An Application of the Inpainting Algorithm for Recovering Packet Losses from Transmitting Sequential Quad Tree Compressed Images Over Wireless Sensor Networks

P. Stavrou, G. Nikolakopoulos*, N. Fanakis, A. Tzes, and T. Theoharis

Abstract— **In this article an application of the inpainting algorithm for recovering packet losses from transmitting sequential quad tree compressed images over wireless sensor networks is presented. The aim of the proposed scheme is to reconstruct the missing information at the receiver side, based on the inpainting algorithm, instead of increasing the overhead of the network by requesting from the transceiver to re-transmit the lost data packets. The proposed architecture initially compresses the images sequentially using the quad tree decomposition algorithm. In the sequel, at the receiver side, the image is reconstructed as a lossy image, based on the available successfully received data packets, while afterwards the proposed scheme applies the image inpainting algorithm, in order to restore any missing partitions of the image. The inpainting algorithm is performed based on information derived from the the received image itself. Experimental results are presented that prove the efficacy of the proposed scheme.**

I. INTRODUCTION

The reconstruction of missing or damaged partitions of images is receiving significant attention nowadays varying, from the restoration of damaged paintings and photographs to the removal and replacement of selected objects [1–3]. Digital techniques are starting to become a widespread way of performing inpainting, ranging from attempts to fully automatic detection and removal of scratches in images, to software tools that allow for a manual more sophisticated process [4].

Moreover, the field of Wireless Sensor Networks has received significant attention in recent years with the number of applications to cover areas such as surveillance, security, monitoring of natural habitats, eco–systems and medical monitoring [5, 6]. Among the features of WSNs that made them so popular in our days are: a) the ability to construct a long–lived system that can be untethered and unattended, b) the easiness of ad–hoc deployment, c) the network's inherit reconfiguration property, d) the redundancy and heterogeneity of the network, and finally e) the rapidly decreasing cost of wireless embedded devices that constitute a WSN. All these benefits are turning this technology in a most promising one, especially in cases that involve large scale applications at geographical areas with difficult access.

P. Stavrou and T. Theoharis are with the University of Athens, Department of Informatics and Telecommunications p.stavrou(theotheo)@di.uoa.gr

G. Nikolakopoulos and A. Tzes are with the University of Patras, Department of Electrical and Computer Engineering, Rio, Achaias, 26500, Greece. gnikolak(tzes)@ece.upatras.gr

* Author to whom correspondence should be addressed.

In parallel to the improvements in the area of WSNs, multimedia applications are becoming dominant in our days, making the need for simultaneous access to multimedia content more demanding than ever. In order to cover this need, new types of fast networks and compression algorithms emerge to satisfy bandwidth needs for delivering multimedia content. From another point of view, the need for high bandwidth and burst modes of packet transmission and reception is in conflict with the communication capabilities of WSNs, which are characterized by low bandwidth, small length of data packets and asynchronous communication among nodes [7]. This situation results in a dominant need to compress the data before their transmission over the network. Such a compression algorithm is the Quad Tree Decomposition (QTD) [8, 9].

Due to the unavoidable existence of packet losses in every wireless communication channel, it is obvious that, in case we transmit an image over the WSN, it would result in a reception of a lossy image at the receiver side. If the transmitted image is decomposed by a QTD approach, in order to reduce the number of the data packets that are required to transmit a full image, then these losses would appear as orthogonal gaps in the received image with no info and of a varying size equal to a power of 2.

The aim and novelty of the presented research is to utilize the theory of inpainting in order to repair the effects of packet losses in transmitting images over Wireless Sensor Networks(WSNs). Instead of following the classical approach of analyzing the image and requesting re-transmission of the missing packets, an act that increases the overhead in the network, the inpainting algorithm is utilized to fill in, with a satisfactory estimation at the receiver side without any interaction of the transmitter, the missing information. This repair is performed by propagating information from the transmitted image.

Although the inpainting algorithm is widely utilized, its combination with compression techniques and wireless sensor nodes declare a new area of research where theories from multiple fields are combined to produce novel approaches in the area of image transmission.

This paper is structured as follows. In Section II, the System Architecture of the proposed scheme is presented, while in Sections 3,4, and 5 modules of the proposed scheme are analyzed. Finally, in Section VI experimental results, that prove the efficacy of the proposed scheme, are presented, followed by conclusions and proposals for the future work in Section VII.

II. System Architecture

The overall transmission and reconstruction scheme is presented in Figure 1 where it is shown that the received images from a camera are decomposed according to the Quad–Tree decomposition factor and afterwards, the produced decomposed image is buffered and transformed in concatenated data streams with a size equal to the buffer size that the nodes of the WSN are utilizing. This data packet

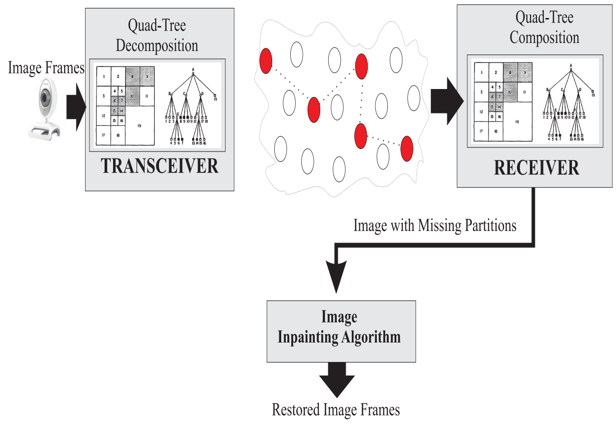

Fig. 1. System Architecture

is forwarded to the WSN's base station (at the transceiver side) that controls the transmission of data over the WSN. From the receiver side, the reverse procedure takes place. The received data packets are inserted to the Quad–Tree composition algorithm. After a complete reception of an image frame, the algorithm of image inpainting is executed in order to fill in the missing partitions, in an off–line manner, and in parallel to the process of receiving the sequential transmitted image. These missing partitions occur due to packet losses over the WSN.

III. Properties of WSN

Although sensor networks posses several characteristics of conventional networks, they also have key differences [5]. Sensor networks combine three important components: sensing, data processing and communication [6]. The nodes that comprise a sensor network are spatially distributed, energy–constrained, self configuring and self–aware. Sensors networks can provide quite effective performance in noisy environments, since they allow sensors to be placed close to signal sources, therefore yielding high Signal to Noise Ratios (SNR). Moreover, the scalability of the sensor network permits monitoring of phenomena widely distributed across space and time, and their architecture makes an ideal infrastructure for robust, reliable and self–repairing systems.

An important issue, related to scalability [12] , is the fact that, after some point, the communication becomes more expensive than computation. The requirements for collaboration and adaptation to "stochastic networking" features (usually due to exogenous factors) impose the need for the development of novel protocols dedicated to sensor networks, such as [13]. Another major concern is energy consumption, which requires a compromise between node collaboration, energy constraints [7] and affects the maximum active communication area. These features affect the routing of communication packets sent over a WSN, which require multiple hops to complete the origin to destination travel. The dynamic nature of the network further implies that the number of hops may be variable. Moreover, the wireless medium dynamics highlight that the stochastic behavior of the transmission range and the impact factor the ambient conditions may have on the latter; the transmitted signal will be propagated through the medium by different mechanisms (such as reflection and diffraction), will experience path loss due to obstacles (e.g. walls, floors, ceilings) and will finally reach its destination via multiple paths. Some of the observed events that can take place among server and client and are responsible for the insertion of packet losses in a WSN are presented in Figure 2 [14] , where the indices K,M,N,and n correspont to the number of tries for transmitting data packets from a client to a server.

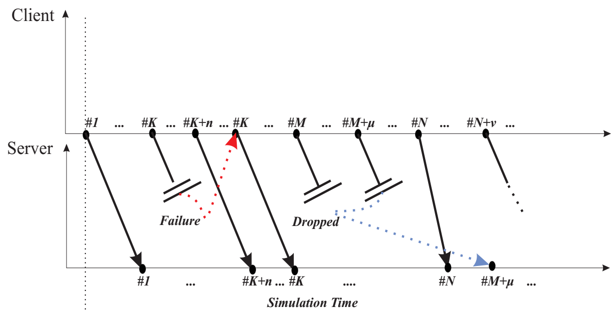

Fig. 2. Phenomena observed during a data packet transmission from Client to Server

In this paper, it is investigated how the effect of packet losses can be eliminated, based on the image inpainting algorithm, instead of requiring the transmitter to resend the missing information. The introduction of this approach can increase the total overhead of the network and will, most probably, cause a great number of packet losses due to deliberate heavy network traffic.

As it will be presented in the following two Sections, the compensation in the desiring quality of the received image frame can be adjusted by utilization of the QT–Decomposition while the restoration of the image's missing information, caused from packet losses, can be effectively simplified by incorporating the inpainting algorithm.

Such features affect the utilization of a WSN for transmitting images, while a novel reconfiguration scheme based on Quad–Tree decomposition of sequential images that increase network bandwidth, with the trade off of reducing the quality of the transmitted image, is proposed. With respect to the cases presented in Figure 2 , it should be mentioned that the most important cases for the image transmission application are the data packet failure transmission and the packet drop–

out cases, where the image data cannot be restored and appeare in the received image as empty regions.

IV. The Quad Tree Algorithm

Hierarchical data structures have become increasingly important representation techniques in the domains of computer graphics, image processing, computational geometry, geographical information systems and robotics [15]. These approaches are based on recursive computation and one of the most successful techniques is the Quad–Tree algorithm [16] , which recursively divides the image into simple geometric regions, and is the method that has been adapted in this research effort [8, 9].

Most images are stored in raster format. When an access to a raster image is made, it is sequential, starting with the top left–most pixel and ending with the bottom right–most pixel, whereas the Quad–Tree algorithm is based on a spatial order [17]. With this method, the image can be divided in half along both axes, all the way down to pixel level. For example, an image of 512x512 pixels size, would be initially divided into four 256x256 pixel regions. Another layer would have each of those regions divided into regions of 128x128 pixels, succeeding all the way down to the single pixel level. This subdividing of blocks allows for the data of the image to be organized by its neighbors. Each subdivision would exist as one of four neighbors, which is the similar case of having a tree–like structure, where the root of the tree is the entire image, then it branches out four times, and again, until its leaves are single pixels. This case is presented in Figure 3.

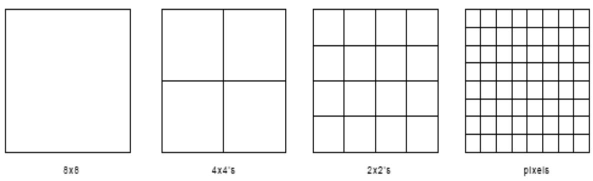

Fig. 3. Quad–Tree decomposition example

By evaluating the image like a tree, it is possible to remove unnecessary leaves and branches from the tree, resulting in the reduction of the Quad–Tree representation size [18]. This is achieved by testing whether each block meets a criterion of homogeneity. In case this criterion is satisfied, the block is not divided any further while if this criterion is not satisfied the block is moreover divided into another four blocks. The process is applied iteratively, until each block meets the homogenity criterion, and results in blocks of different sizes. The similarity criterion, can be tested and applied by utilizing the following rule:

$$max(MX_4 - AVG_4, AVG_4 - MN_4) \leq \frac{R}{L}\left(\frac{AVG_4^{(1-\frac{\gamma}{2})}}{128}\right) \quad (1)$$

Where MX is the maximum value of the four leaves of a branch, MN is the minimum value found on that branch, and AVG is the linear average of the values found on that branch. The right side is the threshold for removing the branches while R is the variable used to adjust the quality versus compression ratio tradeoff for the removal operation. The parameter L refers to a scaling factor for different types of regions. For example L would be 1 for pixels, 2 for 2 x 2s, 4 for 4 x 4s and 8 for 8 x 8s. The remainder of the right side is for displaying device gamma corrections, and because gamma is usually 2, the right side simplifies to 1/128, where 128 represents the ratio of the region to image size. For a pixel array derived from an image that is 256 x 256, it represents 1/128th of the image size, or simply, there are 128 pixel arrays in a 256 x 256 sized image. If a leave is removed, a quadrant will be represented by the average of the pixels it contained before pruning. By applying this rule, we could conclude in Quad–Tree decomposed images, of reduced size, like the Quad–Tree partitioned image that it is presented in Figure 4.

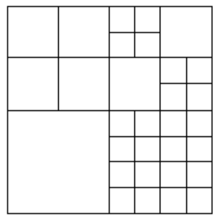

Fig. 4. Optimized Quad–Tree decomposition based on quality threshold

V. The Digital Image Inpainting Algorithm

While popular texture synthesis techniques are able to reproduce large areas of missing pixels from images, recovering packet losses resulting from image transmission over wireless sensor networks require a more subtle and natural approach. The inpainting technique proposed by Bertalmio et al. [19] is able to fill the missing pixel regions regardless of the variety of structures and backgrounds present. In addition, it imposes no limitation to the topology of the region to be repaired. Thus the selection of the inpainting algorithm is the most suitable in this scenario.

Let $I_0(i, j)$ be our discrete gray level image and Ω the region to be inpainted, where (i, j) are the pixel coordinates. As the algorithm executes, it advances through a sequence of images $I(i, j, n)$, where $I(i, j, 0) = I_0(i, j)$ and $\lim_{n\to\infty} I(i, j, n) = I_R(i, j)$ is the resulting inpainted image. At any step the algorithm can be generally described by:

$$I^{n+1}(i, j) = I^n(i, j) + \Delta t I_t^n(i, j), \forall (i, j) \in \Omega, \quad (2)$$

where n is the step, Δt is the rate of improvement and $I_t^n(i, j)$ is the update at each step.

Let $\partial\Omega$ denote the boundary of the region to be inpainted. Our goal is to smoothy propagate the missing information into Ω. In order to achieve the above we must compute the propagation direction $\vec{N}^n(i, j)$ and the information to be propagated $L^n(i, j)$ which define

$$I_t^n(i, j) = \overrightarrow{\delta L^n}(i, j) \cdot \vec{N}^n(i, j), \quad (3)$$

where $\overrightarrow{\delta L^n}(i,j)$ denotes a measure of change in the information $L^n(i,j)$. In order to achieve smoothness in the resulting image, for $L^n(i,j)$ a simple implementation of the discrete Laplacian is used:

$$L^n(i,j) = I^n_{xx}(i,j) + I^n_{yy}(i,j) \tag{4}$$

For $\overrightarrow{N}^n(i,j)$ the direction of the smallest spatial change $\nabla^{\perp} I^n(i,j)$ is used as proposed in [19]. In addition to the procedure described above an anisotropic diffusion technique is applied every few steps of the inpainting algorithm to ensure the correct evolution of the direction coefficient and reduce any noise interfering with the process. The discrete $2D$ anisotropic diffusion used in our algorithm incorporates a 3x3 pixel neighborhood to contribute information and is described by:

$$I(i,j,t+\delta t) = I(i,j,t) + \delta t \left[\sum_{k=-1,k\neq 0}^{1} I(x+k,y,t) + I(x,y+k,t) \right. \tag{5}$$
$$\left. + \frac{\sqrt{2}}{2} \left(I(x-1,y-1,t) + I(x-1,y+1,t) + I(x+1,y-1,t) + I(x-1,y+1,t) \right) \right]$$

VI. Experimental Results

For the experimental verification of the proposed scheme a Zigbee–WSN has been established, consisting of one coordinator node, three routers and one end device. The coordinator was responsible for establishing the WSN network and transmitting the decomposed images as data packets to the Zigbee network. Moreover, the routers were responsible for establishing connections within the WSN network in order to forward the decomposed image data packets to the rest of the network until their arrival to the end device. The end device was the interface of the network to the computer at the receiver side. The image inpainting algorithm was also executed in this computer.

For the presented experimental results, the benchmark image of an 8–bit grayscale image of Lena with an analysis of 256x256 pixels has been used. The test scenario includes the application of different decomposition factors on the same image and the sequential examination of: a) the effect of the packet losses on the same image, and b) the capabilities of the image inpainting alogrithm. The network coordinator, that it is presented in Figure 5 was constructed using a MaxStream XBee XB24BZigbee Modem. In order to interface the XBee modem to the camera computer a custom printed circuit board was built utilizing an FTDI FT232RL chip to interface between the XBee modem serial port and the hosts' USB port. The XBee modem was set up using the provided XBee API communication framework. The communication between the XBee modem and the computer were setup to a Baudrate of 38.4kbps using hardware flow control for the serial port.

For the rest of the network, other custom built printed circuit boards have been designed and utilized based on the same XBee modem device but using different interfaces. The router devices did not require any wired communication interface thus this was omitted providing only a power connection to them. The end device was built around the

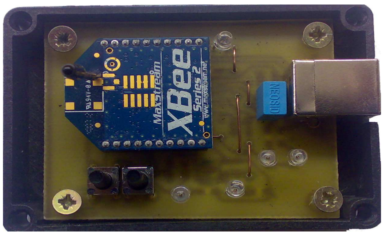

Fig. 5. The designed and implemented coordinator node of the utilized WSN

router devices custom built printed circuit board but with the addition of a serial port interface using a MAX3232 chip in order to interface it with the receiver computer. The parameters that have been utilized in the WSN for our test case are outlined in Table I.

TABLE I
EXPERIMENTAL CONFIGURATION OF THE UTILIZED WSN

Network Characteristics	Values
Number of nodes (N)	5
Coverage Area ($M \times M$)	20x20
Maximum Transmission Range (m)	40
MAC Sub-Layer Protocol	IEEE 802.15.4
Routing Protocol	ZIGBEE
Data size per Packet (B)	68
Data size per Packet(including Overhead) (B)	84
Transmission Interval(sec)	2.5
Distance between nodes(m)	5
Interface Baudrate (Kbps)	38.4
Interface flow control	Hardware (CTS/RTS)
RF Data rate (Kbps)	250
Transmit output power (mW)	1.25

The experiments were done in the following sequence. First the 256x256 pixels grayscale and 8–bit image of Lena was decomposed to its Quad–Tree equivalent with a specific decomposition factor, filling in a buffer with the decomposed image. This buffer was then divided into chunks of 68–bit data packets, accordingly to the provided XBee modem API framework. The packets were then extracted from the receiver computer to its serial port. After the recomposition phase, the received image was been checked for missing partitions. Once the transmitted image has been received and packet losses have been detected, the identified missing regions were masked in red color, as for example it is appearing at the test image of Lenna with manually inserted partitions in Figure 6(left side), and the image was input to the inpainting algorithm. The algorithm would commence the repair process using an iterative multi–resolution approach. For each lower level the image was down–scaled to half the size of the level above and the inpainting algorithm was applied to each frame. In the same Figure 6(right side) the efficacy of the inpainting algorithm are presented.

At each level the algorithm iterates by progressively filling in the missing pixels of the masked region by iteratively applying N steps of inpainting (Eq. 2) and M steps of diffusion (Eq. 5). The process is terminated after a total number of predefined steps have been completed for each

Fig. 6. The masked input image (left side) and the resulting repaired image 4–levels, $N = 12$, $N = 2$ (right side)

level of the multi–resolution approach.

In Figure 7, the first Quad–Tree decomposed frame of Lenna, with a compression factor of $L = 0.3$ (left side) and the received (19.8sec) lossy frame is also displayed in the same Figure (right hand). In Figure 8 the mask operation on the lossy image (left hand) and the results of the image inpainting (right side) are presented.

Fig. 7. QT–decomposed image ($L = 0.3$) (left side) and received lossy image (right side)

Fig. 8. Applied mask on the lossy image (left side) and results of image inpainting (right side)

In Figure 9, the second Quad–Tree decomposed frame of Lenna, with a compression factor of $L = 0.4$ (left side) and the received (13.4sec) lossy frame is also displayed in the same Figure (right side). In Figure 10 the mask operation on the lossy image (left hand) and the results of the image inpainting (right side) are presented.

Finally in Figure 11, the third Quad–Tree decomposed frame of Lenna, with a compression factor of $L = 0.6$ (left side) and the received (10.8sec) lossy frame is also displayed in the same Figure (right side). In Figure 12 the mask operation on the lossy image (left hand) and the results of the image inpainting (right side) are presented.

Fig. 9. QT–decomposed image ($L = 0.4$) (left side) and received lossy image (right side)

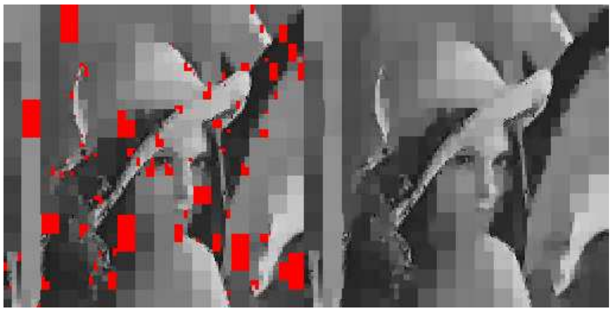

Fig. 10. Applied mask on the lossy image (left side) and results of image inpainting (right side)

Fig. 11. QT–decomposed image ($L = 0.4$) (left side) and received lossy image (right side)

From the experiments above it is obvious that the image inpainting algorithm achieves to reconstruct the missing information, at the lost partitions, for all the selected QT–decomposition factors. Smaller decomposition factors result in smoother restorations of missing partitions while bigger decompositions result in more rough restorations. This observation was expected, as the image inpainting algorithm bases its operation on the information surrounding a missing partion and there is no capability of increasing the level of detail in the reconstructions further than the provided quality of the image frame.

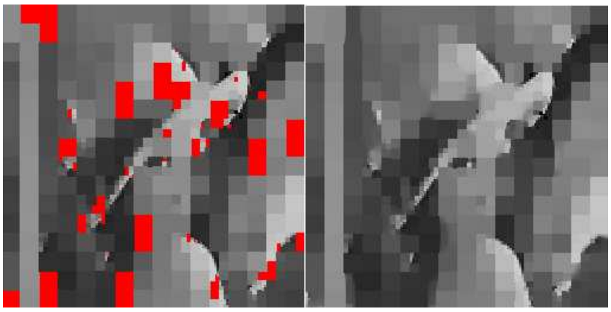

Fig. 12. Applied mask on the lossy image (left side) and results of image inpainting (right side)

By utilizing the provided methodology, we achieve less overhead in the network, as there is no need to track all the transmitted data packets or request from the transmitter to transmit the lost data packets. An advantage that is of paramount importance, especially in highly congested networks, as are the networks for exchanging multimedia content wirelessly.

VII. Conclusions

In this article an application of the inpainting algorithm for recovering packet losses from transmitting sequential quad tree compressed images over wireless sensor networks has been presented. The proposed architecture has been experimentally verified over a WSN with the benchmark image of Lenna and the efficacy of applying the inpainting algorithm to restore the missing information of lossy received images, instead of requesting re–submissions of the missing data sets, has been experimentally proven.

References

[1] A. Kokaram, R. Morris, W. Fitzgerald, P. Rayner, "Detection of Missing Data in Image Sequences," *IEEE Transactions on Image Processing*, vol. 11(4), pp. 1496–1508, 1995.

[2] A. Kokaram, R. Morris, W. Fitzgerald, P. Rayner, "Interpolation of Missing Data in Image Sequences," *IEEE Transactions on Image Processing*, vol. 11(4), pp. 1509–1519, 1995.

[3] A. Efros and T. Leung, "Texture synthesis by non-parametric sampling, *Proc. IEEE International Conference Computer Vision*, pp. 1033–1038, Corfu, Greece, September 1999

[4] C. Braverman, "Photoshop retouching handbook," *IDG Books Worldwide*, 1998.

[5] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, pp. 445–487, 2005.

[6] I. F. Akyildiz, W. Su, Y. Sankarasubramamian, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[7] E. J. Duarte-Melo and M. Liu, "Data gathering wireless sensor networks: organization and capacity," *Computer Networks*, vol. 43, no. 4, pp. 393–422, 2003.

[8] A. I. M. Kunt and M. Kocher, "Second generation image coding techniques," *Proc IEEE*, pp. 549–574, 1985.

[9] M. B. M. Kunt and R. Leonardi, "Recent results in high compression image coding," *IEEE Trans Circuits Syst*, pp. 1306–1336, 1987.

[10] IEEE Std 802.15.4, *Part 15.4: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)*, 2003.

[11] E. C. J. Gutierrez and R. Barrett, *Low-Rate Wireless Personal Area Networks: Enabling Wireless Sensors with IEEE 802.15.4*. IEEE Press, 2003.

[12] K. Langendoen and N. Reijers, "Distributed localization in wireless sensor networks: a quantitive comparison," *Computer Networks*, vol. 43, pp. 499–518, 2003.

[13] J. Zheng and M. J. Lee, *A Comprehensive Study of IEEE 802.15.4*. IEEE Press Book, 2004.

[14] G. Xylomenos and G. Polyzos, "TCP and UDP Performance over a Wireless LAN," in *Proceedings of the IEEE INFOCOM*, 1999, pp. 439–446.

[15] A. Jain, "Image data compression: A review," *Proc IEEE*, 1981.

[16] H. Samet, "The quadtree and related hierarchical data structures," *Computing Surveys*, 1984.

[17] S. Sureshchandran and P. Warter, "Algorithms and architectures for real time image compression using a feature visibility criterion," *In the Proceeding of the International Conference on Systems, Man and Cybernetics*, 1993.

[18] T. L. P. Chou and R. Gray, "Optimal pruning with applications to tree structured source coding and modeling," *IEEE Trans. Inform. Th.*, 1989.

[19] M. Bertalmio and G. Sapiro and V. Caselles and C. Ballester, "Image Inpainting," *SIGGRAPH*, 2000, pp. 417–424.