

## Research Article

# Phase-only Correlation Function by Means of Hartley Transform

Lorenzo Cozzella\* and Giuseppe Schirripa Spagnolo

Department of Mathematics and Physics, University of "Roma Tre", Italy

\*Corresponding author

Lorenzo Cozzella, Department of Mathematics and Physics, University of "Roma Tre", Italy, Email: lorenzo.cozzella@uniroma3.it

Submitted: 21 August 2014

Accepted: 22 August 2014

Published: 22 August 2014

Copyright

© 2014 Cozzella et al.

OPEN ACCESS

## Abstract

In image processing or pattern recognition, Fourier Transform is widely used for frequency-domain analysis. In particular, the Phase Only Correlation (POC) method demonstrates high robustness and accuracy in the pattern matching and the image registration. However, there is a disadvantage in required memory machine because of the calculation of 2D-FFT. In this case, Hartley transform can be a very good substitute for more commonly used Fourier transform when the real input data are concerned. The Hartley transform is similar to the Fourier transform, but it is free from the need to process complex numbers. It also has some distinctive features that make it an interesting choice when a greater efficiency in memory requirements is needed. In this paper we show the correspondence between the Phase-Only Correlation (POC) function obtained by means of FFT and by FHT.

## Keywords

- Cross-correlation
- Phase correlation
- Fast Fourier transform
- Hartley transform
- Pattern recognition
- Image matching

## ABBREVIATIONS

FFT: Fast Fourier Transform; FHT: Fast Harley Transform; POC: Phase-Only Correlation; PIV: Particle Image Velocimetry

## INTRODUCTION

The automatic determination of similarity between two structured data sets is fundamental to the disciplines of pattern recognition and image processing [1,2]. The cross-correlation between two data sets is a good method to measure their similarity. The cross-correlation function is used extensively in pattern recognition and signal detection. We know that projecting one signal onto another is a means of measuring how much of the second signal is present in the first. Since a digital image can be considered as a dataset, cross-correlation can be used to detect the similarity and the lagging between the images. In general, the standard cross correlation [3,4] yields several broad peaks and a main peak whose maximum is not always; therefore, it is difficult to locate the maximum; see Figure 1 (c). One of the alternatives to the cross-correlation function is the Fourier Phase-Only Correlation (POC) function [5,6]. The POC function yields an even sharp maximum at the best match point, as shown in Figure 1(d); therefore, it is easy to locate the maximum.

The Phase-Only Correlation (POC) method demonstrates high robustness and accuracy in the pattern matching and the image registration. However, there is a disadvantage in required memory machine because of the calculation of 2D-FFT.

The Fast Hartley Transform (FHT) can be a valid alternative to Fast Fourier Transform (FFT) [7]. The Hartley transform [8,9] resembles a Fourier transform but it is free from the need to process complex numbers. The Hartley transform also has some better properties and faster algorithms than the Fourier

one, therefore it can represent a valid alternative, particularly interesting when a greater efficiency in memory requirements is needed.

In this work we present a Hartley transform algorithm for accurate and fast elaboration of POC.

This paper is organized as follows: Section 2 give the definition of POC function and its basic properties, the properties of Harley transform and the definition of POC in Hartley space. Section 3 presents a set of experiments for evaluating performance of the proposed methods. In section 4, we end with some conclusions.

## MATERIALS AND METHODS

### Phase-Only Correlation (POC)

The Phase-Only Correlation (POC) function (or simply "phase-correlation") has been successfully applied to high-accuracy image registration tasks for computer vision applications [10,11], for Fingerprint Matching [12], for Iris Recognition [13], Palmar Recognition [14], in PIV analysis [15,16] and in security application [17-19].

Phase correlation algorithm uses the cross-power spectrum to get the translation factor between two images. Assume that there are two images  $I_1(x, y)$  and  $I_2(x, y)$  and the translation between them is as following:

$$I_2(x, y) = I_1(x - x_0, y - y_0). \quad (1)$$

The Fourier transformation:

$$F_2(u, v) = F_1(u, v) \cdot \exp[-j(u x_0 + v y_0)]. \quad (2)$$

In equation (2),  $F_1(u, v)$  and  $F_2(u, v)$  are the Fourier transform of  $I_1(x, y)$  and  $I_2(x, y)$ . The cross-power

spectrum will be:

$$\frac{F_1^*(u, v) \cdot F_2^*(u, v)}{|F_1^*(u, v) \cdot F_2^*(u, v)|} = \exp[-j(u x_0 + v y_0)] \quad (3)$$

with  $F_1^*(u, v)$  the conjugate function of  $F_1(u, v)$ . We can get an impulse function  $\delta(x - x_0, y - y_0)$  about the value of translation invariant  $x_0$  and  $y_0$  by using Fourier inverse transforming to equation (3).

In other words, the phase correlation surface is defined as:

$$r(x, y) = F^{-1} \left[ \frac{F_1^*(u, v) \cdot F_2^*(u, v)}{|F_1^*(u, v) \cdot F_2^*(u, v)|} \right] \quad (4)$$

The phase correlation method provides a distinct sharp peak. Furthermore, it is robust to those types of noise that are correlated to the image function, e.g., uniform variations of illumination, offsets in average intensity.

Figure 2 shows two examples of phase correlation surface.

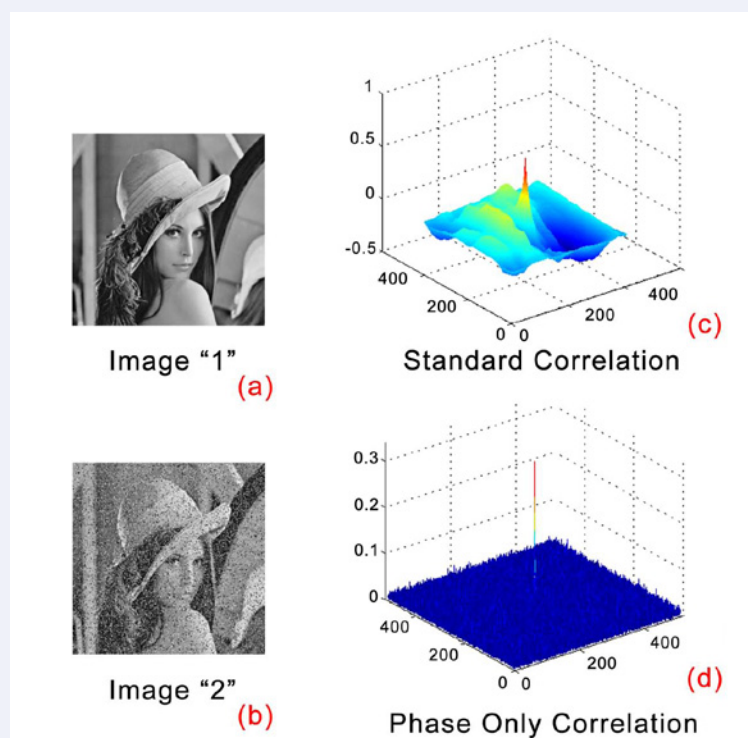
The coordinate  $(x_0, y_0)$  of the maximum of the real-value array  $r(x, y)$  can be used as an estimate of the horizontal and vertical components of translation between  $I_1(x, y)$  and  $I_2(x, y)$  as follows:

$$(x_0, y_0) = \arg \max \text{Re} \{r(x, y)\} \quad (5)$$

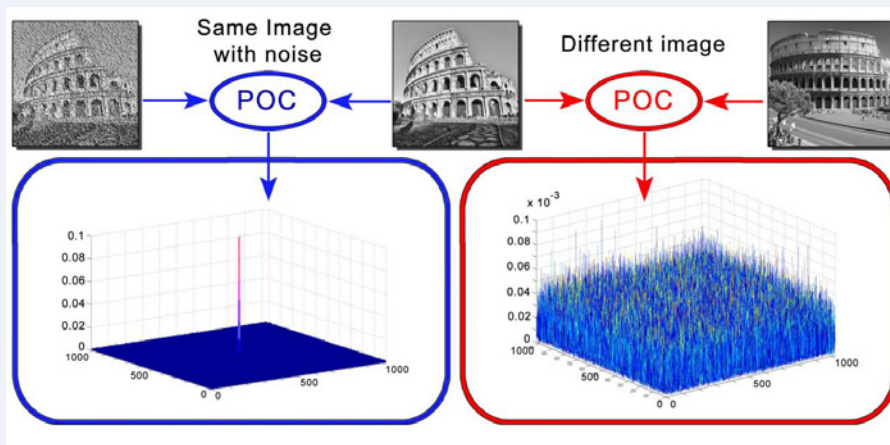
Using the FHT (Fast Hartley Transform) the equation (4), and consequently also the equation (5), can be compute efficiently and without information lost.

## The Hartley transform

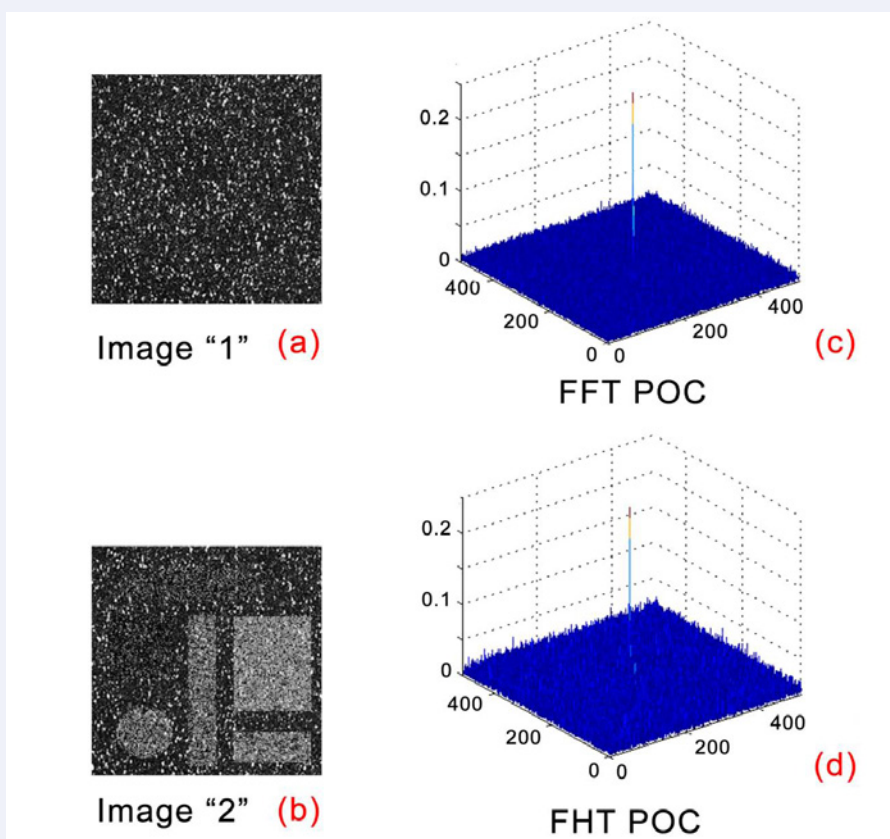
The Hartley transform, first published in 1942 [20], keeps all the useful properties of the Fourier transform and can be used to obtain the power spectrum and perform convolution directly through the output real-valued data without calculating the real and imaginary parts first. The use of the Hartley transform also reduces the processing time, and needs less memory. Therefore, it is particularly well suited for application on mobile device.



**Figure 1** Example of ordinary correlation and POC function. (a) Image "1"; (b) Image "2"; (c) ordinary correlation function between Image "1" and Image "2"; (d) POC function between Image "1" and Image "2".



**Figure 2** Examples of POC function.



**Figure 3** Examples of POC function obtained from two speckle images.

The Hartley transform can be obtained from the Fourier integral by replacing the exponential function  $\exp(-i\omega t) = \cos(\omega t) + i \sin(\omega t)$  by  $\text{cas}(\omega t) = \cos(\omega t) + \sin(\omega t)$ .

The Hartley transform  $S_H(f)$  of a real signal  $s(t)$  and its inverse transform are defined as:

$$S_H(f) = \int_{-\infty}^{+\infty} s(t) \text{cas}(2\pi f t) dt, \quad (6)$$

$$s(t) = \int_{-\infty}^{+\infty} S_H(f) \text{cas}(2\pi f t) df, \quad (7)$$

with  $\text{cas}(2\pi f t) = \cos(2\pi f t) + \sin(2\pi f t)$ .

The Fourier and Hartley transforms are very similar and share many properties. Consequently, many applications using the Fourier transform can be performed by the Hartley transform. Table I shows the main theorems for both transforms.

To derive the relationship between the Fourier and Hartley transforms, their symmetry must be considered. Let us split the Hartley transform  $S_H(f)$  into its even and odd parts  $E_H(f)$  and  $O_H(f)$ , so that:

$$S_H(f) = E_H(f) + O_H(f). \quad (8)$$

The even part of the function if what we get reversing the function (changing  $f$  to  $-f$ ), adding the reversed function to the original and dividing by two, that is

$$E_H(f) = \frac{S_H(f) + S_H(-f)}{2}. \quad (9)$$

The odd part is formed by subtracting the reversed function and dividing by two:

$$O_H(f) = \frac{S_H(f) - S_H(-f)}{2}. \quad (10)$$

Any function may be split uniquely into even and odd parts and from the even and odd parts, if given, the original function may be reconstituted uniquely.

Obviously, also the Fourier transform can be divided into even and odd parts,  $E_F(f)$  and  $O_F(f)$ . Furthermore, it can be shown that

$$E_F(f) = \text{Re}[S_F(f)], \quad (11)$$

$$O_F(f) = \text{Im}[S_F(f)]. \quad (12)$$

By considering the definitions of Hartley and Fourier transforms, we have  $E_H(f) = E_F(f)$  and  $O_H(f) = -O_F(f)$ . Therefore

$$S_H(f) = \text{Re}[S_F(f)] - \text{Im}[S_F(f)], \quad (13)$$

$$\begin{aligned} S_F(f) &= E_F(f) + iO_F(f) = E_H(f) - iO_H(f) \\ &= \frac{1}{2} \{ [S_H(f) + S_H(-f)] - i[S_H(f) - S_H(-f)] \}. \end{aligned} \quad (14)$$

The bidimensional definitions for the Hartley transform and its inverse are [9]

$$G_H(u, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} g(x, y) \text{cas}[2\pi(ux + vy)] dx dy, \quad (15)$$

$$g(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} G_H(u, v) \text{cas}[2\pi(ux + vy)] du dv, \quad (16)$$

The generalization of splitting one-variable functions into even and odd parts is the decomposition into symmetrical and antisymmetrical parts

$$g(x, y) = g_{\text{symm}}(x, y) + g_{\text{antisymm}}(x, y), \quad (17)$$

with

$$g_{\text{symm}}(x, y) = \frac{1}{2} [g(x, y) + g(-x, -y)], \quad (18)$$

$$g_{\text{antisymm}}(x, y) = \frac{1}{2} [g(x, y) - g(-x, -y)]. \quad (19)$$

Analogously to the one-dimensional case, the real part of a 2D Fourier transform is symmetrical and the imaginary part is

antisymmetrical. Therefore

$$E_H(u, v) = \frac{G_H(u, v) + G_H(-u, -v)}{2}, \quad (20)$$

$$O_H(u, v) = \frac{G_H(u, v) - G_H(-u, -v)}{2}, \quad (21)$$

and  $G_F(u, v) = E_H(u, v) - iO_H(u, v)$ , which is the 2D extension of Equation (14)

The cross-correlation  $r(x, y)$  of two functions  $p(x, y)$  and  $q(x, y)$  is:

$$r(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} p(x', y') q(x + x', y + y') dx' dy'. \quad (22)$$

Evaluating the cross-correlation through the integral in Equation (22) is complicated. This task can be significantly simplified in the Fourier space, where  $R_F(u, v) = P_F^*(u, v) \cdot Q_F(u, v)$ ; with  $R_F(u, v)$ ,  $P_F(u, v)$  and  $Q_F(u, v)$  being the Fourier transforms of  $r(x, y)$ ,  $p(x, y)$  and  $q(x, y)$ , respectively. Therefore

$$r(x, y) = \text{FFT}^{-1}\{P_F^*(u, v) \cdot Q_F(u, v)\} = \text{FFT}^{-1}\{R_F\}. \quad (23)$$

In general, the presence of noise makes it difficult the "exact" localization of the cross-correlation peak. In other hand, using POC

$$r(x, y) = \text{FFT}^{-1}\left\{\frac{R_F(u, v)}{|R_F(u, v)|}\right\}. \quad (24)$$

we have a good compromise between peak sharpness and noise tolerance.

Using the Hartley transform,  $R_F$  can be written as [21]

$$\begin{aligned} R_F(u, v) &= \frac{R_H(u, v) + R_H(-u, -v)}{2} - i \frac{R_H(u, v) - R_H(-u, -v)}{2} \\ &= \left[ \frac{P_H(u, v) + P_H(-u, -v)}{2} + i \frac{P_H(u, v) - P_H(-u, -v)}{2} \right] \\ &\quad \cdot \left[ \frac{Q_H(u, v) + Q_H(-u, -v)}{2} - i \frac{Q_H(u, v) - Q_H(-u, -v)}{2} \right] \quad (25) \\ &= \frac{P_H(u, v) \cdot Q_H(u, v) + P_H(-u, -v) \cdot Q_H(-u, -v)}{2} \\ &\quad + i \frac{P_H(u, v) \cdot Q_H(-u, -v) - P_H(-u, -v) \cdot Q_H(u, v)}{2} \end{aligned}$$

where  $R_H(u, v)$  is the Hartley transform of  $r(x, y)$ . From Equation (25), we obtain

$$\begin{aligned} R_H(u, v) + R_H(-u, -v) &= P_H(u, v) \cdot Q_H(u, v) - P_H(u, v) \cdot Q_H(-u, -v) \\ R_H(u, v) - R_H(-u, -v) &= P_H(-u, -v) \cdot Q_H(u, v) - P_H(u, v) \cdot Q_H(-u, -v) \end{aligned} \quad (26)$$

and finally

$$\begin{aligned} R_H(u, v) &= \frac{P_H(u, v) \cdot Q_H(u, v) - P_H(u, v) \cdot Q_H(-u, -v)}{2} \\ &\quad + \frac{P_H(-u, -v) \cdot Q_H(u, v) + P_H(-u, -v) \cdot Q_H(-u, -v)}{2}. \end{aligned} \quad (27)$$

Therefore, the cross-correlation can be computed through the Hartley transform. Indicating with FHT the Fast Hartley Transform

$$r(x, y) = \text{FHT}^{-1} \left\{ \frac{P_H(u, v) \cdot Q_H(u, v) - P_H(u, v) \cdot Q_H(-u, -v)}{2} + \frac{P_H(-u, -v) \cdot Q_H(u, v) + P_H(-u, -v) \cdot Q_H(-u, -v)}{2} \right\} \quad (28)$$

$$= \text{FHT}^{-1} \{ R_H \}$$

Introducing a POF to reduce the noise effect gives

$$r(x, y) = \text{FHT}^{-1} \left\{ \frac{R_H(u, v)}{|R_H(u, v)|} \right\} \quad (29)$$

## RESULTS AND DISCUSSION

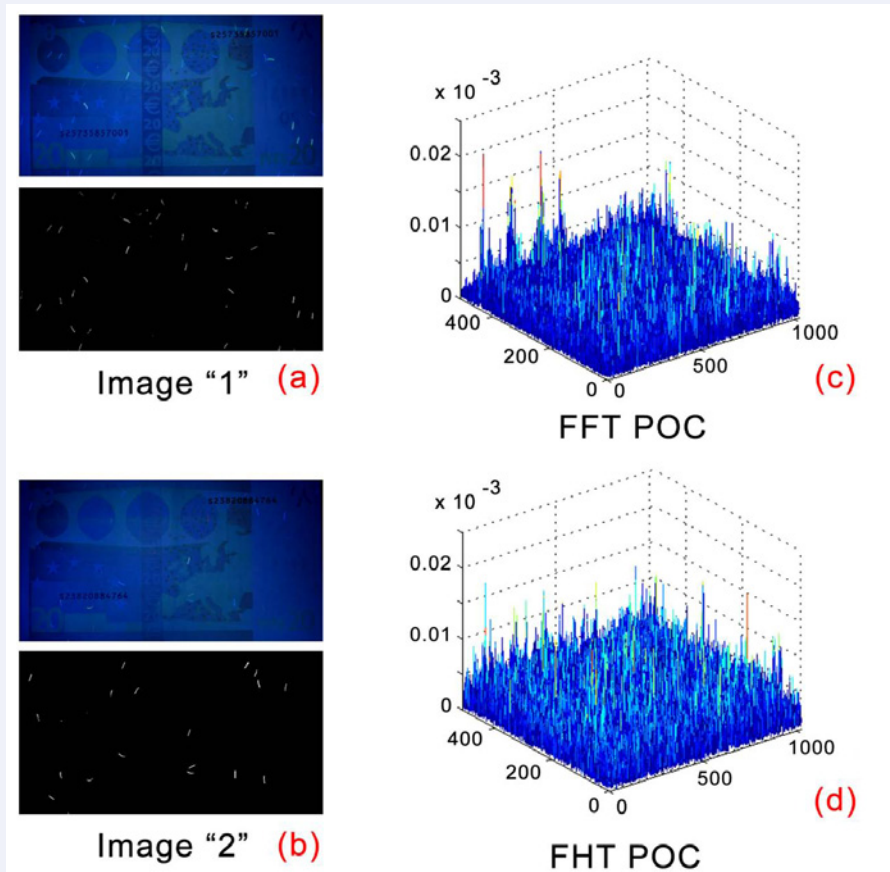
To test the proposed technique, a Fast Hartley Transform (FHT) was implemented by using algorithms reported in ref [7,22-24].

Furthermore, the application of Hartley Phase-Only Correlation (POC) function, in comparison with Fourier phase-only correlation, was tested.

Figure 3 shows the POC obtained by two speckle pattern captured during a PIV test.

In this example, the two images have size 512x512 pixels and on the second image was added artificial. The cross-correlation was computed in the Fourier space by Equation (24), and in the Hartley space by Equation (29). The FFT POC and the FHT POC are very similar; virtually identical.

Figure 4 shows the POC used in the banknote identification [25,26]. In this example, the two images have size 512x1024 pixels.



**Figure 4** Examples of POC function obtained from two different bank note.



**Table 1:** Main theorems for the Fourier and Hartley transform [9].

Theorem	$S(t)$	$S_F(f)$	$S_H(f)$
Similarity	$s(t/T)$	$ T S_F(Tf)$	$ T S_H(Tf)$
Addition	$s_1(t) + s_2(t)$	$S_{F1}(f) + S_{F2}(f)$	$S_{H1}(f) + S_{H2}(f)$
Reversal	$s(-t)$	$S_F(-f)$	$S_H(-f)$
Shift	$s(t-T)$	$e^{-i2\pi Tf} S_F(f)$	$\sin(2\pi Tf) S_H(-f) + \cos(2\pi Tf) S_H(f)$
Modulation	$s(t) \cdot \cos(2\pi f_0 t)$	$\frac{1}{2}[S_F(f-f_0) + S_F(f+f_0)]$	$\frac{1}{2}[S_H(f-f_0) + S_H(f+f_0)]$
Convolution	$s_1(t) * s_2(t)$	$S_{F1}(f) \cdot S_{F2}(f)$	$\frac{1}{2}[S_{H1}(f) \cdot S_{H2}(f) - S_{H1}(-f) \cdot S_{H2}(-f) + S_{H1}(f) \cdot S_{H2}(-f) + S_{H1}(-f) \cdot S_{H2}(f)]$
Autocorrelation	$s(t) * s(t)$	$ S_F(f) ^2$	$\frac{1}{2}[S_H^2(f) + S_H^2(-f)]$
Product	$s_1(t) \cdot s_2(t)$	$S_{F1}(f) * S_{F2}(f)$	$\frac{1}{2}[S_{H1}(f) * S_{H2}(f) - S_{H1}(-f) * S_{H2}(-f) + S_{H1}(f) * S_{H2}(-f) + S_{H1}(-f) * S_{H2}(f)]$
Derivative	$s'(t)$	$i2\pi f S_F(f)$	$-2\pi f S_H(-f)$
2nd derivative	$s''(t)$	$-4\pi^2 f^2 S_F(f)$	$-4\pi^2 f^2 S_H(f)$

The FFT and FHT give identical results, confirming the validity of the proposed theory.

By means of the tests presented in this work, and many others performed in the laboratory, we have found that the FHT requires a comparable number of steps to execute and is of comparable complexity to the FFT. The FHT does have the advantage that the forward and inverse transforms are the same, but this is only of advantage on a limited memory machine.

The FFT and FHT give identical results, confirming the validity of the proposed theory.

By means of the tests presented in this work, and many others performed in the laboratory, we have found that the FHT requires a comparable number of steps to execute and is of comparable complexity to the FFT. The FHT does have the advantage that the forward and inverse transforms are the same, but this is only of advantage on a limited memory machine.

## CONCLUSION

In this paper we have shown the correspondence between the Phase-Only Correlation (POC) function obtained by means of FFT and by FHT.

The Hartley transform is similar to the Fourier transform but it is free from the need to process complex numbers. Furthermore, the fast Hartley transform performs the transformation itself, the convolution and the cross-correlation with fewer additions and

multiplications than the fast Fourier transform. Using a FHT to compute cross-correlation, there is no information lost and the results are identical to those obtained by FFT. Unfortunately, the numbers of all other operations are the same; the differences in timings are negligible.

On the other hand, the Hartley transform does not use complex numbers and therefore needs less memory to store numbers than the Fourier transform.

The Hartley transform a valid alternative, particularly interesting when a greater efficiency in memory requirements is needed.

## REFERENCES

1. Duda RO, Hart PE. Pattern Classification and Scene Analysis, New York: Wiley. 1973; 482.
2. Gonzalez RC, Woods RE. Digital Image Processing. 3<sup>rd</sup> edn. Reading, Massachusetts: Addison-Wesley, 1992.
3. Wiener N. Extrapolation, Interpolation, and Smoothing of Stationary Time Series. M.I.T. Press. Cambridge, Massachusetts. 1949.
4. Stein JY. Digital Signal Processing: A Computer Science Perspective, New York: Wiley. 2000.
5. Horner JL, Gianino PD. Phase-only matched filtering. Applied Optics. 1984; 23: 812-816.
6. Chen QS, Defrise M, Deconinck F. Symmetric phase-only matched filtering of Fourier-Mellin transform for image registration and recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence. 1994; 16: 1156 – 1168.
7. Piccinin F. The Fast Hartley Transform as an alternative to the Fast Fourier Transform. Technical Memorandum SRL-000-TM. Department of Defence, Defence Science and Technology Organization, Salisbury, Surveillance Research Laboratory, South Australia. 1988.
8. Bracewell RN. Discrete Hartley transform. JOSA, 1983; 73: 1832-1835.
9. Bracewell RN. The Hartley transform. New York: Oxford University Press. 1986.
10. Lewis JP. Fast Template Matching. Vision Interface 95, Canadian Image Processing and Pattern Recognition Society, Quebec City, Canada; 1995: 120-123.
11. Kellera Y, Averbuch A. A projection-based extension to phase correlation image alignment. Signal Processing. 2007; 87: 124–133.
12. Ito K, Morita A, Aoki T, Higuchi T, Nakajima H, Kobayashi K. A Fingerprint Recognition Algorithm Combining Phase-Based Image Matching and Feature-Based Matching. Advances in Biometrics Lecture Notes in Computer Science. 2005; 3832: 316-325.
13. Miyazawa K, Ito K, Aoki T, Kobayashi K, Nakajima H. An Effective Approach for Iris Recognition Using Phase-Based Image Matching. IEEE Trans. on Pattern Analysis and Machine intelligence. 2008; 30: 1741-1756.
14. Ito K, Aoki T, Nakajima H, Kobayashi K, Higuchi T. A Palmprint Recognition Algorithm Using Phase-Based Image Matching. Proc. 13<sup>th</sup> IEEE Int'l Configuration Image Processing. 2006; 2669-2672.
15. Eckstein AC, Charonko J, Vlachos P. Phase correlation processing for DPIV measurements. Experiments in Fluids. 2008; 45: 485-500.
16. Thomas M, Misra S, Kambhamettu C, Kirby JT. A robust motion estimation algorithm for PIV. Meas. Sci. Technol. 2005; 16: 865-877.
17. Cozzella L, Simonetti C, Schirripa Spagnolo G. Drug packaging security by means of white-light speckle. Optics and Lasers in Engineering. 2012; 50: 1359–1371.
18. Schirripa Spagnolo G, Cozzella L, Papalillo D. Smartphone Sensors for Stone Lithography Authentication. Sensors. 2014; 1: 8217-8234.
19. Cozzella L, Schirripa Spagnolo G, Leccese F. Biometric-Like Approach for Verifying Artworks Authenticity. Applied Physics Research. 2013; 5: 118-130.
20. Hartley RVL. A More Symmetrical Fourier Analysis Applied to Transmission Problems. Proceedings of the IRE. 1942; 30: 144-150.
21. Sun H. The Hartley transform applied to particle image velocimetry. Meas. Sci Technol. 2002; 13: 1996-2000.
22. Lee DJ, Ramirez M, Sunanda Mitra S. Fast 2-D Hartley transform in 3-D object representation and recognition. Proc. SPIE 1608, Intelligent Robots and Computer Vision X: Neural, Biological, and 3-D Methods. 1992; 302.
23. Meckelburg HJ, Lipka D. Fast Hartley transform algorithm. Electronics Letters. 1985; 21: 341-343.
24. Popovic M, Sevic D. A new look at the comparison of the fast Hartley and Fourier transforms. IEEE Transactions on Signal Processing. 1994; 42: 2178 – 2182.
25. Schirripa Spagnolo G, Cozzella L, Simonetti C. Banknote security using a biometric-like technique: a hylemetric approach. Meas Sci Technol. 2010; 21: 055501.
26. Schirripa Spagnolo G, Cozzella L, Simonetti C. Currency verification by a 2D infrared barcode. Meas. Sci. Technol. 2010; 21: 107002.

### Cite this article

Cozzella L, Spagnolo GS (2014) Phase-only Correlation Function by Means of Hartley Transform. JSM Math Stat 1(1): 1004.