Low-complexity Wavelet Transforms for Multimedia Wireless Sensor Networks

Stephan Rein, Technical University Berlin, Germany, rein@tkn.tu-berlin.de
Martin Reisslein, Arizona State University, USA, reisslein@asu.edu

1. Introduction

Current image processing techniques generally exceed the computational and memory resources of the typically very limited, microcontroller based sensor nodes. The TelosB mote platform [1] for instance, employs a 16 Bit microcontroller with a total RAM memory of 10 kBytes. Floating-point operations are not supported by the hardware. It is possible to extend the limited hardware with DSP units to realize sensor network multimedia applications [2]. We have developed an alternative low-cost approach that implements low-complexity image processing algorithms through a software update; thus, enabling multimedia applications with low-cost sensors. Wavelet algorithms are promising starting-points for such considerations, as they de-correlate the data and give state-of-the-art image compression performance. Our algorithm can be supported by flash memory - a standard component of current sensor platforms [3].

A line-based version of the wavelet transform has been developed in [3], which provides substantial reduction of the memory requirements for the traditional wavelet transform approach. An improved computation methodology readily applicable for limited memory systems is given in [4]. It can however not yet fulfill the low-memory constraints of low-cost sensor nodes, as still 26 kiloBytes are required for a six-level transform of a 512x512 image.

The tutorial in [5] surveys the underlying techniques for the needed wavelet transform algorithm to perform image compression on a low-cost sensor node. A review of the general image wavelet transform, a low memory scheme for its computation -- the so-called fractional wavelet filter, and some performance results of this filter are given in the next sections.

2. Image wavelet transform

The general image wavelet transform computes *subbands* of an input image. These subbands are computed by filtering the image row-by-row, resulting in the so-called L and H matrices, which both have half the dimension of the original image (*horizontal* transform). Then, these matrices are filtered column-by-column, resulting in the four subbands LL, HL, LH, and HH of the first wavelet level (*vertical* transform). For computation of higher wavelet levels, the LL subband is taken as a starting point to compute the four subbands of the second level. Each of these second level subbands has half the dimension of the first level subbands, see Figure 1 for an example. The traditional computational order for such a transform is to first apply the horizontal transform on the complete image and then the vertical transform, thus keeping complete versions of the image in memory.

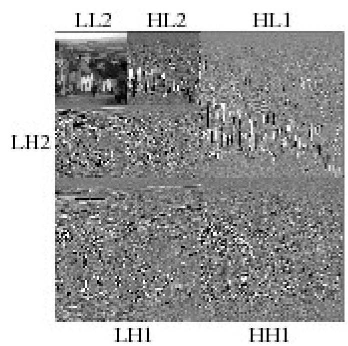

Figure 1: Example of a two-level image wavelet transform.

2. The Fractional Wavelet Filter

The fractional wavelet filter is a computational scheme that computes the image wavelet transform on a sensor node with very small RAM memory with support of an attached flash memory (MMC card). For instance, the transform of a grayscale 256x256 image requires less than 1.5 kBytes of RAM. Because the data on the MMC card can only be accessed in blocks of 512 Bytes, a sample-by-sample access as easily executed with RAM memory on personal computers is not feasible. The fractional filter takes this restriction into account by reading the image samples line-by-line from the MMC card. For the first transform level, the algorithm reads the original image samples while it writes the subbands to a different destination on the MMC card. For the following levels the LL subband gives the input data for the next transform. The filter has two options for the computation of the transform, the classical convolution approach and the more modern so-called *lifting scheme,* which allows the filtering to be computed in-place. For each transform level a new destination matrix is allocated on the MMC card. Since the MMC card has abundant memory this approach does itnot affect the sensor's resources. This approach also allows reconstruction of the image from any transform level.

The fractional filter combines vertical and horizontal transform such that for each input line final results for the horizontal transform are computed, while it computes for the vertical transform only *fractions* of the final results. The input lines are scanned several times to update the fractions until the update results into the final wavelet transform values to be written on the MMC card.

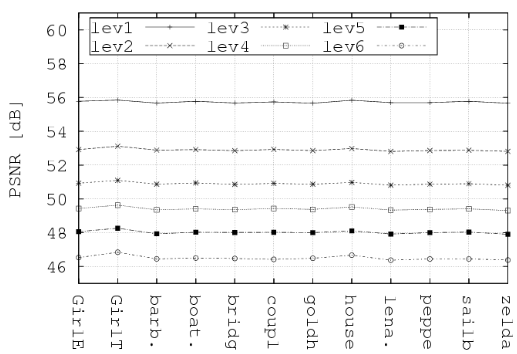

Figure 2: PSNR values for reconstructed pictures using the fixed-point version of the fractional wavelet filter. The transform is not lossless, however, the loss does not affect picture quality when the filter is combined with high compression coding schemes.

The floating-point version of the fractional wavelet filter computes the wavelet transform with high precision, as it uses 32 bit floating-point variables for the wavelet and filter coefficients as well as for the intermediate operations. Thus, the images can be reconstructed essentially without loss of information. Many low-cost micro-controllers do however not provide hardware support for floating-point operations. If floating-point operations are coded, the compiler translates them to integer operations, which often results in prohibitively long computing times. Converting an algorithm from floating- to fixed-point arithmetic typically results in substantial time and energy savings at the expense of lower precision and thorough number range analysis. Thus, using fixed-point arithmetic for the fractional wavelet filter can help to significantly reduce the computational requirements and to reduce the RAM memory needed for representation of the destination subbands. We refer to [5] for tutorial background on using fixed-point arithmetic for wavelet transforms. The PSNR values for a set of example pictures computing the transform with the fractional filter up to levels 1, 2, … , 6 are given in Figure 2.

Figure 1 shows the compression performance of the fractional filter when combined with a suitable image coder [7] compared to JPEG, JPEG2000, and the set partitioning in hierarchical trees (Spiht) algorithm. For high data compression rates, the fractional filter allows for similar compression as JPEG2000, which is considered as state-of-the-art in image compression. A detailed performance evaluation of the fractional filter is given in [6].

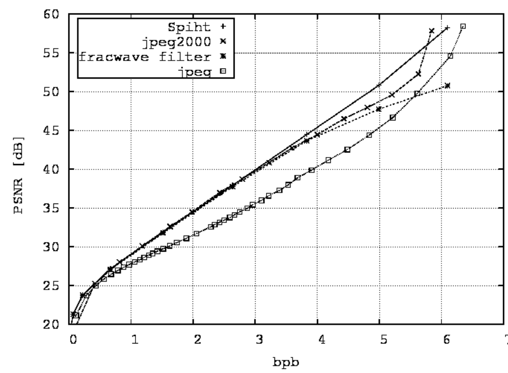

Figure 1: Compression performance (PSNR vs. Bits per Byte (bpb)) for the introduced fractional wavelet filter when combined with a low-memory image coder.

4. Conclusions

This E-Letter briefly reviews the techniques of the low-complexity wavelet transforms for application in low-cost wireless sensor networks. In the past, image wavelet transforms have been considered as being not applicable in such microcontroller-operated networks. However, recent algorithms, including the fractional wavelet filter, leverage the application of wavelet transform in such networks by reducing the memory requirements for the transform, thus making state-of-the-art image compression feasible through only a software update.

References

[1] J. Polastre, R. Szewcyk, and D. Culler, "Telos: enabling ultra-low power wireless research", in *Proc. Int. IEEE/ACM Symposium on Information Processing in Sensor Networks (IPSN*), Apr. 2005, pp. 364-369

[2] I.F. Akyildiz, T. Melodia, K.R. Chowdhury, A survey on wireless multimedia sensor networks, *Computer Networks,* 51 (4) (2007) 921-960.

[3] G. Mathur, P. Desnoyers, D. Ganesan, P. Shenoy, Ultra-low power data storage for sensor networks, in: *Proceedings of the International IEEE/ACM Syposium on Information Processing in Sensor Networks (IPSN),* 2006, pp. 374-381.
[4] C. Chrysafis, A. Ortega, Line-based, reduced memory, wavelet image compression*, IEEE Transactions on Image Processing,* 9 (3) (2000) 378-389.
[4] J. Oliver, M. Malumbres, On the design of fast wavelet transform algorithms with low memory requirements, *IEEE Transactions on Circuits and Systems for Video Technology,* 18 (2) (2008) 237-248.
[5] S. Rein and M. Reisslein. Low-Memory Wavelet Transforms for Wireless Sensor Networks: A Tutorial, *IEEE Communications Surveys and Tutorials*, 14(2): - , Second Quarter 2011. Digital Object Identifier 10.1109/SURV.2011.100110.00059
[6] S. Rein and M. Reisslein. Performance Evaluation of the Fractional Wavelet Filter: A Low-Memory Image Wavelet Transform for Multimedia Sensor Networks , *Ad Hoc Networks*, 9(4):482-496, June 2011. doi: 10.1016/j.adhoc.2010.08.004
[7] S. Rein, S. Lehmann, and C. Guehmann,"Wavelet image two-line coder for wireless sensor node with extremely little RAM", in Proc. of the IEEE Data Compression Conference (DCC), Snowbird, UT, Mar. 2009, pp. 252-261

Stephan Rein studied electrical and telecommunications engineering at RWTH Aachen University and Technical University (TU) Berlin, where he received the Dipl.-Ing. degree in December 2003 and the Ph.D. degree in January 2010. He was a research scholar at Arizona University in 2003, where he conducted research on voice quality evaluation and developed an audio content search machine. From February 2004 to March 2009 he was with the Wavelet Application Group at TU Berlin developing text and image compression algorithms for mobile phones and sensor networks. Since July 2009 he is with the Telecommunication Networks Group at TU Berlin, where he is currently working on multimedia delivery to mobile devices.

Martin Reisslein is an Associate Professor in the School of Electrical, Computer, and Energy Engineering at Arizona State University (ASU), Tempe. He received his Ph.D. in systems engineering from the University of Pennsylvania in 1998. From July 1998 through October 2000 he was a scientist with the German National Research Center for Information Technology (GMD FOKUS), Berlin and lecturer at the Technical University Berlin. He currently serves as Associate Editor for the IEEE/ACM Transactions on Networking and for Computer Networks. He maintains an extensive library of video traces for network performance evaluation, including frame size traces of MPEG-4 and H.264 encoded video, at http://trace.eas.asu.edu. His research interests are in the areas of multimedia networking, optical access networks, and engineering education.