

# Efficiency Analysis of the Autofocusing Algorithm Based on Orthogonal Transforms

Przemysław Śliwiński, Krzysztof Berezowski, Piotr Patronik, Paweł Wachel

Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wrocław, Poland.  
Email: [przemyslaw.sliwinski@pwr.wroc.pl](mailto:przemyslaw.sliwinski@pwr.wroc.pl), [krzysztof.berezowski@pwr.wroc.pl](mailto:krzysztof.berezowski@pwr.wroc.pl), [piotr.patronik@pwr.wroc.pl](mailto:piotr.patronik@pwr.wroc.pl), [pawel.wachel@pwr.wroc.pl](mailto:pawel.wachel@pwr.wroc.pl)

Received September 2013

## ABSTRACT

Efficiency of the autofocusing algorithm implementations based on various orthogonal transforms is examined. The algorithm uses the variance of an image acquired by a sensor as a focus function. To compute the estimate of the variance we exploit the equivalence between that estimate and the image orthogonal expansion. Energy consumption of three implementations exploiting either of the following fast orthogonal transforms: the discrete cosine, the Walsh-Hadamard, and the Haar wavelet one, is evaluated and compared. Furthermore, it is conjectured that the computation precision can considerably be reduced if the image is heavily corrupted by the noise, and a simple problem of optimal word bit-length selection with respect to the signal variance is analyzed.

**Keywords:** Auto-Focusing; Image Variance; Discrete Orthogonal Transforms; Word-Length Selection; Architectural Performance Evaluation

## 1. Introduction

We say that the image is sharp (in focus) when it is the most detailed representation of the actual scene seen via a lens. This intuitive observation led to many heuristic algorithms which measure, like for instance (see e.g. [1-3]):

- the variance of the image,
- the sum of squared Laplacian of the image, or
- the sum of squared values of the image transform by selected edge-detection algorithms.

In this work we consider the first measure for the following reasons:

- it can be shown that the variance of the image produced by the lens is a unimodal function which has a maximum when the image is in focus. Moreover,
- the variance can effectively be estimated even in the presence of a random noise.

Automatic focusing seems to not only be a desired feature of consumer electronic devices like digital cameras and camcorders, but also an important tool used in security or industrial applications (like surveillance or microscopy; cf. [1,4]). The focusing algorithm whose efficiency we examine is a *passive algorithm* (that is, it does not require additional equipment) and operates on the data acquired by the image sensor; see e.g. [5]. We use a variance of the image data as the focus function and to find

its (single) maximum (*i.e.* to get the sharpest image) we employ the *golden section search (GSS)* algorithm; see [6]. Note that fast yet precise focusing in both surveillance and microscopy is rather cumbersome since we have to deal there with a thin *depth-of-focus (DOF)* issue (in the former, it comes from application of large aperture (*fast*) lenses while in the latter, this is a consequence of short distances between a scene and a lens). A thin DOF makes the maximum search problem much harder since it typically implies a flat-tailed focus function with a steeply-sloped peak whose unimodality is easily violated in a noisy environment.<sup>1</sup>

We begin with a problem statement and a focus function formula. Next, we present the focusing algorithm and three implementations of the variance estimate computation routines based on either:

- the discrete cosine,
- the Walsh-Hadamard, or
- the Haar wavelet orthogonal transforms, respectively.

Then, we experimentally establish the energy efficiency of each implementation using an ARM processor simulator. Finally, we shortly examine the minimum word-length selection as a denoising algorithm in the

<sup>1</sup>In the *shape from focus (SFF)* applications, where the three dimensional (3D) scene is reconstructed from a sequence of two-dimensional (2D) images focused at the different distances, thinner DOFs enable higher accuracy reconstructions; see e.g. [3,7,8].

presence of a thermal (Gaussian) noise.

## 2. Autofocusing Algorithm

The proposed algorithm can be considered as the solution to a stochastic approximation problem. The captured scene, the lens system and the image sensor are modelled as follows; see **Figure 1** and *cf.* [5]:

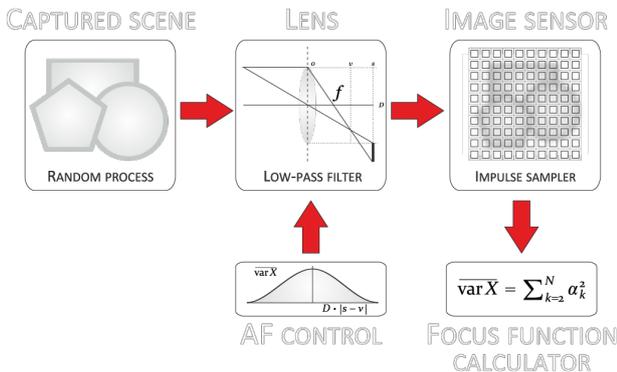
1) The scene is a 2D homogenous second-order stationary process with unknown distribution and unknown correlation function; *cf.* [9].

2) The lens has a circular aperture system that satisfies the first-order optics laws, and is represented by a centered moving average filter of order proportional to the distance  $|s-v|$  of the sensor from the image plane and to the size of the lens aperture  $D$ ; *cf.* [10].

3) The image sensor acts as an impulse sampler on the lens-produced process.

The autofocusing algorithm simply seeks for the maximum of the focus function, which in our case is the variance of the image produced by the lens for a given scene. The assumptions above allow demonstrating the unimodality property of this focus function, and hence, enable an application of the simple *golden-section search* algorithm to find the maximum. Since, for a given scene, lens and a sensor, the number of steps performed by the GSS algorithms is the same, it suffices to compare effectiveness of the variance estimation implementations in order to assess the effectiveness of the whole AF algorithm.

We will now present the equivalence between the variance estimate and the orthogonal expansion of the image data. Thanks to this equivalence we can not only compute the estimate (which can clearly be computed directly from a standard variance estimate formula), but we can also interpret the image as a regression function (which can in turn be estimated separately in order to remove the noise from the image). For simplicity we examine the one-dimensional case (which can be justified by the symmetry argument since, by Assumption 2, the lens aperture is circular).



**Figure 1.** A block diagram of the AF circuit [5].

Let  $X = [x_1 \dots x_N]^T$  be the vector representing the raw image (*i.e.* a sample function of the process produced by the lens). The variance of such an image can be estimated by the standard formula

$$\overline{\text{var } X} = N^{-1} \sum_{n=1}^N x_n^2 - \left( N^{-1} \sum_{n=1}^N x_n \right)^2. \quad (1)$$

Also, the vector  $X$  can be expanded in a discrete orthogonal series  $\{\varphi_k\}$ ,  $k = 1, \dots, N$ , so that

$$X = \sum_{k=1}^N \alpha_k \varphi_k(k/N), \text{ where } \alpha_k = N^{-1} \sum_{n=1}^N x_n \varphi_k(n/N).$$

By the Parseval's identity, we have that

$$N^{-1} \sum_{n=1}^N x_n^2 = \alpha_1^2 + \sum_{k=2}^N \alpha_k^2.$$

Hence, if the first term  $\varphi_1$  of the orthogonal series expansion is a constant function (as it is the case in all three considered series), then we can easily ascertain that the squared value of the first expansion coefficient  $\alpha_1 = N^{-1} \sum_{n=1}^N x_n$ , equals to second term in the variance estimate (1). As a result, we obtain the equivalent variance estimation formula

$$\overline{\text{var } X} = \sum_{k=2}^N \alpha_k^2,$$

in which the expansion coefficients  $\{\alpha_k\}$  are computed using the appropriate fast transform algorithm [11,12].

## 3. Orthogonal Transforms

Here we recollect some basic properties of the considered transforms. Each of them has the corresponding matrix representation

$$Y = AX,$$

where  $A$  is a square  $N \times N$  orthogonal matrix (*i.e.* a matrix of the unit *spectral norm*,  $\|A\|_2 = 1$ ) and  $Y = [\alpha_1 \dots \alpha_N]^T$  is the vector collecting the transformed image (*i.e.* the transform coefficients).

### 3.1. DCT Transform Matrix

The matrix of the *discrete cosine transform (DCT)* consists of the cosine basis functions sampled at the uniform grid. The matrix is orthogonal and with entries; see *e.g.* [12,13]:

$$A[k, n] = \cos\left(\frac{\pi k}{N} \left(n + \frac{1}{2}\right)\right), k, n = 0, \dots, N-1.$$

Observe that:

- The matrix dimension  $N$  is an arbitrary natural number (*i.e.* it is not restricted to powers of two like the remaining transforms (this is because the orthogonal systems of the trigonometric functions remain ortho-

gonal on a uniform discrete grid [14,15]).

- The matrix entries are real numbers, that is, in the transform computations their truncated (approximated) values are used.
- A direct implementation has a polynomial complexity  $O(N^2)$ .
- The *Fast Discrete Cosine Transform* is an *in situ* algorithm and requires only  $O(N \log N)$  operations, *i.e.*, can be computed in a linearithmic (*i.e. quasilinear*) time.

**Example 1:** For  $N = 4$  the DCT matrix takes the form

$$A = \frac{1}{2\sqrt{2}} \begin{bmatrix} \frac{\sqrt{2}}{\sqrt{\sqrt{2}+2}} & \frac{\sqrt{2}}{\sqrt{2-\sqrt{2}}} & \frac{\sqrt{2}}{-\sqrt{2-\sqrt{2}}} & \frac{\sqrt{2}}{-\sqrt{\sqrt{2}+2}} \\ \frac{\sqrt{2}}{\sqrt{2-\sqrt{2}}} & -\frac{\sqrt{2}}{\sqrt{\sqrt{2}+2}} & -\frac{\sqrt{2}}{\sqrt{\sqrt{2}+2}} & \frac{\sqrt{2}}{\sqrt{2-\sqrt{2}}} \\ \frac{\sqrt{2}}{\sqrt{2-\sqrt{2}}} & -\frac{\sqrt{2}}{\sqrt{\sqrt{2}+2}} & \frac{\sqrt{2}}{\sqrt{\sqrt{2}+2}} & -\frac{\sqrt{2}}{\sqrt{2-\sqrt{2}}} \end{bmatrix}$$

### 3.2. Walsh-Hadamard Transform Matrix

The matrix of the *Walsh-Hadamard (W-H) transform* has the following properties; see e.g. [11,16]:

- The matrix dimension  $N$  is a dyadic natural number (*i.e.* it is a power of two).
- The entries are equal either 1 or -1, *i.e.* there is no multiplication in the transform algorithm.
- A direct implementation has a polynomial complexity  $O(N^2)$ .
- The *Fast Walsh Transform* is an *in situ* algorithm and requires  $O(N \log N)$  operations, *i.e.*, can also be computed in a linearithmic (*i.e. quasilinear*) time.

**Example 2:** For  $N = 4$  the W-H matrix takes the form

$$A = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

### 3.3. Haar Transform Matrix

The matrix corresponding to the *Haar Wavelet Transform* algorithm is somehow unique when compared to the former two. In particular:

- A direct implementation results in a linearithmic complexity  $O(N \log N)$ .
- The *Fast Haar Transform* requires merely  $O(N)$  operations, *i.e.*, it can be computed in a linear time (its lifting version can furthermore be computed *in situ*). Nevertheless,
- Its dimension is an integral power of two (like in the *W-H* case—in order to preserve orthogonality).
- The entries are not integer like in the DCT case (they are integral powers of  $\sqrt{2}$  instead).

**Example 3:** The matrix of the four-point Haar transform has the following entries; see e.g. [12]:

$$A = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

Note that the Haar transform matrices become sparser (*i.e.* they have more-and-more zero entries) as their dimensions grow.

## 4. Energy Consumption Evaluation

From the formal point of view, all implementations are equivalent, that is, given the same image data, they always yield the same value of the variance estimate. However, different numerical properties of the transforms suggest that the energy consumption of each implementation may vary significantly.

In order to verify this conjecture, we measured experimentally the energy consumptions of the variance calculation. The experiments were run on two sample pictures (**Figure 2**). The pictures were scaled to three sizes:  $32 \times 32$ ,  $64 \times 64$ , and  $128 \times 128$ , and converted to 8-bit grayscale bitmaps. We used the DCT, Walsh-Hadamard and Haar transforms implemented in the *FXT* library, [17], and compiled them with the standard *GCC* compiler.<sup>2</sup> For our energy consumption assessment we employed the *Panalyzer* simulator of the *ARM* processor, which is a popular *SimpleScalar* simulator augmented with the power models of this processor.<sup>3</sup>

The calculations were performed using the double precision floating-point numbers. Additionally, in attempt to exploit the specifics of the Walsh-Hadamard transform, we calculated the variance using the fixed-point implementation with the help of a separate integer number routine<sup>4</sup>. In all simulations, we measured the total energy consumption of the processor microarchitecture.

The results are shown in **Figure 3**. Clearly, the energy consumption does not depend on the particular image and grows with the image size. Moreover:

- The implementation based on the Haar transform was the most efficient energy consumption-wise.
- Next in order was the implementation based on the DCT transform.
- The least efficient was the Walsh-Hadamard transform-based implementation, both in floating- and fixed-point versions.

While it seems to be obvious that the Haar transform implementation has the best efficiency (as the result of the transform's linear computational complexity), the smaller energy consumption of the DCT implementation in relation to the Walsh one is somehow surprising and

<sup>2</sup>GCC 4.0.1 compiler, all optimization options disabled.

<sup>3</sup>Sim-analyzer 2.0.3; see <http://web.eecs.umich.edu/~panalyzer/>

<sup>4</sup>See: <http://math.ewha.ac.kr/~jylee/SciComp/sc-crandall/walsh.c>



Figure 2. The two sample images I and II.

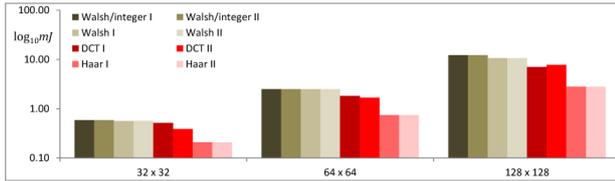


Figure 3. Energy consumption of the three implementations measured in mJ for the processor clocked at 500 MHz.

requires further study (it can be related with e.g. a better low-level optimization of the DCT transform in the FXT library). In turn, the almost identical results of the fixed- and floating-point Walsh implementations suggest that only a very small fraction of energy is consumed by the arithmetic instructions in relation to the whole algorithm.

## 5. Word-Length Selection and Denoising

In this section we consider the word-length problem (*i.e.* the required precision of the data representation) in the presence of noise. The stochastic nature of the noise nature creates random ‘fake’ local maxima in the focus function estimate in which the GSS algorithm can easily stuck (see however the analysis of the GSS algorithm in [18], where the probability of such an event is shown to vanish with the growing number of pixels). The simplest way to attenuate the influence of the noise is to average multiple pictures taken at each focus distance. This solution however results in a slower and more power consuming focusing routine. In embedded systems, one should therefore prefer more energy efficient, single-image, approaches based on e.g. nonparametric regression estimation; see [19,20].

Inspired by the one of the most prominent estimation techniques, the wavelet shrinking, we examine here the simple algorithm reducing the noise and based on the word-length selection problem solution; *cf.* e.g. [21]. We assume here that the additive random noise  $\varepsilon_k$  is *i.i.d.* and has the Gaussian distribution  $N(0, \sigma_\varepsilon^2)$ . Thus, each pixel value can be described as

$$x_n = u_n + \varepsilon_n.$$

where  $u_n$  are the actual (noiseless) raw images. The un-

iformly quantized (*i.e.* finite precision) versions of  $x_n$ , for  $M$  fractional bits, can be represented as

$$x_n = \frac{2^M \lfloor u_n + \varepsilon_n + 1/2 \rfloor}{2^M},$$

where  $\lfloor \cdot \rfloor$  denotes the standard *floor function*.

To establish the minimum number of fractional bits  $M$  such that the inaccuracy in the final (transformed) image imposed by the quantization error do not exceed the variance error introduced by the noise  $\varepsilon_n$ , we consider the following *mean squared error (MSE)* error of the transformed image

$$\begin{aligned} \text{MSE } Y &= E \sum_{k=1}^N (\alpha_k - \bar{\alpha}_k)^2 \\ &\leq 2E \sum_{k=1}^N (\alpha_k - \hat{\alpha}_k)^2 + 2E \sum_{k=1}^N (\hat{\alpha}_k - \alpha_k)^2, \end{aligned}$$

where  $\{\alpha_k\}$  are the pixels of the transformed image computed from the noiseless raw image pixels  $\{u_n\}$ ,  $\{\hat{\alpha}_k\}$  are the pixels of the transformed image computed in exact arithmetic from the noisy pixels  $\{x_n\}$ , and  $\{\bar{\alpha}_k\}$  are its quantized versions. For simplicity we will examine the above error for the Walsh-Hadamard transform only. We have

$$\begin{aligned} \alpha_k &= \frac{1}{N} \sum_{n=1}^N u_n A[k, n], \\ \hat{\alpha}_k &= \frac{1}{N} \sum_{n=1}^N (u_n + \varepsilon_n) A[k, n], \text{ and} \\ \bar{\alpha}_k &= \frac{1}{N} \sum_{n=1}^N \frac{2^M \lfloor u_n + \varepsilon_n + 1/2 \rfloor}{2^M} A[k, n]. \end{aligned}$$

Exploiting the orthonormality property of the transform matrix  $A$  and the independence of the noise  $\varepsilon_n$ , we obtain that

$$E \sum_{k=1}^N (\alpha_k - \hat{\alpha}_k)^2 = \sigma_\varepsilon^2 \text{ and } E \sum_{k=1}^N (\hat{\alpha}_k - \bar{\alpha}_k)^2 \leq N 2^{-2(M+1)}.$$

Comparison of these two error terms yields the following word-length selection formula

$$M = \left\lceil \frac{\log_2 N - \log \sigma_\varepsilon^2}{2} \right\rceil,$$

which guarantees that, given the number of fractional bits is  $M$ , the quantization-induced inaccuracy does not exceed (in the MSE sense) the noise-induced inaccuracy.

**Remark 1:** Since the noise ‘‘occupies’’ the least significant bits which are truncated by the quantization, one can expect that for the selected  $M$  the noise will also be (partially) shrunk (this aspect needs however a more careful further examination). Alternatively, one can also consider performing the classic shrinkage algorithm on the transformed image (prior to the variance calculation) in order to remove the noise [19,20].

## 6. Final Remarks

In the paper we presented three implementations of the image variance estimate evaluation which is the core and the most computationally demanding part of the autofocusing algorithm. We provided an experimental evidence that the implementation based on the fast Haar transform has a much better (by a wide margin) energy efficiency than the remaining two implementations based on the discrete cosine and the Walsh-Hadamard transforms. Somehow unexpectedly, the experiments revealed that there is no advantage of using the integer number Walsh-Hadamard transform over the cosine one. Finally, having in mind an ASIC implementation of the algorithm, we also proposed the word-selection algorithm which determines the required precision of the image data with respect to the size of the image data and to the variance of the noise present in the data. The actual benefit of this algorithm needs however to be verified experimentally.

## Acknowledgements

The work is supported by the NCN grant UMO-2011/01/B/ST7/00666.

## REFERENCES

- [1] F. C. A. Groen, I. T. Young and G. Lighthart, "A Comparison of Different Focus Functions for Use in Autofocus Algorithms," *Cytometry*, Vol. 6, No. 2, 1985, pp. 81-91. <http://dx.doi.org/10.1002/cyto.990060202>
- [2] M. Subbarao and J.-K. Tyan, "Selecting the Optimal Focus Measure for Autofocusing and Depth-from-Focus," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 8, 1998, pp. 864-870. <http://dx.doi.org/10.1109/34.709612>
- [3] A. N. R. R. Hariharan, "Shape-from-Focus by Tensor Voting," *IEEE Transactions on Image Processing*, Vol. 21, No. 7, 2012, pp. 3323-3328. <http://dx.doi.org/10.1109/TIP.2012.2190612>
- [4] E. Krotkov, "Focusing," *International Journal of Computer Vision*, Vol. 1, No. 3, 1987, pp. 223-237. <http://dx.doi.org/10.1007/BF00127822>
- [5] P. Śliwiński, "Autofocusing with the Help of Orthogonal Series Transforms," *International Journal of Electronics and Telecommunications*, Vol. 56, No. 1, 2010, pp. 31-37. <http://dx.doi.org/10.2478/v10177-010-0004-5>
- [6] J. Kiefer, "Sequential Minimax Search for a Maximum," *Proceedings of the American Mathematical Society*, Vol. 4, No. 3, 1953, pp. 502-506. <http://dx.doi.org/10.1090/S0002-9939-1953-0055639-3>
- [7] S. K. Nayar and Y. Nakagawa, "Shape from Focus," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 8, 1994, pp. 824-831. <http://dx.doi.org/10.1109/34.308479>
- [8] K. S. Pradeep and A. N. Rajagopalan, "Improving Shape from Focus Using Defocus Cue," *IEEE Transactions on Image Processing*, Vol. 16, No. 7, 2007, pp. 1920-1925. <http://dx.doi.org/10.1109/TIP.2007.899188>
- [9] J. W. Goodman, "Statistical Optics," Wiley-Interscience, New York, 2000.
- [10] S. J. Ray, "Applied Photographic Optics," 3rd Edition, Focal Press, Oxford, 2004.
- [11] K. Beauchamp, "Applications of Walsh and Related Functions," Academic Press, Waltham, 1984.
- [12] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, "Numerical Recipes in C: The Art of Scientific Computing," Cambridge University Press, Cambridge, 1993.
- [13] D. Taubman and M. Marcellin, "JPEG2000. Image Compression Fundamentals, Standards and Practice," Kluwer Academic Publishers, 2002, Vol. 642. <http://dx.doi.org/10.1007/978-1-4615-0799-4>
- [14] G. Szego, "Orthogonal Polynomials," 3rd Edition, American Mathematical Society, Providence, RI, 1974.
- [15] V. Mathews and G. Sicuranza, "Polynomial Signal Processing," Wiley, New York, 2000.
- [16] B. Fino and V. Algazi, "Unified Matrix Treatment of the fast Walsh-Hadamard Transform," *IEEE Transactions on Computers*, Vol. 100, No. 11, 1976, pp. 1142-1146. <http://dx.doi.org/10.1109/TC.1976.1674569>
- [17] J. Arndt, "Matters Computational: Ideas, Algorithms, Source Code," Springer-Verlag New York, Inc., New York, NY, USA, 2010.
- [18] P. Śliwiński and P. Wachel, "Application of Stochastic Counterpart Optimization to Contrast-Detection Autofocusing," *International Conference on Advances in Computing, Communications and Informatics (ICACCI 2013)*, Mysore, India, 22-25 August 2013, pp. 333-337. <http://dx.doi.org/10.1109/ICACCI.2013.6637193>
- [19] W. Härdle, G. Kerkycharian, D. Picard and A. Tsybakov, "Wavelets, Approximation, and Statistical Applications," Springer-Verlag, New York, 1998. <http://dx.doi.org/10.1007/978-1-4612-2222-4>
- [20] L. Györfi, M. Kohler, A. Krzyżak and H. Walk, "A Distribution-Free Theory of Nonparametric Regression," Springer-Verlag, New York, 2002. <http://dx.doi.org/10.1007/b97848>
- [21] G. Constantinides, P. Cheung and W. Luk, "Wordlength Optimization for Linear Digital Signal Processing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 22, No. 10, 2003, pp. 1432-1442. <http://dx.doi.org/10.1109/TCAD.2003.818119>