaging*, vol. 19, no. 3, Article ID 033003, 2010.
- [19] M. Imran, N. Ahmad, K. Khursheed, M. ONils, and N. Lawal, "Low complexity background subtraction for wireless vision sensor node," in *Proceedings of the Euromicro Conference on Digital System Design (DSD '13)*, pp. 681–688, Los Alamitos, Calif, USA, 2013.
- [20] B. Dey and M. K. Kundu, "Robust background subtraction for network surveillance in h.264 streaming video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 10, pp. 1695–1703, 2013.
- [21] S. Herrero and J. Bescós, "Background subtraction techniques: systematic evaluation and comparative analysis," in *Advanced Concepts for Intelligent Vision Systems*, pp. 33–42, Springer, 2009.
- [22] R. Trestian, A.-N. Moldovan, O. Ormond, and G. Muntean, "Energy consumption analysis of video streaming to android mobile devices," in *Proceedings of the IEEE Network Operations and Management Symposium (NOMS '12)*, pp. 444–452, IEEE, Maui, Hawaii, USA, April 2012.
- [23] J. J. Ahmad, H. A. Khan, and S. A. Khayam, "Energy efficient video compression for wireless sensor networks," in *Proceedings of the 43rd Annual Conference on Information Sciences and Systems (CISS '09)*, pp. 629–634, IEEE, March 2009.
- [24] M. A. Hoque, M. Siekkinen, and J. K. Nurminen, "Energy efficient multimedia streaming to mobile devices—a survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 579–597, 2012.
- [25] M. Akhlaq and T. R. Sheltami, "Performance comparison of video compression and streaming over wireless ad hoc and sensor networks using MPEG-4 and H.264," in *Networked Digital Technologies*, vol. 294 of *Communications in Computer and Information Science*, pp. 368–377, Springer, Berlin, Germany, 2012.
- [26] P. S. Boluk, S. Baydere, and A. E. Harmanci, "Perceptual quality-based image communication service framework for wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 14, no. 1, pp. 1–18, 2014.
- [27] I. F. Akyildiz, T. Melodia, and K. R. Chowdury, "Wireless multimedia sensor networks: a survey," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 32–39, 2007.
- [28] P. S. Boluk, K. Irgan, S. Baydere, and A. E. Harmanci, "Image quality estimation in wireless multimedia sensor networks: an experimental study," in *Broadband Communications, Networks, and Systems*, vol. 66 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 226–241, Springer, Berlin, Germany, 2012.
- [29] H. Oztarak, K. Akkaya, and A. Yazici, "Lightweight object localization with a single camera in wireless multimedia sensor networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '09)*, pp. 1–6, IEEE, 2009.
- [30] Y. Shen, W. Hu, J. Liu, M. Yang, B. Wei, and C. T. Chou, "Efficient background subtraction for real-time tracking in embedded camera networks," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems (SenSys '12)*, pp. 295–308, ACM, November 2012.
- [31] T. Imboden, K. Akkaya, and Z. Moore, "Performance evaluation of wireless mesh networks using IEEE 802.11s and IEEE 802.11n," in *Proceedings of the IEEE International Conference on Communications (ICC '12)*, pp. 5675–5679, Ottawa, Canada, June 2012.
- [32] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognition Letters*, vol. 27, no. 7, pp. 773–780, 2006.
- [33] W. Lao, J. Han, and P. H. N. de With, "Automatic surveillance analyzer using trajectory and body-based modeling," in *Proceedings of the Digest of Technical Papers International Conference on Consumer Electronics (ICCE '09)*, pp. 1–2, IEEE, Las Vegas, Nev, USA, January 2009.
- [34] N. M. Oliver, B. Rosario, and A. P. Pentland, "A bayesian computer vision system for modeling human interactions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 831–843, 2000.
- [35] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.
- [36] S.-K. Kwon, A. Tamhankar, and K. R. Rao, "Overview of H.264/MPEG-4 part 10," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 186–216, 2006.
- [37] I. E. Richardson, *H. 264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia*, John Wiley & Sons, 2004.
- [38] P. Seeling and M. Reisslein, "Video transport evaluation with H.264 video traces," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 1142–1165, 2012.
- [39] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.
- [40] L. Chiariglione, "Moving picture experts group (mpeg)," *Scholarpedia*, vol. 4, no. 2, article 6600, 2009.
- [41] I. Draft, "Recommendation and final draft international standard of joint video specification (ITU-T REC. H. 264—ISO/IEC 14496-10 AVC)," Tech. Rep. JVTG050 33, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, 2003.
- [42] M. Dai and D. Loguinov, "Analysis and modeling of mpeg-4 and h. 264 multi-layer video traffic," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, vol. 4, pp. 2257–2267, IEEE, 2005.
- [43] F. H. P. Fitzek and M. Reisslein, "MPEG-4 and H.263 video traces for network performance evaluation," *IEEE Network*, vol. 15, no. 6, pp. 40–54, 2001.
- [44] A. Ksentini, M. Naimi, and A. Guéroui, "Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture," *IEEE Communications Magazine*, vol. 44, no. 1, pp. 107–114, 2006.
- [45] R. Meier, *Professional Android 4 Application Development*, John Wiley & Sons, 2012.

- [46] S. Liang, *The Java TM Native Interface: Programmer's Guide and Specification*, Addison-Wesley Professional, 1999.
- [47] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly, 2008.
- [48] Z. Yang, *A Power Monitor for Android-Based Mobile Platforms*, EECS, University of Michigan, Ann Arbor, Mich, USA, 2012.
- [49] J. Klaue, "Evalvid—a video quality evaluation tool-set," <http://www2.tkn.tu-berlin.de/research/evalvid/fw.html>.
- [50] Y. Yang, "An evaluation of statistical approaches to text categorization," *Information Retrieval*, vol. 1, no. 1-2, pp. 69–90, 1999.
- [51] J. Lee and M. Park, "An adaptive background subtraction method based on kernel density estimation," *Sensors*, vol. 12, no. 9, pp. 12279–12300, 2012.
- [52] M. Al Najjar, S. Ghosh, and M. Bayoumi, "A hybrid adaptive scheme based on selective Gaussian modeling for real-time object detection," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '09)*, pp. 936–939, IEEE, Taipei, Taiwan, May 2009.
- [53] M. Piccardi, "Background subtraction techniques: a review," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '04)*, vol. 4, pp. 3099–3104, October 2004.
- [54] E. P. Ong, X. Yang, W. Lin et al., "Perceptual quality and objective quality measurements of compressed videos," *Journal of Visual Communication and Image Representation*, vol. 17, no. 4, pp. 717–737, 2006.
- [55] A. Vacavant, T. Chateau, A. Wilhelm, and L. Lequivre, "A benchmark dataset for outdoor foreground/background extraction," in *Computer Vision—ACCV 2012 Workshops*, J. I. Park and J. Kim, Eds., vol. 7728 of *Lecture Notes in Computer Science*, pp. 291–300, Springer, Berlin, Germany, 2013.
- [56] A. M. McIvor, "Background subtraction techniques," in *Proceedings of the Image and Vision Computing Conference*, vol. 1, pp. 155–163, Auckland, New Zealand, 2000.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

