PERGAMON

Computers in Biology and Medicine 33 (2003) 277–292

Computers in Biology and Medicine

www.elsevier.com/locate/compbiomed

Confidential storage and transmission of medical image data☆

R. Norcen[a], M. Podesser[b], A. Pommer[a], H.-P. Schmidt[b], A. Uhl[a,b,*]

[a] *Department of Scientific Computing, Salzburg University, Jakob-Haringerstr. 2, Salzburg 5020, Austria*
[b] *School of Telematics & Network Engineering, Carinthia Tech Institute, Klagenfurt, Austria*

Abstract

We discuss computationally efficient techniques for confidential storage and transmission of medical image data. Two types of partial encryption techniques based on AES are proposed. The first encrypts a subset of bitplanes of plain image data whereas the second encrypts parts of the JPEG2000 bitstream. We find that encrypting between 20% and 50% of the visual data is sufficient to provide high confidentiality.

Keywords: Confidentiality for medical images; AES; Selective or partial encryption; JPEG2000

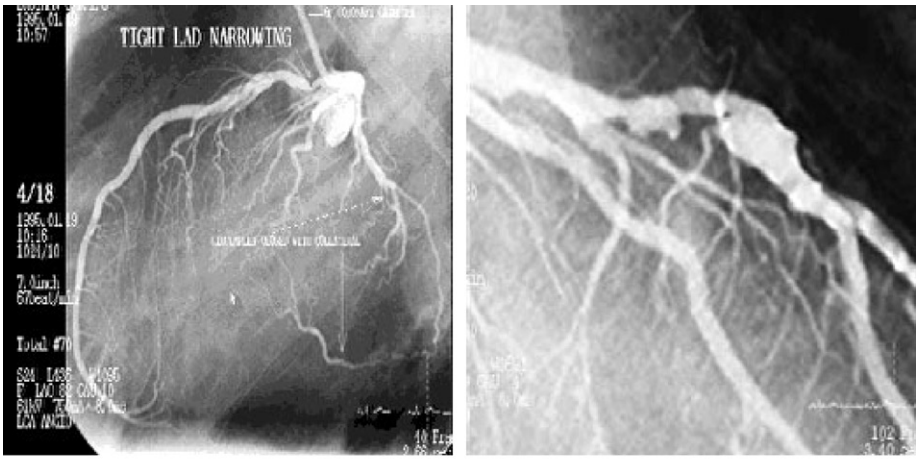
1. Introduction

The organization of todays health systems often suffers from the fact that different doctors do not have access to each others patient data. The enormous waste of resources for multiple examinations, analyses, and medical check-ups is an immediate consequence. In particular, multiple acquisition of almost identical medical image data and loss of former data of this type has to be avoided to save resources and to provide a time-contiguous medical report for each patient. A solution to these problems is to create a distributed database infrastructure where each doctor has electronic access to all existing medical data related to a patient, in particular to all medical image data acquired over the years. Additionally, many medical professionals are convinced that the future of health care will be shaped by teleradiology and technologies such as telemedicine in general. These facts show very clearly that there is urgent need to provide and protect the confidentiality of patient related medical image data when stored in databases and transmitted over networks of any kind.

The storage and transmission of medical image data differs significantly from storage and transmission of common visual data for multimedia applications. It is constrained by the fact that most

☆ This work was partially supported by the Austrian Science Fund FWF, project no. P15170.
* Corresponding author. Tel.: +43-662-8044-6303; fax: +43-662-8044-172.
E-mail address: uhl@cosy.sbg.ac.at (A. Uhl).

0010-4825/03/$ - see front matter
doi:10.1016/S0010-4825(02)00094-X



(a) Angiol (b) Angio2

Fig. 1. Testimages (angiograms) used in experiments.

radiologists are not willing to base a diagnosis on an image that has been compressed in a lossy way. This is partially due to legal reasons (depending on the corresponding country's laws) and partially due to the fear of misdiagnosis because of lost data in the compression procedure [1]. Therefore, only lossless techniques are accepted, which limits the amount of compression considerably (to a factor of about 3 in contrast to factors of 100 or more achievable in lossy schemes [2]). A possible solution to this problem is to use *selective* compression where parts of the image that contain crucial information (e.g. microcalcifications in mammograms) are compressed in a lossless way whereas regions containing unimportant information are compressed in a lossy manner [3]. In any case, we will restrict the discussion to lossless data formats.

In this work, we discuss techniques for efficient encryption of medical image data. In Section 2, we review basic terms and techniques to provide confidentiality for storage and transmission applications. Section 3 first introduces the term selective encryption and discusses application scenarios for a reasonable use of this technique. Subsequently, two concrete techniques for selective encryption of medical image data are proposed and analyzed. First, in Section 3.1 we introduce selective bitplane encryption which can be applied in environments where no compression scheme is involved. Second, selective JPEG2000 bitstream encryption is discussed in Section 3.2. Finally, we derive conclusions and compare the proposed techniques.

In Fig. 1 we display the testimages used in our experiments. We have decided to use angiograms since they represent an important class of medical image data (e.g. [4]).

In order to make the explanations and experiments of the proposed techniques simpler, we assume 512×512 pixels images to be given in 8 bit/pixel (bpp) precision. Extensions to images of different acquisition types (e.g. [5]), higher bitdepth or non-squared format are straightforward.

2. Principles of confidential storage and transmission of visual data

Images and videos (often denoted as visual data) are data types which require enormous storage capacity or transmission bandwidth due to the large amount of data involved. In order to provide

reasonable execution performance for encrypting such large amounts of data, only symmetric encryption (as opposed to public key cryptography) can be used. As done in most current applications with demand for confidentiality, public key techniques are used for key exchange or signature generation only (such schemes are usually denoted as "hybrid"). The Advanced Encryption Standard (AES) [6] is a recent symmetric block cipher which is going to replace the Data Encryption Standard (DES) in all applications where confidentiality is really the aim. AES operates on 128-bit blocks of data and uses 128, 196, or 256 bit keys. For more information including links to various source code see the official NIST AES-page http://csrc.nist.gov/encryption/aes/. We use AES as the basic cryptographic building block in all techniques described.

Symmetric block ciphers in general and AES in particular may be operated in different modes. The simplest case, Electronic Codebook Mode (ECB), encrypts a block of plaintext data M_i to the corresponding block of ciphertext data C_i. The main advantage of the independent encryption and decryption of blocks is that no order among blocks needs to be maintained during processing (which is good for database applications or parallel processing). On the other hand, the simple structure makes this mode vulnerable to certain types of attacks (e.g. block replay or codebook attacks). To overcome these limitations, Cipher Block Chaining (CBC) may be used. Here, a block of plaintext is encrypted in dependence of the preceding block of ciphertext, i.e. C_i is obtained by encrypting $P_i \oplus C_{i-1}$. As a consequence, a block cannot be decrypted without having decrypted all preceding ones and therefore its ciphertext depends on these blocks. For certain applications, the restriction to a certain block-size may not be appropriate. Besides the use stream ciphers for such applications, block ciphers may be operated in Cipher FeedBack (CFB) mode to satisfy the requirement for encryption of arbitrary sized data. CFB uses a queue onto which the block cipher is applied as required. Initially, the queue is filled with random data and the queue is encrypted. Subsequently, encrypted data is retrieved from the left side of the queue and XORed with the plaintext data. The resulting ciphertext bits are on the one hand stored or transmitted, on the other hand fed into the queue from the right side. Then the queue is encrypted again and the system is ready for the next plaintext bit(s). For more details on block cipher modes and their corresponding advantages and disadvantages see [7].

There are two ways to provide confidentiality to a storage or transmission application. First, confidentiality is based on mechanisms provided by the underlying computational infrastructure. The advantage is complete transparency, i.e. the user or a specific application does not have to take care about confidentiality. The obvious disadvantage is that confidentiality is provided for all applications, no matter if required or not, and that it is not possible to exploit specific properties of certain applications. To give a concrete example, consider the distributed database infrastructure mentioned in the introduction. If the connections among the components are based on TCP/IP internet connections (which are not confidential by itself of course), confidentiality can be provided by creating a Virtual Private Network (VPN) using IPSec (which extends the IP protocol by adding confidentiality and integrity features). In this case, the entire visual data is encrypted for each transmission which puts a severe load on the encryption system. The second possibility is to provide confidentiality is on the application layer. Here, only applications and services are secured which have a demand for confidentiality. The disadvantage is that each application needs to take care for confidentiality by its own, the advantage is that specific properties of certain applications may be exploited to create more efficient encryption schemes or that encryption is omitted if not required. Selective encryption of medical image data takes advantage of the redundancy in visual data and is therefore classified into the second category.

3. Selective encryption of medical image data

In the area of multimedia security, the terms "selective encryption" (SE) or "partial encryption" denote techniques which trade off security for computational complexity. They are designed to protect multimedia content and fulfil the security requirements for a particular multimedia application. For example, real-time encryption for an entire video stream using classical ciphers requires much computation time due to the large amounts of data involved. If we assume a database of medical images as described in the introduction, several 100 requests of this type need to be served concurrently which obviously puts severe demands on the encryption process. Therefore, a reduction of computational demand is desirable for this application. The same is true if components with low processing power are involved in a teleradiology application, e.g. using mobile image capturing clients. In selective encryption of visual data, application specific data structures are exploited to create more efficient encryption systems (see e.g. encryption of MPEG video streams [8]). Consequently, selective encryption only protects the visually most important parts of an image or video representation relying on a secure but slow "classical" cipher. The first attempts in this direction have been made to secure DCT-based multimedia representations (see e.g. [8–17]), wavelet based [18–21,17] and quadtree based representations [22,23] have been considered also. Recently, selective encryption schemes based on selective bitplane encryption [24], resistant to bit errors [25], and compliant to video formats [26] have been proposed for wireless environments.

Intuitively, SE seems to be a good idea in any case since it is always desirable to reduce the computational demand involved in signal processing applications. However, the security of such schemes is always lower as compared to full encryption. The only reason to accept this drawback are *significant* savings in terms of processing time or power. Therefore, the environment in which SE should be applied needs to be investigated thoroughly in order to decide whether its use is reasonable or not. In the following we discuss scenarios for the reasonable use of SE of visual data in medical applications and therefore restrict the discussion to lossless data formats. See [27] for a comprehensive discussion of application scenarios for lossless and lossy environments.

We may distinguish between two types of SE depending on whether the image data are given as plain image data (scenario data) or in form of a bitstream resulting from prior compression (scenario bitstream). For example, in on-line applications the plain image data may be accessed directly after being captured by a digitizer before being compressed. On the other hand, as soon as visual data has been stored or transmitted once it has been compressed in some way which is true for most off-line applications. Note that whenever compression is involved it has always to be performed prior to encryption since the statistical properties of encrypted data prevent compression from being applied successfully. Moreover, the reduced amount of data after compression decreases the computational demand of the subsequent encryption stage.

In the following analysis, t denotes the time required to perform an operation, E is the encryption function, SE the selective encryption function, C is compression, P is the preprocessing involved in the selective encryption scheme (where P means the extraction of relevant features), and $\gg$ means significantly larger. Please note that the processing time t is not equivalent to computational complexity: for example, if compression is performed in hardware and encryption in software, the time required for compression will be considerably lower as for encryption, contrasting to the relation if both operations are performed in software.

Scenario data: Encryption is applied directly to the raw image data I. The following condition must be fulfilled in order to justify the use of SE:

$$t(E(I)) \gg t(P) + t(SE(I)). \tag{1}$$

Note that if $t(C(I)) + t(E(C(I))) < t(E(I))$, the image data is compressed using a lossless codec and the considerations of scenario bitstream (see below) apply. Usually, this is not the case since $t(C(I)) \gg t(E(I))$ is true for almost all high quality lossless codecs and symmetrical ciphers. Additionally, the data reduction of lossless schemes is much lower as compared to lossy ones making the contribution of $t(E(C(I)))$ significant as well. When applying encryption to the raw image data, P denotes the identification of relevant features in the raw image data which may be done in various ways (see next section for an example). However, it is crucial that $t(P)$ is not too large to satisfy Eq. (1). If $t(P)$ can be made small, SE is a reasonable approach in this setting. A concrete sample technique for this scenario is selective bitplane encryption as discussed in Section 3.1.

Scenario bitstream: Given the bitstream B resulting from prior compression, the following condition must be fulfilled in order to justify the use of SE:

$$t(E(B)) \gg t(P) + t(SE(B)). \tag{2}$$

In this case, P is the identification of relevant features in the compressed bitstream. Depending on the type of bitstream, $t(P)$ may range from negligibly small to a considerably large amount of time. If the bitstream is embedded or is composed of several quality layers, the identification of parts subjected to SE is straightforward ($t(P) = 0$)—the first part of the embedded bitstream or the base layer is encrypted only. A concrete sample technique for this scenario is selective encryption of a JPEG 2000 bitstream as discussed in Section 3.2.

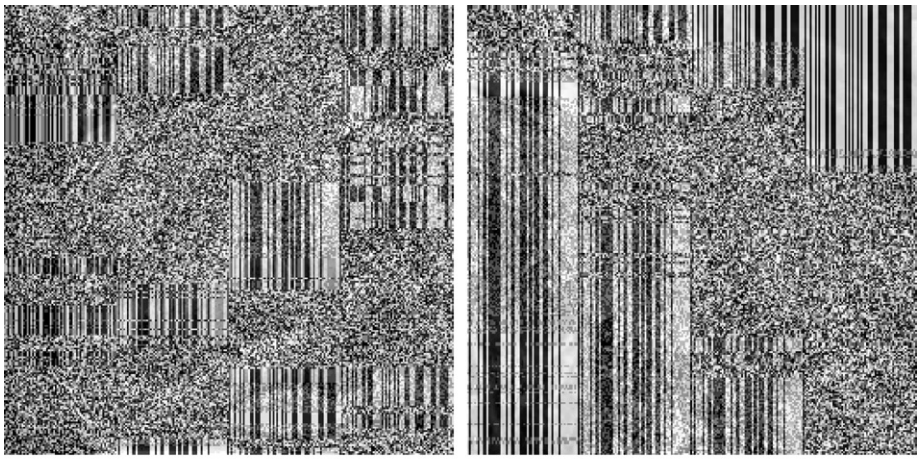
Note that if instead of the bitstream B the raw image data I is given initially in the scenario bitstream, the image data I needs to be compressed first. In this case, the condition for a reasonable use of SE changes to

$$t(C(I)) + t(E(C(I))) \gg t(C(I)) + t(P) + t(SE(C(I))). \tag{3}$$

This condition is very hard to satisfy, since $t(C(I)) \gg t(E(C(I)))$ holds for most compression schemes and symmetrical ciphers if both schemes are executed in software. Therefore, the difference between $t(E(C(I)))$ and $t(SE(C(I)))$ often does not matter in practice and therefore the decrease in terms of security often does not justify the marginal savings in processing time as achieved by SE in a software based system of this type. If compression is done in hardware and encryption in software, the situation is entirely different of course and SE makes sense.

3.1. Selective bitplane encryption

We assume a target environment, where due to the low processing power of the involved hardware compression and decompression of visual data is not reasonable or possible (e.g. mobile clients). Additionally, as bandwidth increases, and therefore it becomes relatively cheaper, the requirement for compression becomes a less stringent condition, which is especially true for lossless applications. The reason is that the data reduction of lossless compression schemes is much lower as compared to lossy ones making the respective application less profitable. In applications where image data is acquired the plain image data may be accessed directly after being captured by a digitizer without



(a) Angio1, 8.63 dB (b) Angio2, 7.38 dB

Fig. 2. Visual examples for selective encryption of two bitplanes, direct reconstruction.

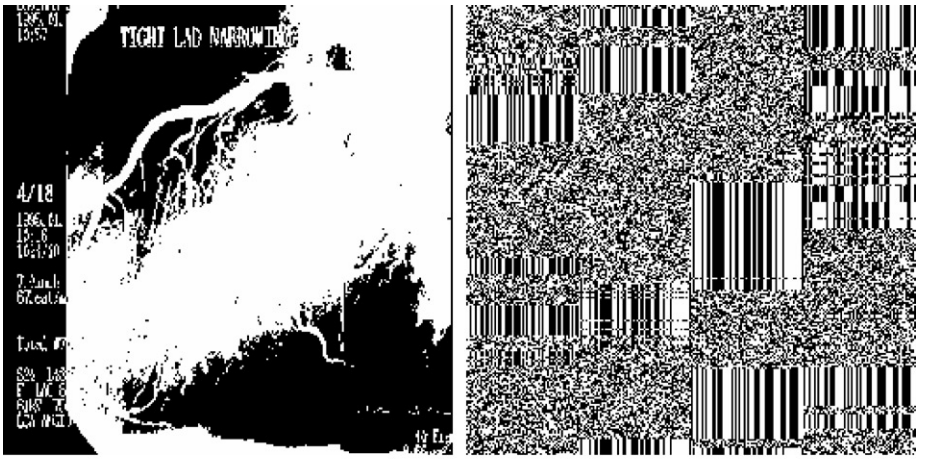
being compressed. For example, we may assume the images to be captured by a hand-hold device and subsequently transmitted. A concrete sample application for this scenario is teleradiology with mobile image capturing clients and wireless image transmission to enable fast and exact on-site diagnosis after an accident.

We consider the 8 bpp image data in the form of 8 bitplanes, which means that we slice the image into 8 binary images (i.e. bitplanes), the values of each bitplane are composed of the bitvalues in the corresponding position of the binary representation of the pixels. The SE approach is to AES encrypt a subset of the bitplanes only, starting with the bitplane containing the most significant bit (MSB) of the pixels. Each possible subset of bitplanes may be chosen for SE, however, the minimal percentage of data to be encrypted is 12.5% (when encrypting the MSB bitplane only), increasing in steps of 12.5% for each additional bitplane encrypted. Note that it is of course important to encrypt the MSB first and continue with the bitplanes corresponding to the next bits in the binary representation. The computational effort for preprocessing (P, i.e. accessing the bits in the binary representation of the image data) is negligible, therefore we may assume $t(P) = 0$ making SE a profitable approach. The encrypted bitplanes are transmitted together with the remaining bitplanes in plain text.

We use an AES implementation in ECB mode with blocksize 128 bit and a 128 bit key. Each 128 bit block is filled with a quater of a bitplane line ($512/4 = 128$ bits).

Fig. 2 shows as example both directly reconstructed angiograms after selectively encrypting 2 bitplanes (i.e. 25% of the original data have been encrypted). Whereas in the case of encrypting the MSB only structural information is still visible, encrypting two bitplanes leaves no useful information in the reconstruction, at least when directly reconstructing the image data.

Note the pattern reminiscent of a bar code at certain positions in the images. Fig. 3 shows the original and encrypted MSB bitplane of Angio1 where this pattern is exhibited even clearer. This phenomenon due to the fact that AES encryption is used with identical key for all blocks in ECB mode. Consequently, if there are identical plain text quater-lines directly situated above each other which also adhere to the AES block-border (i.e. starting at pixel positions 0, 128, 256, or 384),



(a) MSB bitplane (b) encrypted bitplane

Fig. 3. Plaintext and encrypted MSB bitplane of Angio1.

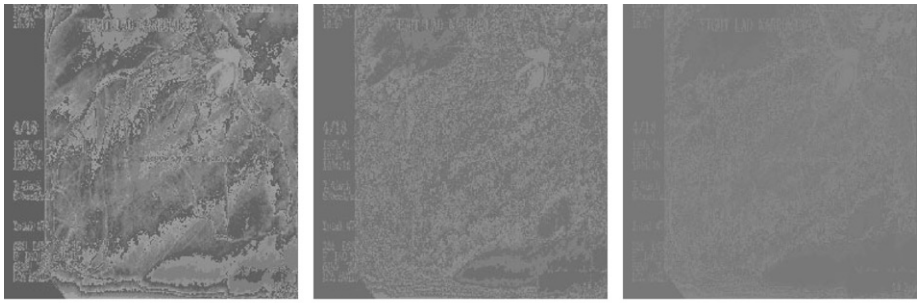
these data produce identical ciphertext blocks. Identical blocks of ciphertext are again arranged as identical quater-lines thereby generating the barcode effect. This effect can be avoided by using AES in CBC mode.

In the following, we want to assess the security of selective bitplane encryption by conducting a simple ciphertext-only attack. As seen in Fig. 2, when directly reconstructing the selectively encrypted images the encrypted parts introduce noise-type distortions. Therefore, an attacker would replace the encrypted parts by artificial data mimicking typical images ("replacement attack", see also [24]). In particular, an encrypted bitplane is replaced by a constant 0 bitplane and the resulting decrease in average luminance is compensated by adding 64 to each pixel if only the MSB bitplane was encrypted, 96 if the MSB and next bitplane have been encrypted, and so on. Subsequently, reconstruction is performed as usual, treating the encrypted and replaced parts as being non-encrypted.

Fig. 4 shows three visual examples of image reconstructions as obtained by the replacement attack (2–4 bitplanes are encrypted). Whereas a direct reconstruction of an image with 2 bitplanes encrypted suggests this setting to be "safe" (with 8.63 dB quality, see Fig. 2a), the replacement attack reveals that structural information and text is still present in the reconstructed image (with 12.42 dB quality, see Fig. 4a). However, the visual information is severely alienated. Obviously, not only the visual appearance but also the numerical PSNR values have been significantly improved by the replacement attack. In any case, even if a replacement attack is mounted, encrypting 4 bitplanes (i.e. 50% of the original data) leads to perfectly satisfying confidentiality (Fig. 4c).

3.2. Selective encryption of the JPEG2000 bitstream

Wavelet-based image processing methods in general have gained much attention in the biomedical imaging community. Applications range from pure biomedical image processing techniques such as noise reduction, image enhancement, and detection of microcalcifications in mammograms to



(a) 2 bitplanes, 12.42 dB (b) 3 bitplanes, 12.13 dB (c) 4 bitplanes, 12.02 dB

Fig. 4. Replacement attack against selective bitplane encryption of Angio1.

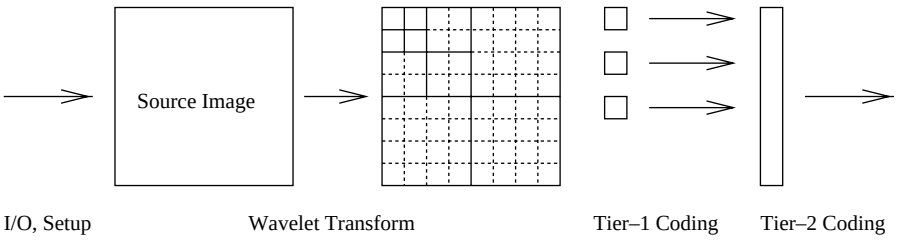


Fig. 5. JPEG2000 coding pipeline.

computed tomography (CT), magnetic resonance imaging (MRI), and functional image analysis (positron emission tomography (PET) and functional MRI) [28,29].

Image compression methods that use wavelet transforms [30] (which are based on multiresolution analysis—MRA) have been successful in providing high compression ratios while maintaining good image quality. Therefore, they have replaced DCT based techniques in recent standards for still image coding: JPEG2000 [31] and VTC (visual texture coding in MPEG-4 [32]). As JPEG2000 also offers a lossless mode fully compatible with the lossy one, JPEG2000 will replace lossless JPEG and other lossless compression standards in medical applications in the near future.

The JPEG2000 image coding standard is based on a scheme originally proposed by Taubman and known as EBCOT ("Embedded Block Coding with Optimized Truncation" [33]). The major difference between previously proposed wavelet-based image compression algorithms such as EZW or SPIHT (see [30]) is that, after performing a global wavelet transform, EBCOT as well as JPEG2000 operate on independent, non-overlapping blocks of transform coefficients which are coded in several bit layers to create an embedded, scalable bitstream. Instead of zerotrees, the JPEG2000 scheme depends on a per-block quad-tree structure since the strictly independent block coding strategy precludes structures across subbands or even code-blocks. These independent code-blocks are passed down the "coding pipeline" shown in Fig. 5 and generate separate bitstreams. The wavelet coefficients inside a code-block are processed from the most significant bitplane towards the least significant. Furthermore, in each bitplane the bits are scanned in a maximum of three passes called coding passes. Finally, during each coding pass, the scanned bits with their context value are sent to a

context-based adaptive arithmetic encoder that generates the code-block's bitstream. This procedure is called Tier-1 encoding.

The rate-distortion optimal merging of these bitstreams into the final one is based on a sophisticated optimization strategy and is called Tier-2 encoding. This last procedure carries out the creation of the so-called layers which roughly stand for successive qualities at which a compressed image can be optimally reconstructed. These layers are built in a rate-allocation process that collects, in each code block, a certain number of coding-passes codewords. Hence, in a code-block, the bitstream is distributed into a certain number of layers. The final JPEG2000 bitstream is organized as follows. The main header is followed by packets of data which are all preceded by a packet header. In each packet appear the codewords of the code-blocks that belong to the same image resolution and layer, the header identifies the data. Depending on the arrangement of the packets, different progression orders may be specified. Among others, resolution and layer progressive are most important for grayscale images. In layer progression order, the packets corresponding to the first layer are arranged first and cover data contained in all resolutions, followed by packets corresponding to the second layer and so on. Vice versa, in resolution progression order the packets corresponding to the first resolution level are arranged first (these contain data of all layers).

For selectively encrypting the JPEG2000 bitstream we have two general options. First, we do not care about the structure of the bitstream and simply encrypt a part, e.g. the first 10% of the bitstream. In this case, the main header and a couple of packets including packet header and packet data are encrypted. Since basic information necessary for reconstruction usually located in the main header is not available at the decoder, encrypted data of this type cannot be reconstructed using a JPEG2000 decoder. Although this seems to be desirable at first sight, an attacker could reconstruct the missing header data using the unencrypted parts, and, additionally, no control over the quality of the remaining unencrypted data is possible. Therefore, the second option is to design a JPEG2000 bitstream format compliant encryption scheme which does not encrypt main and packet header but only packet data. This is what we propose in the following.

In order to achieve format compliance, we need to access and encrypt data of single packets. Since the aim is to operate directly on the bitstream without any decoding we need to discriminate packet data from packet headers in the bitstream. This can be achieved by using two special JPEG2000 optional markers which were originally defined to achieve transcoding capability, i.e. manipulation of the bitstream to a certain extent without the need to decode data. Additionally, these markers of course increase error resilience of the bitstream. These markers are "start of packet marker" (SOP - 0xFF91) and "end of packet marker" (EPH - 0xFF92). The packet header is located between SOP and EPH, packet data finally may be found between EPH and the subsequent SOP. For example, using the official JAVA JPEG2000 reference implementation (JJ2000—available at http://jj2000.epfl.ch) the usage of these markers may be easily invoked by the options `-Peph on -Psop on`.

Having identified the bitstream segments which should be subjected to encryption we note that packet data is of variable size and does not at all adhere to multiples of a block ciphers block-size. Consequently, ECB and CBC encryption modes cannot be applied and we have to employ AES in CFB mode for encryption. Information about the exact specification of the cryptographic techniques used (e.g. key exchange) may be inserted into the JPEG2000 bitstream taking advantage of so-called termination markers. Parts of the bitstream bounded by termination markers are automatically ignored during bitstream processing and do not interfere with the decoding process. Note that a JPEG2000 bitstream which is selectively encrypted in the described way is fully compli-

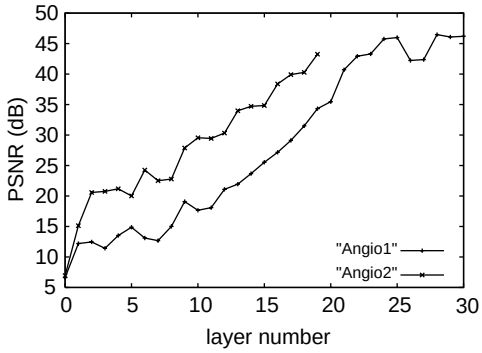


Fig. 6. Effect of encrypting two consecutive layers at different positions in the bitstream.

ant to the standard and can therefore be decoded by any codec which adheres to the JPEG2000 specification.

In the first experiment we encrypt all packets corresponding to two consecutive layers (i.e. these packets contain all resolutions for these two layers) in the bitstream, reconstruct the resulting bitstream and measure the PSNR quality of the resulting images. Note, that the higher the PSNR values, the better is the images' quality. Consequently, in order to make selective encryption efficient, we look for low PSNR quality (since encrypted images should be of low quality of course). In Fig. 6 we clearly note that image quality is low (i.e. that encryption has the desired effect) if two of the first layers are encrypted, whereas PSNR increases steadily if layers of higher order are encrypted. This exactly corresponds to the desired properties of an embedded progressive bitstream where important information is arranged and transmitted first. As a consequence, when using a selective encryption approach, we always have to encrypt the first packets in the JPEG2000 bitstream first.

In the following, we want to investigate whether resolution progressive order or layer progressive order is more appropriate for selective JPEG2000 bitstream encryption. We arrange the packet data in either of the two progression orders, encrypt an increasing number of packet data bytes, reconstruct the images and measure the corresponding quality.

Interestingly, we obtain different results for the two angiograms in Fig. 7. Whereas layer progression is more suited for selectively encrypting Angio1, resolution progression is superior for Angio2. In order to relate the numerical values to visual appearance, two reconstructed versions of Angio1, corresponding to the two progression orders, are displayed in Fig. 8. In both cases, 1% of the entire packet data has been encrypted.

Here, the visual appearance corresponds well to the numerical PSNR values. Whereas no details are visible using layer progression (Fig. 8a at 7.28 dB), textual information and even high frequency visual information is visible using resolution progression (Fig. 8b at 9.1 dB). In contrast, as can be seen in Fig. 9, the reconstruction with higher PSNR values (Fig. 9a at 8.79 dB, layer progression) does not reveal any useful information whereas some text may be identified at the lower right corner even at 7.45 dB in Fig. 9b using resolution progression.

Please note also the difference in coarseness of the noise pattern resulting from encryption between resolution and layer progression. Since in resolution progression data corresponding to the higher levels of the wavelet transform is encrypted, the noise introduced by the cipher is propagated by the

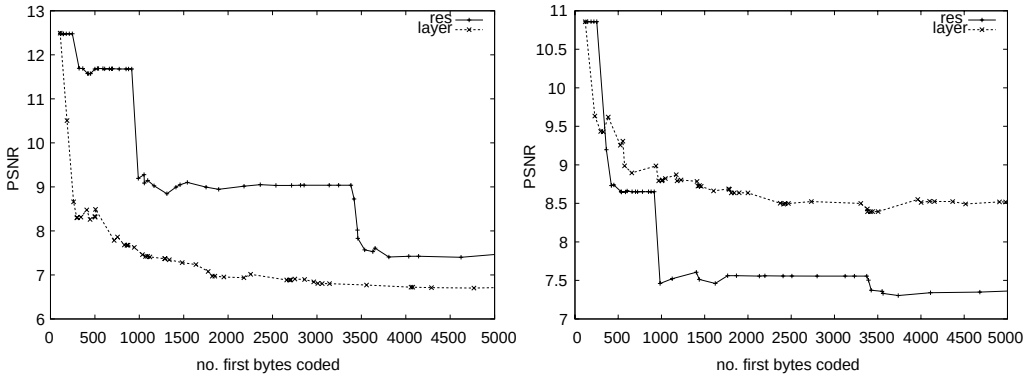
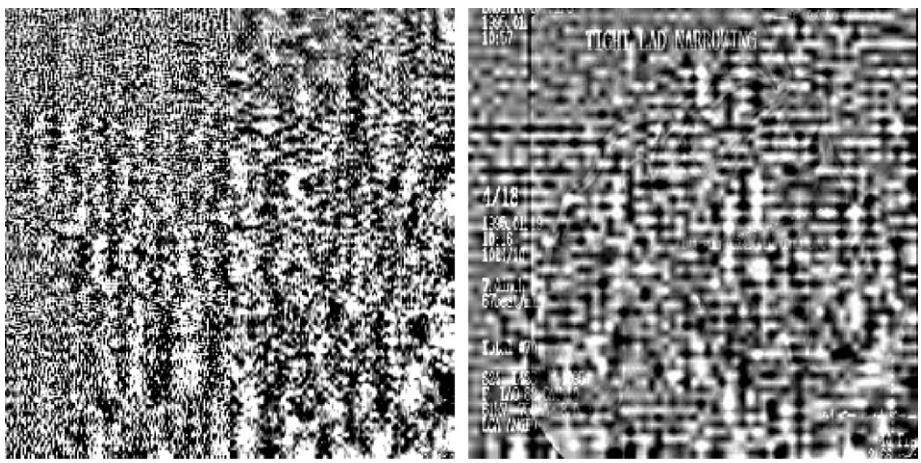


Fig. 7. Comparison of selective encryption (PSNR of reconstructed images) using resolution or layer progressive encoding.

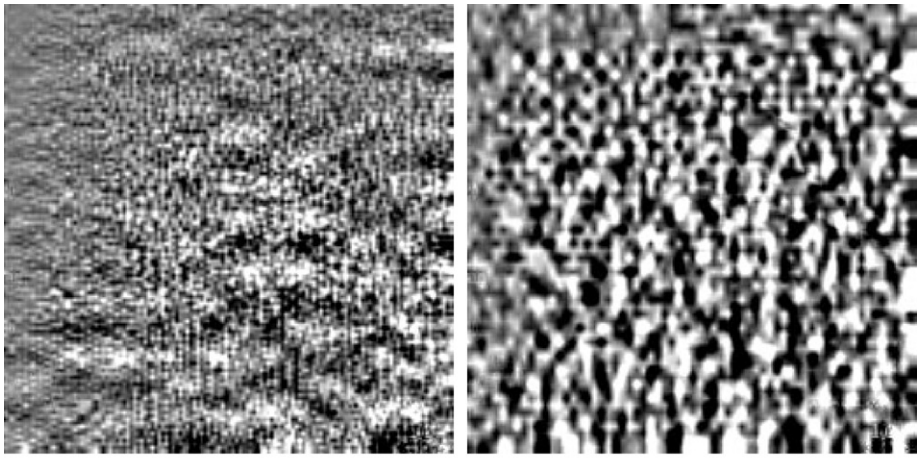


(a) layer progressive, 7.28 dB (b) resolution progressive, 9.10 dB

Fig. 8. Comparison of selective encryption (visual quality of reconstructed Angio1 where 1% of the bitstream data have been encrypted) using resolution or layer progressive encoding.

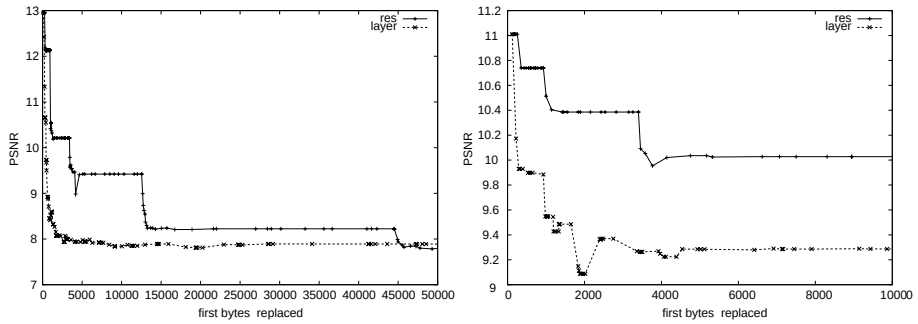
repeated inverse transform and thereby magnified resulting in a much coarser pattern as compared to layer progression. When summarizing numerical and visual results, it seems that encrypting 1–2% of the packet data in layer progressive mode is sufficient to provide confidentiality for the JPEG2000 bitstream. This is a very surprising result of course.

Similar to the last section, we want to assess the security of the scheme in the following. However, the replacement attack cannot be mounted in the same way since replacing the encrypted parts by some constant bits does not have the desired effect, since these values are arithmetically decoded and the corresponding model depends on earlier results and corrupts the subsequently required states. Therefore, the reconstruction result is a noise-like pattern similar as obtained by directly reconstructing the encrypted bitstream. Again, we exploit a built-in error resilience functionality in JJ2000 to



(a) layer progressive, 8.79 dB (b) resolution progressive, 7.45 dB

Fig. 9. Comparison of selective encryption (visual quality of reconstructed Angio2 where 1% of the bitstream data have been encrypted) using resolution or layer progressive encoding.



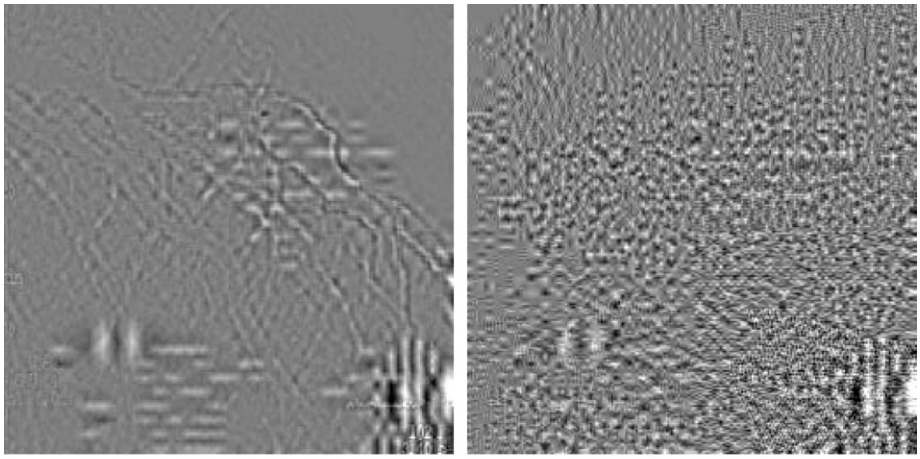
(a) Angio 1 (b) Angio 2

Fig. 10. PSNR of reconstructed images after replacement attack using resolution or layer progressive encoding.

simulate a bitstream-based replacement attack. An error resilience segmentation symbol in the code-words at the end of each bit-plane can be inserted. Decoders can use this information to detect and conceal errors. This method in invoked in JJ2000 encoding using the option `-Cseg_symbol on`.

If an error is detected during decoding (which is of course the case if data is encrypted) it means that the bit stream contains some erroneous bit that have led to the decoding of incorrect data. This data affects the whole last decoded bit-plane. Subsequently, the affected data is concealed and no more passes should be decoded for this code-block's bit stream. The concealment resets the state of the decoded data to what it was before the decoding of the affected bit-plane started. Therefore, the encrypted packets are simply ignored during decoding.

Using this technique, we again compare selective JPEG2000 encryption using resolution and layer progressive mode by reconstructing images with a different amount of encrypted packets. Decoding



(a) 1% encrypted, 10.51 dB (b) 20% encrypted, 9.90 dB

Fig. 11. Visual quality of reconstructed Angio2 after replacement attack using resolution encoding.

is done using error concealment. In Fig. 10 we immediately recognize that the PSNR values are significantly higher as compared to directly reconstructed images (see Fig. 7). Additionally, it turns out that this time for both images layer progression is superior (i.e. lower quality) to resolution progression in terms of PSNR.

Again, the numerical values have to be related to visual inspection. Fig. 11a shows a reconstruction of the selectively compressed Angio2 image, where the first 1% of the packets in resolution progressive mode have been encrypted and the reconstruction is done using the error concealment technique. In this case, this leads to a PSNR value of 10.51 dB, whereas the directly reconstructed image has a value of 7.45 dB (see Fig. 9b). The text in the right corner is clearly readable and even the structure of the blood vessels is exhibited.

When increasing the percentage of encrypted packet data steadily, we finally result in 20% of the packet data encrypted where neither useful visual nor textual information remains in the image (see Fig. 11b). This result is confirmed also with other images and can be used as a rule of thumb for a secure use of selective encryption of the JPEG2000 bitstream.

4. Conclusion

Computationally efficient techniques for confidential storage and transmission of medical image data have been discussed. We propose two types of partial encryption techniques based on AES: The first encrypts a subset of bitplanes of plain image data whereas the second encrypts parts of the JPEG2000 bitstream. For both techniques, the percentage of data subjected to encryption while maintaining high confidentiality is significantly reduced as compared to full encryption. However, in selective bitplane encryption up to 50% need to be encrypted whereas in the case of JPEG2000 bitstream encryption the encryption of 20% data already delivers a satisfying result. The difference in terms of percentage becomes even bigger in terms of absolute values since bit-

plane data is not compressed at all. This large difference is due to the fact that important visual features are concentrated at the begin of the embedded JPEG2000 bitstream and may therefore be protected effectively whereas the visual features are spread across bitplanes to a much larger extent.

Summary

We discuss computationally efficient techniques for confidential storage and transmission of medical image data. In contrast to providing confidentiality at the infrastructure level (e.g. when using IPSec), we target confidentiality at the application level in order to exploit application specific properties to create more efficient schemes. After discussing reasonable application scenarios for selective or partial encryption, two types of techniques of this type based on AES are proposed. The first encrypts a subset of bitplanes of plain image data whereas the second encrypts parts of the JPEG2000 bitstream. We discuss simple ciphertext-only attacks against both encryption schemes which reveal that encrypting only a fraction of the original data is sufficient to provide high confidentiality. When using selective bitplane encryption up to 50% need to be encrypted whereas in the case of JPEG2000 bitstream encryption the protection of 20% data already delivers a satisfying result.

References

[1] S. Wong, L. Zaremba, D. Gooden, H.K. Huang, Radiologic image compression—a review, Proc. IEEE 83 (2) (1995) 194–219.
[2] P.C. Cosman, R.M. Gray, R.A. Olshen, Evaluating quality of compressed medical images: SNR, subjective rating, and diagnostic accuracy, Proc. IEEE 82 (6) (1994) 919–932.
[3] A. Bruckmann, A. Uhl, Selective medical image compression techniques for telemedical and archiving applications, Comput. Biol. Med. 30 (3) (2000) 153–169.
[4] I.L. Bajec, P. Trunk, N. Zimic, D. Oseli, Virtual coronary cineangiography, Comput. Biol. Med. (2003), to appear.
[5] P. Trunk, B. Gersak, R. Trobec, Topical cardiac cooling—computer simulation of myocardial temperature changes, Comput. Biol. Med. (2003), to appear.
[6] J. Daemen, V. Rijmen, The Design of Rijndael: AES—the Advanced Encryption Standard, Springer, Berlin, 2002.
[7] B. Schneier, Applied Cryptography, 2nd Edition: Protocols, Algorithms and Source Code in C, Wiley Publishers, New York, 1996.
[8] Lintian Qiao, Klara Nahrstedt, Comparison of MPEG encryption algorithms, Int. J. Comput. Graphics (Special Issue Data Security Image Commun. Networks) 22 (3) (1998) 437–444.
[9] I. Agi, L. Gong, An empirical study of secure MPEG video transmissions, in: ISOC Symposium on Network and Distributed Systems Security, San Diego, CA, 1996, pp. 137–144.
[10] A.M. Alattar, G.I. Al-Regib, S.A. Al-Semari, Improved selective encryption techniques for secure transmission of MPEG video bit-streams, in: Proceedings of the 1999 IEEE International Conference on Image Processing (ICIP'99), IEEE Signal Processing Society, 1999.
[11] Thomas Kunkelmann, Applying encryption to video communication, in: Proceedings of the Multimedia and Security Workshop at ACM Multimedia '98, Bristol, England, September 1998, pp. 41–47.
[12] T. Maples, G. Spanos, Performance study of a selective encryption scheme for the security of networked real-time video, in: Proceedings of the Fourth International Conference on Computer Communications and Networks (ICCCN'95), Las Vegas, NV, 1995.

[13] P.A. Schneck, K. Schwan, Authenticast: an adaptive protocol for high-performance, secure network applications, Technical Report, Georgia Institute of Technology, Atlanta, GA, USA, 1997.

[14] C. Shi, B. Bhargava, A fast MPEG video encryption algorithm, in: Proceedings of the ACM Multimedia 1998, Boston, USA, 1998, pp. 81–88.

[15] L. Tang, Methods for encrypting and decrypting MPEG video data efficiently, in: Proceedings of the ACM Multimedia 1996, Boston, USA, November 1996, pp. 219–229.

[16] Tsung-Li Wu, S. Felix Wu, Selective encryption and watermarking of MPEG video (extended abstract), in: H.R. Arabnia (Ed.), Proceedings of the International Conference on Image Science, Systems, and Technology, CISST '97, Las Vegas, USA, February 1997.

[17] Wenjun Zeng, Shawmin Lei, Efficient frequency domain video scrambling for content access control, in: Proceedings of ACM Multimedia, 1999, Orlando, FL, USA, November 1999, pp. 285–293.

[18] Raphaël Grosbois, Pierre Gerbelot, Touradj Ebrahimi, Authentication and access control in the JPEG 2000 compressed domain, in: A.G. Tescher (Ed.), Applications of Digital Image Processing XXIV, Proceedings of SPIE, Vol. 4472, San Diego, CA, USA, July 2001.

[19] A. Pommer, A. Uhl, Wavelet packet methods for multimedia compression and encryption, in: Proceedings of the 2001 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Victoria, Canada, August 2001, IEEE Signal Processing Society, 2001, pp. 1–4.

[20] A. Pommer, A. Uhl, Selective encryption of wavelet packet subband structures for obscured transmission of visual data, in: Proceedings of the Third IEEE Benelux Signal Processing Symposium (SPS 2002), Leuven, Belgium, March 2002, IEEE Benelux Signal Processing Chapter, 2002, pp. 25–28.

[21] T. Uehara, R. Safavi-Naini, P. Ogunbona, Securing wavelet compression with random permutations, in: Proceedings of the 2000 IEEE Pacific Rim Conference on Multimedia, Sydney, December 2000, IEEE Signal Processing Society, pp. 332–335.

[22] H. Cheng, X. Li, On the application of image decomposition to image compression and encryption, in: P. Horster (Ed.), Communications and Multimedia Security II, IFIP TC6/TC11 Second Joint Working Conference on Communications and Multimedia Security, CMS '96, Essen, Germany, September 1996, Chapman & Hall, London, pp. 116–127.

[23] H. Cheng, X. Li, Partial encryption of compressed images and videos, IEEE Trans. Signal Process. 48 (8) (2000) 2439–2451.

[24] M. Podesser, H.-P. Schmidt, A. Uhl, Selective bitplane encryption for secure transmission of image data in mobile environments, CD-ROM in: Proceedings of the Fifth IEEE Nordic Signal Processing Symposium (NORSIG 2002), Tromso-Trondheim, Norway, October 2002, file cr1037.pdf, IEEE Norway Section.

[25] Ali Saman Tosun, Wu Chi Feng, On error preserving encryption algorithms for wireless video transmission, in: ACM Multimedia 2001, Ottawa, Canada, October 2001, pp. 302–307.

[26] J. Wen, M. Severa, W. Zeng, M. Luttrell, W. Jin, A format-compliant configurable encryption framework for access control of multimedia, in: Proceedings of the IEEE Workshop on Multimedia Signal Processing, MMSP '01, Cannes, France, October 2001, pp. 435–440.

[27] C.J. Skrepth, A. Uhl, Selective encryption of visual data: classification of application scenarios and comparison of techniques for lossless environments, in: B. Jerman-Blazic, T. Klobucar (Eds.), Advanced Communications and Multimedia Security, IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security, CMS '02, Portoroz, Slovenia, September 2002, Kluwer Academic Publishing, Dordrecht, 2002, pp. 213–226.

[28] M. Unser, A. Aldroubi, A review of wavelets in biomedical applications, Proc. IEEE 84 (4) (1996) 626–638.

[29] A. Aldroubi, M. Unser (Eds.), Wavelets in Medicine and Biology, CRC, Boca Raton, FL, USA, 1996.

[30] P.N. Topiwala (Ed.), Wavelet Image and Video Compression, Kluwer Academic Publishers Group, Boston, 1998.

[31] D. Taubman, M.W. Marcellin, JPEG2000—Image Compression Fundamentals, Standards and Practice, Kluwer Academic Publishers, Dordrecht, 2002.

[32] Iraj Sodagar, H.J. Lee, P. Hatrack, Ya-Qin Zhang, Scalable wavelet coding for synthetic/natural hybrid coding, IEEE Trans. Circuits Systems Video Technol. 9 (2) (1999) 244–254.

[33] D. Taubman, High performance scalable image compression with EBCOT, IEEE Trans. Image Process. 9 (7) (2000) 1158–1170.

Roland Norcen and Andreas Pommer received their B.S. and M.S. (Computer Science) degrees from the Salzburg University where they are currently enrolled in a Ph.D. program. The submitted work is related to the topics of their respective Ph.D. theses.

Martina Podesser and Hans-Peter Schmidt are both graduate students at the Telecommunication & Network Engineering School at Carinthia Tech Institute in Klagenfurt. Their contribution to this work has been done in the context of the system security lab which is part of the curriculum on communication networks.

Andreas Uhl received the B.S. and M.S. degrees (both in Mathematics) from Salzburg University and he completed his Ph.D. on Applied Mathematics at the same University. He is currently associate professor with tenure in computer science affiliated with the Department of Scientific Computing and with the Research Institute for Software Technology. He is also part-time lecturer at the Carinthia Tech Institute. His research interests include multimedia signal processing (with emphasis on compression and security issues), parallel and distributed processing, and numbertheoretical methods in numerics.