

# A FAST ALGORITHM FOR CALCULATING THE DUAL GABOR WINDOW WITH INTEGER OVERSAMPLING

*Nagesh K. Subbanna and Yonina C. Eldar*

Department of Electrical Engineering  
Technion - Israel Institute of Technology  
Haifa, Israel - 32000

## ABSTRACT

**In this paper, we consider Gabor expansion of signals that are oversampled in the time-frequency domain by an integer factor. We demonstrate that using a combination of non minimum norm dual windows and discrete Fourier transforms (DFT), we can obtain the Gabor expansion very efficiently. The conventional method used to solve this problem is compared with our method and the advantages of our method are explicitly shown.**

## 1. INTRODUCTION

It is often necessary to represent signals in the combined time-frequency domain as many applications in the fields of pattern classification, computer vision, and texture segmentation, among others, require localization of frequencies in time (or space). Representation in the time-frequency domain can be accomplished by many techniques, among which the Gabor representation is one of the most popular. Important advantages of the Gabor representation include uniform resolution and optimal resolution in the time-frequency domain with the Gaussian window function.

We consider a discrete time, finite energy signal  $f[k] \in C^L$ , with Gabor coefficients  $c_{m,n}$  given by:

$$c_{m,n} = \sum_{k=0}^{L-1} f[k]g^*[k-na]e^{-j2\pi mbk/L}, \quad (1)$$

where  $a$  and  $b$  are the shifts along the time and frequency axes respectively, and  $g[k]$  is the window function used. The signal  $f[k]$  is reconstructed from the coefficients  $c_{m,n}$  by [1]

$$f[k] = \sum_{m=0}^{\bar{b}-1} \sum_{n=0}^{\bar{a}-1} c_{m,n}\gamma_{m,n}[k], \quad (2)$$

where  $\gamma_{m,n}[k] = \gamma[k-na]e^{j2\pi mbk/L}$  is a dual Gabor window,  $\bar{a} = L/a \in N$  and  $\bar{b} = L/b \in N$  are the number of shifts along the time and frequency axes respectively and  $N$

denotes the set of natural numbers. The task is to efficiently compute a dual Gabor window  $\gamma[k]$ .

In this paper, we consider the case in which the representation is overcomplete, so that the vectors  $g_{m,n}[k] = g[k-na]e^{j2\pi mbk/L}$  are linearly dependent. This case corresponds to the condition  $ab < L$  [1]. We further assume integer oversampling, meaning that  $L$  is divisible by  $ab$ , which is the choice used most often in practice [3], [2], [4]. In the case of oversampling, the expansion is not unique so that there are infinite choices available for the dual Gabor window  $\gamma[k]$ . The most popular (and traditional) choice is the minimum norm dual window [3] of the window function  $g[k]$  used. In a previous paper [7], we discussed the problems of computational complexity and instability often seen in the computation of the minimum norm dual window and introduced non minimum norm dual windows. In this paper, we propose a fast method to compute the dual window  $\gamma[k]$  based on discrete Fourier transforms (DFTs) and non minimum norm dual windows for integer oversampling. We also derive an easily verifiable condition for the existence of the non minimum norm dual window for integer oversampling. We analyze the complexity of computing both the minimum norm dual window and the non minimum norm dual window and show that our method is much faster.

The paper is organized as follows: in Section 2, we focus on the formulation of the problem, briefly introduce the standard method to solve this problem and consider non minimal duals. In Section 3, we mention some properties of the non minimal dual of the type introduced in Section 2 and use them to develop existence conditions for the non minimum dual. In Section 4, we present a DFT based technique for computing the dual that is faster than the traditional techniques. Some examples are considered in Section 5.

## 2. NON MINIMUM NORM DUALS

Consider the expansion of a discrete time, finite energy signal  $f[k]$  of length  $L$ . Given a set of Gabor vectors  $g_{m,n}[k] = g[k-na]e^{j2\pi mbk/L}$ ,  $k = 0, 1, \dots, L-1$  that constitute an overcomplete basis for  $C^L$ , our problem is to reconstruct

the signal from the coefficients (1). In matrix form, the coefficients can be written as

$$c = G^* f, \quad (3)$$

where  $f = f[k]$  is a vector of length  $L$  with elements  $f[k]$ ,  $c$  is a vector of length  $\bar{a}\bar{b}$  with elements  $\{c_{m,n}\}$ ,  $G$  is a matrix of size  $L \times \bar{a}\bar{b}$ , given by

$$G = \begin{bmatrix} g_{0,0}[0] & \cdots & g_{\bar{a}-1,\bar{b}-1}[0] \\ g_{0,0}[1] & \cdots & g_{\bar{a}-1,\bar{b}-1}[1] \\ \vdots & \vdots & \vdots \\ g_{0,0}[L-1] & \cdots & g_{\bar{a}-1,\bar{b}-1}[L-1] \end{bmatrix}, \quad (4)$$

and  $(\cdot)^*$  denotes the conjugate transpose of the corresponding matrix. We reconstruct the signal  $f[k]$  from the coefficients  $c$  using (2), where the dual Gabor matrix satisfies the equation

$$\Gamma G^* = I. \quad (5)$$

Here  $\Gamma$  is the matrix with columns  $\gamma_{m,n}[k] = \gamma[k - na]e^{j2\pi mbk/L}$ . The traditional choice is the minimum norm dual  $\tilde{\Gamma}$  given by

$$\tilde{\Gamma} = G^{*\dagger} = (GG^*)^{-1}G, \quad (6)$$

where  $G^{*\dagger}$  denotes the minimum norm pseudoinverse of the matrix  $G^*$  and  $\tilde{\Gamma}$  is a matrix with columns  $\tilde{\gamma}_{m,n}[k] = \tilde{\gamma}[k - na]e^{j2\pi mbk/L}$ . Since the vectors  $g_{m,n}[k]$  span the space  $C^L$ , the matrix  $GG^*$  is always invertible. The standard method to compute the minimum norm pseudoinverse of  $G^*$  for integer oversampling uses the fact that the matrix  $GG^*$  is unitarily diagonalizable by the unitary matrix  $F_B$  [4], which is given by

$$F_B = F \otimes I_a. \quad (7)$$

Here  $I_a$  is an identity matrix of size  $a \times a$ ,  $\otimes$  denotes the Kronecker product, and  $F$  is a Fourier matrix of size  $\bar{a} \times \bar{a}$ , given by

$$F = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & e^{-\frac{j2\pi}{\bar{a}}} & \cdots & e^{-\frac{j2\pi(\bar{a}-1)}{\bar{a}}} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & e^{-\frac{j2\pi(\bar{a}-1)}{\bar{a}}} & \cdots & e^{-\frac{j2\pi(\bar{a}-1)(\bar{a}-1)}{\bar{a}}} \end{bmatrix}. \quad (8)$$

Therefore we have that

$$F_B S F_B^* = R, \quad (9)$$

where  $R$  is a diagonal matrix,  $S = GG^*$  and  $F_B$  is defined by (7). Using (9), the computation of the inverse of the matrix  $S$  is straightforward; it is given by

$$S^{-1} = F_B^* R^{-1} F_B. \quad (10)$$

The minimum norm dual window is then computed as [3], [4]

$$\tilde{\gamma} = F_B^* R^{-1} F_B g, \quad (11)$$

where  $\tilde{\gamma} = \tilde{\gamma}[k]$  and  $g = g[k]$ ,  $k = 0, \dots, L-1$  are vectors of length  $L$ . The matrix  $\tilde{\Gamma}$  is then obtained from the dual window function  $\tilde{\gamma}[k]$  by shifting and modulating  $\tilde{\gamma}[k]$ .

The computation of the matrix  $GG^*$  is often computationally demanding [7]. Therefore, to obtain a computationally more efficient algorithm, we propose a dual of the form  $\{d_{m,n} = d[k - na]e^{j2\pi mbk/L}\}$  whose vectors span  $C^L$  and are columns of the matrix

$$D = (HG^*)^{-1}H, \quad (12)$$

where  $H$  is a matrix that corresponds to an overcomplete basis for  $C^L$ , with vectors  $h_{m,n}[k] = h[k - na]e^{j2\pi mbk/L}$  and has the same dimensions as  $G$ . As we show below and in [7], with a correct choice of  $H$ , the problems of computational difficulty and stability can be alleviated. Note that, the minimum norm dual  $(GG^*)^{-1}G$  is a special case of  $(HG^*)^{-1}H$ , where  $H = G$ .

### 3. EXISTENCE CONDITIONS FOR THE NON MINIMUM NORM DUAL

In this section, we develop an easily verifiable condition under which a non minimum norm dual of the form (12) exists, when  $L$  is divisible by  $ab$ . To this end, we first note that the matrix  $P = HG^*$  has a block circulant structure for any choice of  $H$  [3], [5]. In particular,  $P = \mathcal{C}(A_0, \dots, A_{\bar{a}-1})$  where  $A_p, p \in 0, \dots, \bar{a}-1$  is an  $a \times a$  matrix, and  $\mathcal{C}$  stands for rotation of the blocks  $A_p$ , so that  $P$  has the following structure:

$$P = \begin{bmatrix} A_0 & A_1 & \cdots & A_{\bar{a}-1} \\ A_{\bar{a}-1} & A_0 & \cdots & A_{\bar{a}-2} \\ \vdots & \vdots & \cdots & \vdots \\ A_1 & A_2 & \cdots & A_0 \end{bmatrix}. \quad (13)$$

In general (when  $L$  is not divisible by  $ab$ ), the matrices  $A_p, p \in 0, \dots, \bar{a}-1$  have no particular structure. However, when  $L$  is divisible by  $ab$ , the matrices  $A_p, p \in 0, \dots, \bar{a}-1$  are either diagonal matrices or null matrices. More specifically, the matrices  $A_p, p \in 0, L/ab, \dots, (b-1)L/ab$  are diagonal matrices and the other matrices are null matrices [5].

In our previous paper [7], we showed that if  $P$  is invertible, then a dual of the form (12) exists. When  $L$  is divisible by  $ab$ , we have an easily verifiable condition for the invertibility of  $P$  as shown below in Theorem 1. The condition for invertibility of  $P$  relies on the definition of the DFT of a block circulant matrix which is given by

$\mathcal{F}(P) = \mathcal{C}(\tilde{A}_0, \dots, \tilde{A}_{\bar{a}-1})$  where

$$\tilde{A}_s = \sum_{r=0}^{\bar{a}-1} e^{-j2\pi rs/\bar{a}} A_r, s \in 0, \dots, \bar{a}-1. \quad (14)$$

**Theorem 1** The matrix  $P = \mathcal{C}(A_0, A_1, \dots, A_{\bar{a}-1})$  is invertible if and only if there is no zero along the principal diagonal and every  $k$ -th subdiagonal of the DFT of  $P$ , where  $k = \pm a, \pm 2a, \dots, \pm(\bar{a}-1)a$ .

**Proof:** We calculate the DFT of  $P$  according to (14). However, from [5], we know that only  $A_p, p \in 0, L/ab, \dots, (b-1)L/ab$  are diagonal matrices and the others are null matrices. Therefore, the calculation of the DFT of  $P$  as shown in (14), reduces to

$$\tilde{A}_s = A_0 + e^{-j2\pi s/b} A_{\frac{L}{ab}} + \dots + e^{-j2\pi(b-1)s/b} A_{\frac{L}{ab}(b-1)}. \quad (15)$$

From (15), we see that the Fourier components  $\tilde{A}_s, s \in 0, \dots, \bar{a}-1$  are also diagonal.

From [6], we know that matrix  $P$  is invertible if and only if all Fourier components  $\tilde{A}_s, s \in 0, 1, \dots, \bar{a}-1$  are invertible. As the Fourier components  $\tilde{A}_s$  are diagonal matrices they are invertible if and only if there is no zero along their diagonal. Since  $\mathcal{F}(P)$  is also a block circulant matrix where each block  $\tilde{A}_s$  is of size  $a \times a$ , and each matrix  $\tilde{A}_s$  is a diagonal matrix, we can see  $\mathcal{F}(P)$  has its non zero entries only on the principal diagonal and every  $k$ -th subdiagonal where  $k = \pm a, \pm 2a, \dots, \pm(\bar{a}-1)a$ . If there is no zero along the diagonals of  $\tilde{A}_s, s \in 0, \dots, \bar{a}-1$ , there will be no zero along the diagonal or  $k$ -th subdiagonal either. Thus the statement is proved.  $\square$

The dual window function of  $d[k], k = 0, \dots, L-1$  is calculated by

$$d = P^{-1}h, \quad (16)$$

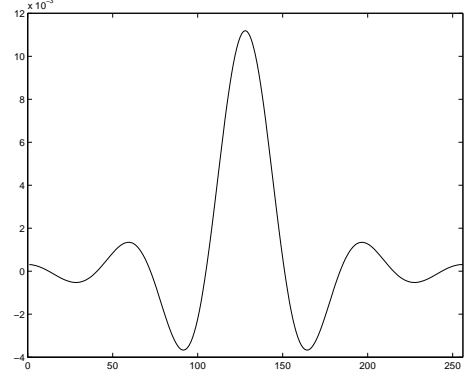
where  $d = d[k]$  and  $h = h[k], k \in 0, \dots, L-1$  are both vectors of length  $L$ . Therefore, computing the inverse of  $P$  gives us the dual window function. The inverse of  $P$  is given by  $P^{-1} = \mathcal{C}(B_0, B_1, \dots, B_{\bar{a}-1})$  where

$$B_r = \frac{1}{\bar{a}} \sum_{s=0}^{\bar{a}-1} e^{j2\pi rs/\bar{a}} \tilde{A}_s^{-1}, 0 \leq r \leq \bar{a}-1. \quad (17)$$

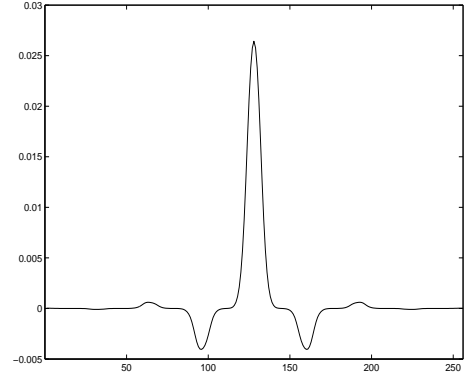
Here  $\tilde{A}_s$  are the Fourier components generated as shown in (15).

#### 4. COMPUTATIONAL ISSUES

In many practical applications, the window function  $g[k]$  is chosen to be a Gaussian function given by  $g[k] = e^{-k^2/\sigma_1^2}, \sigma_1 > 0$ , where  $\sigma_1$  is the spread of the window function. We may choose  $h[k]$  also to be a Gaussian of the form  $h[k] = e^{-k^2/\sigma_2^2}, \sigma_2 > 0$ . We select  $\sigma_2$  such that the



**Fig. 1.** The minimum dual window for lattice constants  $a = 8, b = 8, \sigma_1 = \sigma_2 = 16$ , and  $L = 256$ .



**Fig. 2.** The non minimum dual window for lattice constants  $a=8, b=8, \sigma_1 = 16, \sigma_2 = 2$ , and  $L = 256$ .

effective spread of  $h[k]$  is  $a$ . Outside this effective spread, the value of the window is negligibly small (less than a thousandth of the value of the window at the peak). The effective spread of the window  $h[k]$  is from  $-a/2$  to  $a/2$ . Therefore, the matrix  $H$  is made block diagonal for all practical purposes and looks as shown below.

$$H = \begin{bmatrix} H_0 & \varepsilon & \cdots & \varepsilon \\ \varepsilon & H_1 & \cdots & \varepsilon \\ \vdots & \vdots & \cdots & \vdots \\ \varepsilon & \varepsilon & \cdots & H_{\bar{a}-1} \end{bmatrix} \quad (18)$$

The  $\varepsilon$  is used to denote that the value of the elements in these blocks is extremely small. Each block  $H_v, v \in 0, \dots, \bar{a}-1$  and  $\varepsilon$  is of size  $a \times \bar{b}$ .

In the creation of the matrix  $P$  it can be seen that, because we limited the spread of  $h[k]$ , the elements of  $P$  are generated by  $P_{k, k+q\bar{b}} = \bar{b} e^{-(k-na)^2/\sigma_2^2} e^{-(k+q\bar{b}-na)/\sigma_1^2}, q \in 0, \dots, b-1$  and  $k-na \in 0, 1, \dots, a$  and 0 otherwise.

**Theorem 2:** [7] If  $g[k] = e^{-k^2/\sigma_1^2}, 0 < \sigma_1 < \infty$  and

$h[k] = e^{-k^2/\sigma_2^2}$ ,  $0 < \sigma_2 < \infty$  and  $L$  is divisible by  $ab$ , then the matrix  $P$  is invertible.  $\square$

The theorem allows us to use all Gaussian functions in our implementations of the algorithm when  $L$  is divisible by  $ab$ . The proof of the theorem follows directly from Theorem 1.

#### 4.1. Algorithm:

The algorithm to compute the dual window when  $L$  is divisible by  $ab$  is summarized as follows:

1. Create the matrix  $P$  choosing the window function  $h[k]$  with spread  $\sigma_2$  such that the matrix  $H$  is of the form (18).
2. Compute the DFT for each block  $A_k$ ,  $k \in 0, \dots, \bar{a}-1$  of  $P$  using (15).
3. Invert the individual diagonal matrices and recombine them using (17) to obtain  $P^{-1}$ .
4. Multiply  $P^{-1}$  with the vector  $h$  to obtain the dual window function  $d$ .

It is easy to obtain the matrix  $D = (HG^*)^{-1}H$  using  $d[k]$  by shifting and modulating the dual window function  $d[k]$ .

#### 4.2. Computational Complexity

The computational complexity of the traditional algorithm [3], [4] to compute the minimum norm dual window is given below. The number of operations necessary to generate the matrix  $S$  is  $(\bar{a} + 1)ab$  multiplications and  $ba\bar{a}$  additions [3]. The computational complexity of unitarily diagonalizing and inverting  $S$  is  $O(2L\log(\bar{a}))$ [4]. Therefore, the overall complexity of the algorithm is  $O(L\log(\bar{a})) + ab(\bar{a} + 1)$ .

Usually  $\bar{a}$  is very large compared to  $a$  since the shift needs to be small enough to capture all the details of the signal accurately. Therefore, anything that removes  $\bar{a}$  component from the multiplications will simplify the procedure.

In our case, the number of operations necessary to generate the matrix  $P$  is  $2ab$  and the number of additions is 0. The computational complexity for computing the Fourier components and inverting  $P$  is given by  $O(2L\log(b))$ . The overall computational complexity of our algorithm is given by  $O(2L\log(b)) + 2ab$ . Since  $b < \bar{a}$  and  $2ab < (\bar{a} + 1)ab$ , our technique is much faster than the traditional algorithm.

### 5. EXAMPLES

We now consider examples of two possible dual windows of a typical Gaussian Window with  $\sigma_1 = 16$ . Figure 1 shows a typical minimum norm dual window with  $\sigma_1 = \sigma_2 = 16$ . The lattice constants  $(a, b)$  were both set to 8 and the length of the signal was 256. Figure 2 shows the non minimum

dual with  $\sigma_2 = 4$ . Since  $a$  is 8 (quadruple the value of  $\sigma_2$ ), the values of  $h[k]$  outside twice the spread can be neglected. As can be seen in Figure 2, even the non minimal dual window gives a good localization.

The number of operations necessary to compute the minimum norm dual is roughly 8 times the number of operations required to calculate the non minimum norm dual shown in Figure 2.

It has been observed that when  $\sigma_1$  is very different from the shift  $a$ , the determinant of the matrix  $S = GG^*$  is very close to zero (though it is never equal to zero) [7]. In pattern recognition applications, frequently there arise situations where  $\sigma_1$  has to be very different from the shift  $a$ . In such a case inverting  $S$  is computationally expensive and requires very high precision arithmetic. To alleviate this problem, a proper choice of  $\sigma_2$ , which is different from  $\sigma_1$  is necessary, and which makes the determinant of  $P$  much larger than the determinant of  $S$ . With our method it is possible to find a function  $h[k]$  which can make this problem more tractable.

To conclude, advantages of our method include a faster computational rate, as was shown above, better flexibility as shown by freedom to determine the form of  $h[k]$  quite independent of  $g[k]$  and freedom to optimize the localization with respect to the number of operations and stability based on the end user's requirement and resources available.

### 6. REFERENCES

- [1] J. Wexler and S. Raz, "Discrete Gabor Expansions", *Signal Processing*, 21(3):pp 207-221, November 1990.
- [2] M. Zibulski and Y. Y. Zeevi, "Oversampling in the Gabor scheme", *IEEE Transactions on Signal Processing*, 41(8): pp 2679-2687, August 1993.
- [3] S. Qiu and H. Feichtinger, "Discrete Gabor Structures and Optimal Representation", *IEEE Transactions on Signal Processing*, 43(10):pp 2258-2268, October 1995.
- [4] S. Qiu, "Block circulant Gabor-matrix structure and discrete Gabor transforms", *Optical Engineering*, Vol. 34, pp 2872-2878, October 1995
- [5] N. K. Subbanna and Y. C. Eldar, "Efficient algorithms to compute dual Gabor windows", in preparation.
- [6] A. Beck, Y. C. Eldar, and A. Ben Tal, "Minmax Mean Squared Error Estimation of multichannel signals", preprint
- [7] N. K. Subbanna and Y. C. Eldar, "Efficient Gabor expansion using Non Minimal Dual Gabor Windows", submitted to ICECS 2004.