Confidential Transmission of Lossless Visual Data: Experimental Modelling and Optimization

Bubi G. Flepp-Stars[1⋆], Herbert Stögner[1], and Andreas Uhl[1,2]

[1] Carinthia Tech Institute, School of Telematics & Network Engineering
Primoschgasse 8, A-9020 Klagenfurt, Austria
[2] Salzburg University, Department of Scientific Computing
Jakob-Haringerstr.2, A-5020 Salzburg, Austria

Abstract. We discuss confidential transmission of visual data in lossless format. Based on exemplary experimental data, we model the costs of the three main steps of such a scheme, i.e. compression, encryption, and transmission. Subsequently, we derive a cost optimal strategy for employment of confidential visual data transmission. Our approach may be applied to any environment provided that the costs of the single main steps in the target environment are known.

1 Introduction

The organization of large amounts of visual content in database infrastructures which are distributed worldwide shows very clearly that there is urgent need to provide and protect the confidentiality of sensitive visual data (e.g. patient related medical image data) when transmitted over networks of any kind.

The storage and transmission of visual image data in lossless formats differs significantly from storage and transmission of common visual data for multimedia applications. It is constrained by the fact that in lossless environments the amount of compression is typically limited to a factor of about 2-3 in contrast to factors of 100 or more achievable in lossy schemes [4].

There exist several reasons why a lossy representation may not be acceptable or a lossless one may be preferable:

- Due to requirements of the application a loss of image data is not acceptable (e.g., in medical applications because of reasons related to legal aspects and diagnosis accuracy [17], in geographical information systems (GIS) due to the required integrity of the data).
- Due to the low processing power or limited energy resources of the involved hardware, compression and decompression of visual data may not be possible or desired at all (e.g. mobile and wireless clients in the context of pervasive/ubiquitous computing).

⋆ This artificial name represents the following group of students working on this project in the framework of the multimedia 1 laboratory (winterterm 2003/2004): G. Biebl, J. Burger, T. Freidl, J. Gruber, P. Leidner, R. Pfarrhofer, J. Podritschig, S. Rebernig, T. Salbrechter, and G. Stanossek

A. Lioy and D. Mazzocchi (Eds.): CMS 2003, LNCS 2828, pp. 252–263, 2003.

– Due to the high bandwidth available at the communication channel lossy compression is not necessary and lossless formats are therefore preferred.

A possible solution to the first issue 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 [2]. However, legal questions and problems related to the usability of such a scheme remain unsolved. Therefore, we restrict the discussion to lossless data formats.

In the context of applications involving visual data transmission, we may distinguish whether the image data is given as plain image data (e.g. after being captured by a digitizer or CCD array) or in form of a bitstream resulting from prior compression. The first application scenario has been denoted as "on-line" (e.g. video conferencing, surveillance applications) and the latter "off-line" (since the respective applications like video on demand or photo CD-ROM are purely retrieval based) [9,14]. Note that the involvement of compression technology might be not mandatory in the on-line scenario, especially when focusing onto lossless techniques due to the limited compression gain in this case.

In this work we focus on computationally efficient schemes to provide confidentiality for the transmission of visual data in a lossless on-line scenario. In particular, we seek to optimize the interplay of the three main steps of such a scheme, i.e. compression, encryption, and transmission in the sense of minimal computational effort and energy consumption. Based on exemplary experimental data, we model the costs of the three involved processing steps and subsequently derive a cost optimal strategy for employment of confidential visual data transmission in the target environment. Specifically, we aim at the question whether the compression stage is required in any case to result in an overall cost optimal scheme or not.

Additionally, we consider "selective encryption" (SE) for trading off computational complexity for security, where application specific data structures are exploited to create more efficient encryption systems (see e.g. SE of MPEG video streams [1,6,11,13,15,18], of wavelet-based encoded imagery [3,7,10,18], and of quadtree decomposed images [3]). The basic idea of SE is to protect (i.e. encrypt) the visually most important parts of an image or video representation only, relying on a secure but slow "classical" cipher.

Section 2 provides an introduction to principles of confidential transmission of visual data. In section 3 we present the building blocks of our target environment and model the respective computational costs based on experimental data. Cost optimal employment of several different configurations of confidential visual data transmission using full encryption and SE is derived in section 4. In the conclusion we summarize the main results and give an outlook to further work in this direction.

2 Principles of Confidential 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, usually symmetric encryption is used in practical applications. On the other hand, key management is a difficult issue in symmetric systems. As done in most current applications with demand for confidentiality, public key techniques like RSA are used for key exchange or digital signature generation only (such schemes are usually denoted as "hybrid").

The Advanced Encryption Standard AES [5] 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. The RSA algorithm [12] is the most popular public key algorithm nowadays and operates with key sizes of 512 bit upwards. For achieving confidentiality, the data is encrypted using the public key of the receiver who may decrypt the message using his private key. Vice versa, for generating a digital signature, the data is signed (i.e. encrypted) using the private key of the signer. The signature may be verified with the corresponding public key. We use AES and RSA as the basic cryptographic building blocks in section 2.

There are two ways to provide confidentiality to a 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 themselves 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 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. For example, all approaches involving selective encryption are classified into the second category since SE takes advantage of the redundancy in visual data and therefore takes place at the application level.

3 Basic Building Blocks: Compression, Encryption, and Transmission

In any (storage or) transmission application no matter if lossy or lossless, compression 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. Therefore, the processing chain has a fixed order (compression – encryption – transmission). In the following subsections, we introduce the basic technology and model the costs in terms of computational demand of the stages in the processing chain in the target environment. The hardware platform used is a 996 MHz Intel Pentium III with 128 MB RAM, the Network is 100 MBit/s Ethernet.

3.1 Lossless Compression

In order to model a compression scheme which trades off compression efficiency for computational demand, we use the JBIG reference implementation in a selective mode where a different amount of bitplanes of 8 bpp grayscale images is compressed. So our scheme ranges from applying no compression at all to compressing a certain number of bitplanes using JBIG (starting from the MSB bitplane since the achievable compression ratio is the highest here). Finally, instead of applying JBIG to all bitplanes, JPEG 2000 (the Jasper C implementation) in lossless mode is used since the compression results are better as compared to full JBIG coding.

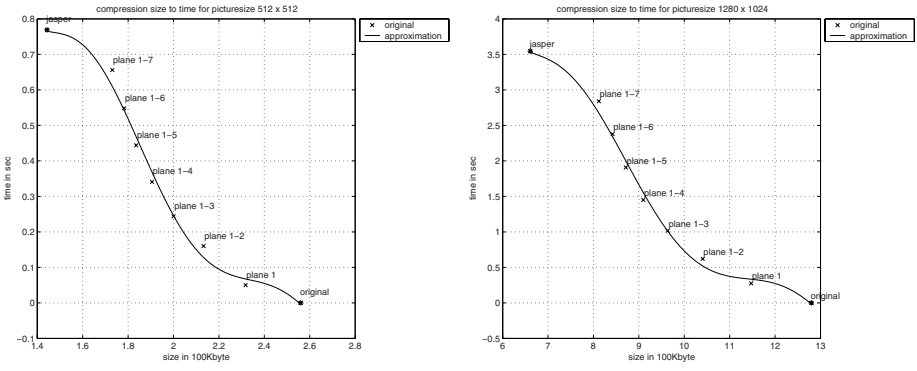
A set of 20 testimages in two sizes is subjected to all compression settings and the obtained file sizes and compression timings are averaged for the 512×512 and 1280×1024 pixels images, respectively. The resulting measurement points as depicted in Fig. 1 clearly show the decreasing compression time for increasing data size.

In order to obtain a closed expression to be handled easily in an analytical way for the optimization, we approximately interpolate the measurement points by a 6th-order polynomial and result in the following formulas (also visualized in Fig. 1) for the compression behaviour of our test system (where x is the resulting data size in 100 KByte after compression and t is the compression time in seconds):

$$t = 7.32x^6 - 90.89x^5 + 463.02x^4 - 1237.72x^3 + 1829.28x^2 - 1417.22x + 450.73 \quad (1)$$

$$t = -0.07x^5 + 1.73x^4 - 21.99x^3 + 153.94x^2 - 562.09x + 841.21 \quad (2)$$

Equations (1) and (2) correspond to Figs. 1.a and 1.b, respectively.



(a) 512x512 pixels images (b) 1280x1024 pixels images

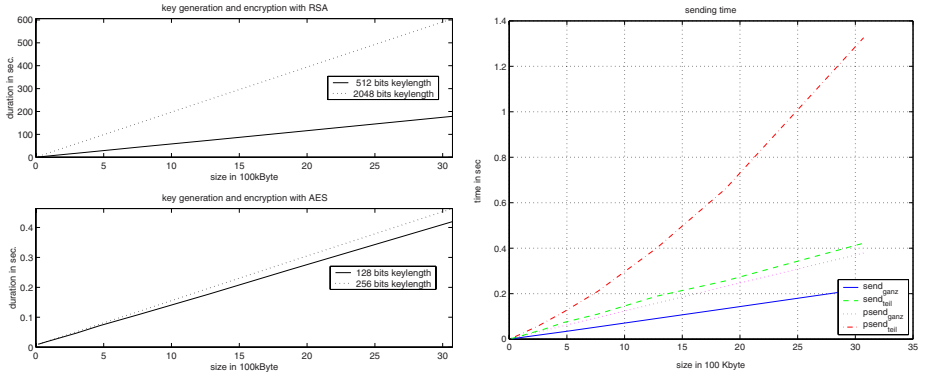
Fig. 1. Trandeoff between compression timings and the resulting data amount after compression.

3.2 Encryption and Transmission

In order to model encryption behaviour, we use the C++ RSA implementation given in [16] and the C++ AES implementation available at the web (`http://fp.gladman.plus.com/cryptography_technology/rijndael/`). Equations (3) to (6) relate the amount of data encrypted in 100 kByte to the processing time (given in seconds), the corresponding curves are visualized in Fig. 2.a. In contrast to the compression case, we notice a purely linear behaviour. Note that RSA is employed for reasons of obtaining a rich variety in the overall behaviour of the processing chain. In practice, one would hardly use a public-key system to encrypt visual data. Of course, the time demand of RSA is several orders of magnitude higher as compared to AES. Performance differences among encryption schemes with the exhibited magnitude could result from applying hardware or software based approaches in real-life systems.

$$t = 5.81x \quad \text{(RSA 512 bit)} \tag{3}$$
$$t = 19.72x \quad \text{(RSA 2048 bit)} \tag{4}$$
$$t = 0.01x \quad \text{(AES 128 bit)} \tag{5}$$
$$t = 0.02x \quad \text{(AES 256 bit)} \tag{6}$$

Finally, the transmission stage is modeled by using the message passing library PVM for sending data between two computers over the Ethernet connection. We use four different modes. The routine pvm_send sends a message stored in the active send buffer to the PVM process identified by tid. On the other hand, the routine pvm_psend takes a pointer to a buffer buf, its length



(a) Encryption (b) Transmission

Fig. 2. Processing timings for varying data amount.

len, and its data type datatype and sends this data directly to the PVM task identified by tid. Data is sent as a whole block in mode "ganz" whereas it is sent in pieces of 1 KByte in mode "teil". Again, the data size is varied and the time required to transmit the data is measured and fitted by a polynomial (see Fig. 2.b and equations (7) - (10). We notice small differences among the transmission schemes with pvm_psend in "teil" mode being the most expensive one. All schemes are dominated by the linear term, higher order terms vanish with the precision considered.

$$t = 0.01x \quad \text{pvm\_send, teil} \tag{7}$$
$$t = 0.02x \quad \text{pvm\_psend, teil} \tag{8}$$
$$t = 0.007x \quad \text{pvm\_send, ganz} \tag{9}$$
$$t = 0.01x \quad \text{pvm\_psend, ganz} \tag{10}$$

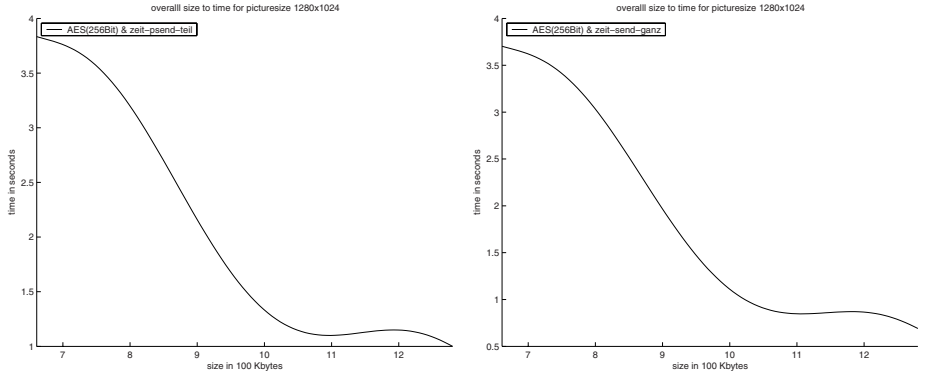
Summarizing we notice that AES encryption and transmission operate on a similar level of time demand, whereas RSA is much more expensive. As expected, both processing stages exhibit linear behaviour.

4 Cost Optimal Configuration of Confidential Visual Data Transmission

The processing chain for confidential transmission of visual data as defined in the last section (compression – encryption – transmission) may have a fixed order, but does not necessarily involve all stages in full completeness in any case. For example, in SE only a subset of the data is subjected to encryption.

However, the entire system might have to maintain a certain level of security which puts constraints on the lower bound of data subjected to encryption. Similarly, compression must not necessarily be applied to the full amount of data or might be omitted at all for complexity reasons. Again, limited transmission bandwidth may impose a constraint on the possible datarate and therefore enforce the employment of compression to a certain extent. The aim of this section is to identify a cost optimal way (in terms of processing time) to operate our processing chain. Concerning constraints imposed by the target environment, we assume the channel to be of infinite bandwidth and SE as being secure enough for the target application.

The first configuration considered is to process 1280×1024 images and to use AES as cipher. In Fig. 3 we clearly see that the overall shape of the resulting time curves closely matches the curve modeling compression behaviour (see Fig. 1.b). In particular, the curves are almost monotonically decreasing and attain their minima at the right edge of the x-axis, corresponding to a data size of about 1290 KB. When matching this result with Fig. 1.b, we identify the optimal operation mode as to perform no compression at all. This is a surprising result at first sight of course, however, when considering the low processing cost of encryption and transmission as compared to compression, it is obvious.



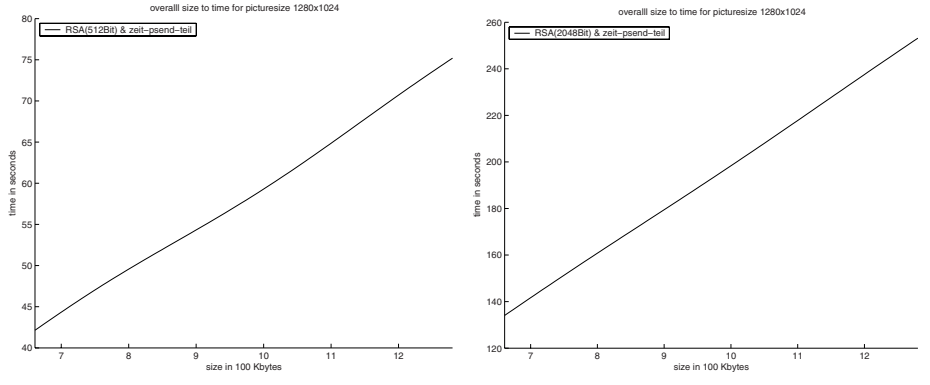
(a) AES(256) with psend_teil (b) AES(256) with send_ganz

Fig. 3. Overall timings for varying data amount processing 1280x1024 pixels images using AES.

The only difference between Figs. 3.a and 3.b is the employment of the most and least expensive transmission modes. As expected, this results in almost no difference in the final overall time curves. The same is true whether 128 or 256 bit AES is used (figures are not shown). As a consequence, the formulas describing

the overall behaviour are identical to Equation (2) with the linear terms replaced by $-562.05x$ and $-562.07x$, respectively.

The second configuration investigated is to process 1280×1024 images and to use RSA as cipher. Fig. 4 shows entirely different shapes as seen before. Almost linearly increasing curves describe the time behaviour of the system, attaining their respective minima at the left edge of the x-axis, corresponding to a data size of 660 KB. The fact that these curves are monotonically increasing may be easily verified by showing the first derivative of the corresponding formulas to be larger as zero over the entire range considered. Relating this finding to Fig. 1.b, the optimal operation mode is to perform maximal compression, in this case lossless JPEG 2000.



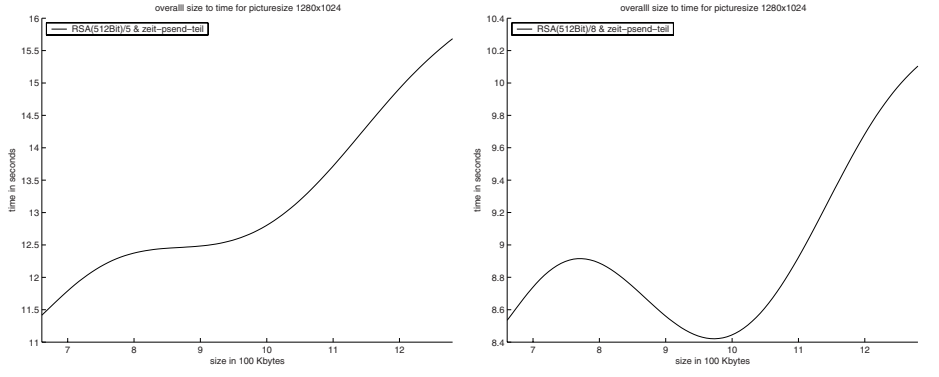
(a) RSA(512) with psend_teil (b) RSA(2048) with psend_teil

Fig. 4. Overall timings for varying data amount processing 1280x1024 pixels images using RSA.

The employment of RSA with differently sized keys (i.e. 512 bit for Fig. 4.a and 2048 bit for Fig. 4.b) does change neither the shape of the curve nor the position of the minimum, the curve is simply upshifted a certain amount (which dramatically shows the dominance of encryption in this configuration). The formulas describing the overall behaviour are again identical to Equation (2) with the linear terms replaced by $-556.26x$ and $-542.35x$, respectively.

Having identified the optimal operation modes of our test system with full encryption, we now focus onto selective encryption (SE). In this context, SE only encrypts a subset of the bitplanes of the binary representation of the visual data. For results concerning the security of this approach see [8,14]. Besides the interesting properties of this technique itself, we are interested whether there exist optimal operation modes where a partial compression is required (instead of no compression with AES and full compression with RSA). Selective encryption

is modeled by reducing the processing time of encryption to the corresponding amount, i.e. by modifying Equations (3) - (6) accordingly. Fig. 5 shows the shapes of the resulting overall curves when processing 1280×1024 pixels images using selective encryption with RSA (512 bit key).



(a) 20 % encrypted with psend_teil (b) 12.5 % encrypted with psend_teil

Fig. 5. Overall timings for varying data amount processing 1280x1024 pixels images using 512 bit RSA with selective encryption.

Whereas encrypting 20 % of the visual data still results in a monotonically increasing curve (Fig. 5.a), the reduction of encryption to 1/8 (i.e. 12.5 %) of the data results in a curve showing a local minimum close to the middle of the range of the x-axis (Fig. 5.b). Equations (11) and (12) show the corresponding formulas for the overall time behaviour.

$$t1(x) = 0.71x^5 + 1.73x^4 - 21.99x^3 + 153.94x^2 + \\ -560.91x + 841.21 \tag{11}$$

$$t2(x) = 0.71x^5 + 1.73x^4 - 21.99x^3 + 153.94x^2 + \\ -561.34x + 841.21 \tag{12}$$

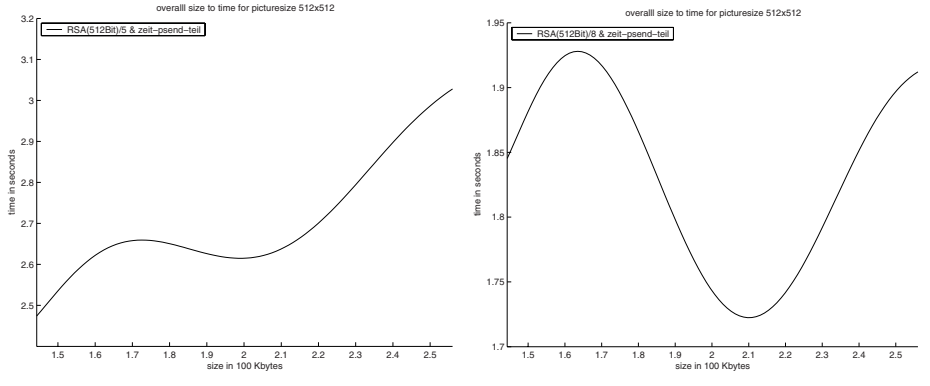
In order to analytically determine the point of optimal configuration for the SE restricted to 12.5 % encryption, we need to determine the local minimum of Equation (12). In order to do that we set the first derivative to zero

$$t2'(x) = 3.55x^4 + 6.91x^3 - 65.96x^2 + 307.88x - 561.41 = 0 \tag{13}$$

and obtain two roots in the area of interest $[6.6, 13]$ (compare Fig. 1.b): $x = 7.72$ and $x = 9.71$. Obviously, the local minimum we are looking for is

attained at $x = 9.71$ (which can be computing by verifying $t2'(9.71) < 0$ and is also shown in Fig. 5.b). Relating this result to Fig. 1.b we may deduce that the optimal configuration of the entire system compressing 1280×1024 pixels images and applying selective encryption with 512 bit RSA to 12.5 % of the data is to compress 3 out of 8 bitplanes with JBIG.

Finally, we cover the case of processing 512×512 pixels images with 512 bit RSA. Fig. 6 shows similar shapes of the resulting overall time curves compared to the case of larger images before, equations (14) and (15) show the corresponding formulas. Note that only relatively small differences in the linear term cause the significant different shapes as compared to the "compression" Equation (1). Again we focus on the interesting case of encrypting 12.5 % of the data (Fig. 6.b).



(a) 20 % encrypted with psend_teil (b) 12.5 % encrypted with psend_teil

Fig. 6. Overall timings for varying data amount processing 512x512 pixels images using 512 bit RSA with selective encryption.

$$t3(x) = 7.32x^6 - 90.89x^5 + 463.02x^4 - 1237.72x^3 + 1829.28x^2 + \\ -1416.04x + 450.73 \quad (14)$$

$$t4(x) = 7.32x^6 - 90.89x^5 + 463.02x^4 - 1237.72x^3 + 1829.28x^2 + \\ -1416.47x + 450.73 \quad (15)$$

Similarly, in order to determine the point of optimal configuration, we need to determine the local minimum of equation (15). In order to do that we set the first derivative to zero

$$t4'(x) = 43.93x^5 - 454.43x^4 + 1852.1x^3 - 3713.16x^2 + 3658.56x - 1416.47 = 0 \quad (16)$$

and obtain two roots in the area of interest $[1.4, 2.6]$ (compare Fig. 1.a): $x = 1.64$ and $x = 2.10$. Obviously, the local minimum we are looking for is attained at $x = 2.10$ (which can be computing by verifying $t4'(2.10) < 0$ and is also shown in Fig. 6.b). As before, we identify the optimal operation mode of the entire system to apply selective compression to 2 bitplanes out of 8 (to be seen from Fig. 1.a).

5 Conclusion

In the context of confidential transmission of visual data in lossless format, we have modeled the costs of the three main steps of such a scheme, i.e. compression, encryption, and transmission. Based on the behaviour of our exemplary system, we have shown that depending on the type of encryption involved, the optimal configuration of the entire system may be to operate without compression, with full compression, or even with partial compression. In future work we will additionally include constraints coming from the target environment (e.g. limited channel bandwidth, certain level of security in SE) into our optimization. Additionally we will model the dependency between selective compression and selective encryption more clearly.

Acknowledgements. This work has been partially supported by the Austrian Science Fund FWF, project 15170.

References

[1] A. M. Alattar, G. I. Al-Regib, and 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)*, volume 4, pages 256–260, Kobe, Japan, October 1999. IEEE Signal Processing Society.

[2] A. Bruckmann and A. Uhl. Selective medical image compression techniques for telemedical and archiving applications. *Computers in Biology and Medicine*, 30(3):153 – 169, 2000.

[3] H. Cheng and X. Li. Partial encryption of compressed images and videos. *IEEE Transactions on Signal Processing*, 48(8):2439–2451, 2000.

[4] P.C. Cosman, R.M. Gray, and R.A. Olshen. Evaluating quality of compressed medical images: SNR, subjective rating, and diagnostic accuracy. *Proceedings of the IEEE*, 82(6):919–932, 1994.

[5] J. Daemen and V. Rijmen. *The Design of Rijndael: AES — The Advanced Encryption Standard*. Springer Verlag, 2002.

[6] Thomas Kunkelmann. Applying encryption to video communication. In *Proceedings of the Multimedia and Security Workshop at ACM Multimedia '98*, pages 41–47, Bristol, England, September 1998.

[7] R. Norcen, M. Podesser, A. Pommer, H.-P. Schmidt, and A. Uhl. Confidential storage and transmission of medical image data. *Computers in Biology and Medicine*, 33(3):277–292, 2003.

[8] M. Podesser, H.-P. Schmidt, and A. Uhl. Selective bitplane encryption for secure transmission of image data in mobile environments. In *CD-ROM Proceedings of the 5th IEEE Nordic Signal Processing Symposium (NORSIG 2002)*, Tromso-Trondheim, Norway, October 2002. IEEE Norway Section. file cr1037.pdf.

[9] A. Pommer and A. Uhl. Application scenarios for selective encryption of visual data. In J. Dittmann, J. Fridrich, and P. Wohlmacher, editors, *Multimedia and Security Workshop, ACM Multimedia*, pages 71–74, Juan-les-Pins, France, December 2002.

[10] A. Pommer and A. Uhl. Selective encryption of wavelet packet subband structures for secure transmission of visual data. In J. Dittmann, J. Fridrich, and P. Wohlmacher, editors, *Multimedia and Security Workshop, ACM Multimedia*, pages 67–70, Juan-les-Pins, France, December 2002.

[11] Lintian Qiao and Klara Nahrstedt. Comparison of MPEG encryption algorithms. *International Journal on Computers and Graphics (Special Issue on Data Security in Image Communication and Networks)*, 22(3):437–444, 1998.

[12] B. Schneier. *Applied cryptography (2nd edition): protocols, algorithms and source code in C*. Wiley Publishers, 1996.

[13] C. Shi and B. Bhargava. A fast MPEG video encryption algorithm. In *Proceedings of the Sixth ACM International Multimedia Conference*, pages 81–88, Bristol, UK, September 1998.

[14] Champskud J. Skrepth and Andreas Uhl. Selective encryption of visual data: Classification of application scenarios and comparison of techniques for lossless environments. In B. Jerman-Blazic and T. Klobucar, editors, *Advanced Communications and Multimedia Security, IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security, CMS '02*, pages 213 – 226, Portoroz, Slovenia, September 2002. Kluver Academic Publishing.

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

[16] M. Welschenbach. *Kryptographie in C und C++. Zahlentheoretische Grundlagen, Computer-Arithmetik mit großen Zahlen, kryptographische Tools*. Springer Verlag, 1998.

[17] S. Wong, L. Zaremba, D. Gooden, and H.K. Huang. Radiologic image compression – a review. *Proceedings of the IEEE*, 83(2):194–219, 1995.

[18] Wenjun Zeng and Shawmin Lei. Efficient frequency domain video scrambling for content access control. In *Proceedings of the seventh ACM International Multimedia Conference 1999*, pages 285–293, Orlando, FL, USA, November 1999.