Gaussian Mixture Bandpass Filter Design for Narrow Passband Width by Using a FIR Recursive Filter

Yukihiko Yamashita *and Toru Wakahara †

March 31, 2022

Abstract

Bandpass filters (BPFs) are very important to extract target signals and eliminate noise from the received signals. In this research, we propose a concept of Gaussian mixture BPF (GMBPF) of which impulse response is symmetric. It can be approximately realized using finite impulse response (FIR) recursive filters; hence, its calculation complexity does not depend on the length of the impulse response. The property makes GMBPF ideal for narrow bandpass filtering applications. We conducted experiments to demonstrate the advantages of the proposed GMBPF over FIR filters designed by a MATLAB function with regard to the computational complexity.

***Keywords*—** Gaussian mixture bandpass filter, sliding Fourier transform, attenuated sliding Fourier transform, FIR recursive filter

1 Introduction

Bandpass filters (BPFs) are widely used in signal processing and image processing applications to extract target signals and eliminate noise from the received signals. In digital signal processing, they can be realized by a finite impulse response (FIR) filter or a recursive filter of an infinite impulse response (IIR) by approximating the impulse response given by the inverse discrete time Fourier transform or by the approximate conversion from continuous-time filters such as the Butterworth filter and the Chebyshev filter [1, 2, 3, 4, 5, 6]. Although BPFs with narrow passband width are important to extract the carrier signal from the received signal in communication application, the problem is that the length of the impulse response becomes long and number of taps of the FIR filter becomes large. This results in high computational complexity in narrow passband BPFs designed using a FIR filter. By using IIR filters, this problem can be reduced to some extent. However, if a steeper frequency roll-off characteristics is required, the order of the filter becomes high and its computational complexity increases. Furthermore, IIR filters have another problem that their impulse response is not symmetric and they do not have a linear phase.

Recently, methods to calculate Gaussian smoothing using the sliding Fourier transform (SFT) are extensively studied because computational complexity of SFT does not depend on smoothing scale [7]. Elboher et al. [8] proposed using a kernel integral as SFT to calculate Gaussian smoothing. Sugimoto et al. [9, 10, 11, 12] conducted in-depth studies on calculating the Gaussian smoothing by sliding discrete cosine transform. Since the absolute value of all the poles of the transfer function of SFT is 1, calculation errors are accumulated and the output is likely to diverge. To solve the problem, Yamashita et al. [13, 14] proposed to use the attenuated SFT (ASFT) for Gaussian smoothing. They also applied them to Morlet wavelet transform [15].

In this study, we propose to use SFT/ASFT to design a BPF, considering a function given by the product of Gaussian function in discrete time and $\cos \omega n$. Because the Fourier transform of a Gaussian function is a Gaussian function, the Fourier transform of this product is going to be the sum of two Gaussian functions with translations of $\pm\omega$ on the frequency axis. The frequency characteristics of BPF of which impulse response is given by the product is a Gaussian function. This is the same as the Gabor transform [16] and the BPF is easily extended by constructing a filter of which impulse response is a sum of such products. It becomes a BPF of which frequency characteristics is a sum of Gaussian functions. We call it the Gaussian mixture BPF (GMBPF). We propose to implement an approximate GMBPF given by the sum of several frequency components of SFT/ASFT. Furthermore, each frequency component of SFT/ASFT can be calculated using the first order recursive filter with subtraction at the end of its impulse response. Hence, even if the passband width of GMBPF becomes narrow, the calculation steps are not changed although the number of taps of the FIR filter realized by the recursive filter

*Corresponding author, Yukihiko Yamashita is with the School of Environment and Society, Tokyo Institute of Technology, Tokyo, 152–8552, Japan email: yamasita@tse.ens.titech.ac.jp.

†Toru Wakahara is with the Faculty of Computer and Information Sciences, Hosei University, Tokyo, 184–8584, Japan, email: wakahara@hosei.ac.jp.

increases. Furthermore, a symmetric impulse response is approximated by the proposed GMBPF, and it has an approximated property of a linear phase. Capizzi et al. [17, 18] proposed a filter of which frequency characteristics is a sum of Gaussian functions. However, the filter is realized by its truncated impulse response of the filter so that its computational complexity increases is inversely proportional to the passband width.

We finally perform the experiments to to showcase that GMBPF can realize a BPF of a narrow passband width by comparing its transfer function with that of FIR filters, highlight the advantage of ASFT over SFT by comparing their floating-point calculation errors, and evaluate the computation time of GMBPFs with FIR filters.

The remainder of this paper is organized as follows. In Section 2, we summarize the calculation methods of Gaussian smoothing using SFT/ASFT. In Section 3, we propose the GMBPF and discuss its calculation method. In Section 4, we perform experiments to show the advantages of the proposed algorithm. Section 5 concludes the paper.

2 Gaussian smoothing by FIR recursive filter

Let $x[n]$ and σ be an input signal defined in $[0, N-1]$ with an integer N and a standard deviation for the Gaussian smoothing. Then, the Gaussian function and the Gaussian smoothing are expressed by

$$G[n] = \sqrt{\frac{\gamma}{\pi}} e^{-\gamma n^2}, \tag{1}$$

$$x_{\mathrm{G}}[n] \equiv \sum_{k=-K}^{K} G[k]x[n-k], \tag{2}$$

where $\gamma = 1/(2\sigma^2)$. For the calculation, we assume that values of $x[n]$ are extended properly even if $n < 0$ or $n \geq N$. Usually, they are either zero or the values on the edges of the interval. Because $G[n]$ decays rapidly, the summation is calculated in the interval $[-K, K]$ for an integer K. The computational complexity of Eq.(2) is $O(KN)$. A larger σ requires larger K and more calculation steps.

2.1 Gaussian smoothing by SFT

Let $\beta = \pi/K$ and $c_p[n]$ and $s_p[n]$ be the p-th order components of the SFT of the input signal $x[n]$ in the interval $[-K, K]$. They are given by

$$c_p[n] + is_p[n] = \sum_{k=-K}^{K} x[n-k]e^{i\beta pk}. \tag{3}$$

As shown in the next subsection, $c_p[n]$ and $s_p[n]$ can be obtained easily by using a first-order recursive FIR filter. Note that the bases in this SFT are not orthogonal.

We define an approximation of the Gaussian function ($-K \leq k \leq K$) with Fourier series by

$$G[k] \simeq \sum_{p=0}^{P} a_p \cos(\beta pk). \tag{4}$$

A set of coefficients a_p can be decided by minimizing a criterion based on the mean square error (MSE). Then, the smoothed signal is given by

$$x_{\mathrm{G}}[n] \simeq \sum_{p=0}^{P} a_p c_p[n]. \tag{5}$$

As discussed next subsection, each component of the SFT can be calculated with a computational complexity of $O(N)$, whereas the complexity of smoothing is of the order of $O(PN)$. By setting the value of P from 2 to 6, we can substantially decrease the calculation steps as compared with calculating Eq.(2) directly.

2.2 Gaussian smoothing using ASFT

Although SFT can be realized by a FIR recursive filter, the error is accumulated because $|e^{i\beta pk}| = 1$. It causes a problem of calculation error especially when we use a single-precision floating-point unit. In order to reduce the accumulated errors, Gaussian smoothing by ASFT was proposed.

Let $\alpha(> 0)$, β be a decay constant and π/K, respectively, and let $\tilde{c}_p[n]$ and $\tilde{s}_p[n]$ be the p-th order components of ASFT of interval $[-K, K]$ for the input signal $x[n]$. They are given by

$$\tilde{c}_p[n] + i\tilde{s}_p[n] = \sum_{k=-K}^{K} x[n-k]e^{\alpha k + i\beta pk}. \tag{6}$$

Since we have

$$e^{\alpha n}e^{-\gamma n^2} = e^{\frac{\alpha^2}{4\gamma}}e^{-\gamma\left(n-\frac{\alpha}{2\gamma}\right)^2}, \tag{7}$$

the attenuation can be converted to translation in the time domain. Choosing α such that $n_0 = \frac{\alpha}{2\gamma}$ becomes an integer and shift the output, we can obtain the true Gaussian smoothed signal because we have

$$G[n-n_0] = e^{-\frac{\alpha^2}{4\gamma}}e^{\alpha n}G[n]. \tag{8}$$

Eq.(4) yields

$$\sum_{k=-K}^{K} G[k]\left(e^{\alpha k}x[n-k]\right) \simeq \sum_{p=0}^{P} a_p\tilde{c}_p[n]. \tag{9}$$

Then, we have

$$\sum_{k=-K}^{K} G[k-n_0]x[n-k] \simeq e^{-\frac{\alpha^2}{4\gamma}}\sum_{p=0}^{P} a_p\tilde{c}_p[n]. \tag{10}$$

We assume that n_0 is small compared to σ. Because the values of $G[n]$ for $|n| \geq \sigma$ decrease rapidly, we have

$$x_{\mathrm{G}}[n] \simeq e^{-\frac{\alpha^2}{4\gamma}}\sum_{p=0}^{P} a_p\tilde{c}_p[n+n_0]. \tag{11}$$

Next, we explain a method to calculate the ASFT of the input signal $x[n]$. We define a first-order FIR recursive filter of order $2K$ by

$$\tilde{v}_{(2K)}[n] = e^{-\alpha-i\beta p}\tilde{v}_{(2K)}[n-1] + x[n] - e^{-2\alpha K}x[n-2K]. \tag{12}$$

By using its output $\tilde{v}_{(2K)}[n]$, the ASFT can be calculated by

$$\begin{aligned}&\tilde{c}_p[n] - i\tilde{s}_p[n]\\ =&(-1)^p e^{-\alpha K}(\tilde{v}_{(2K)}[n+K] + e^{-2\alpha K}x[n-K])).\end{aligned} \tag{13}$$

When $\alpha > 0$, by the attenuation factor $e^{-\alpha}$ in eq.(12), $\tilde{v}[n]$ is bounded for bounded calculation error in a step. As a result, the calculation of the ASFT is stabilized even using the single-precision floating-point calculations. Assuming $\alpha = 0$ and $n_0 = 0$ in Eq.(12), we can obtain a formula for SFT.

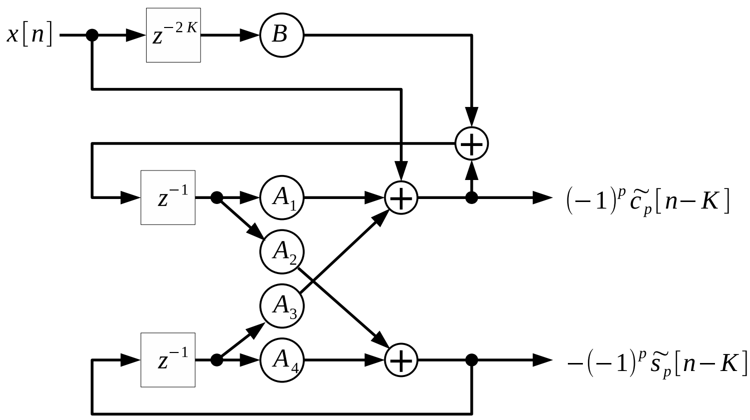


Figure 1: Recursive filter for a component of the ASFT ($A_1 = A_4 = e^{\alpha}\cos\beta$, $-A_2 = A_3 = e^{\alpha}\sin\beta$, $B = -e^{2\alpha K}$)

Fig. 1 shows the structure of a FIR recursive filter to calculate a component of the ASFT. It can be seen from Fig. 1 that five multiplications and three additions are necessary for a component at each point.

3 GMBPF by IIR structured FIR filter

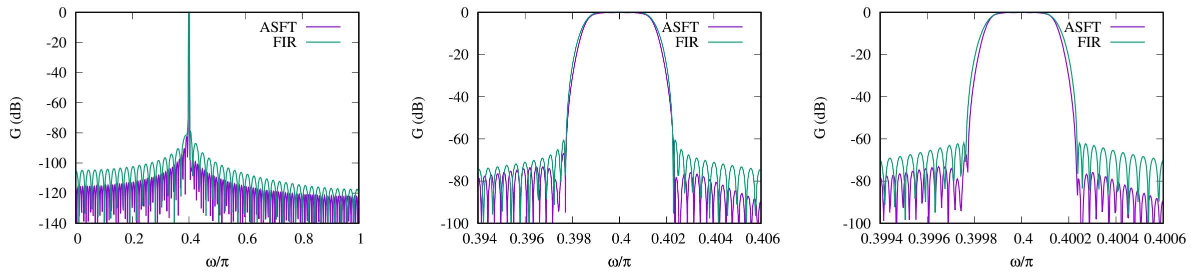
3.1 Definition of GMBPF

Assuming $\{\omega_l\}_{l=0}^{L-1}$ be a set of angular frequencies, the impulse response of GMBPF is given as

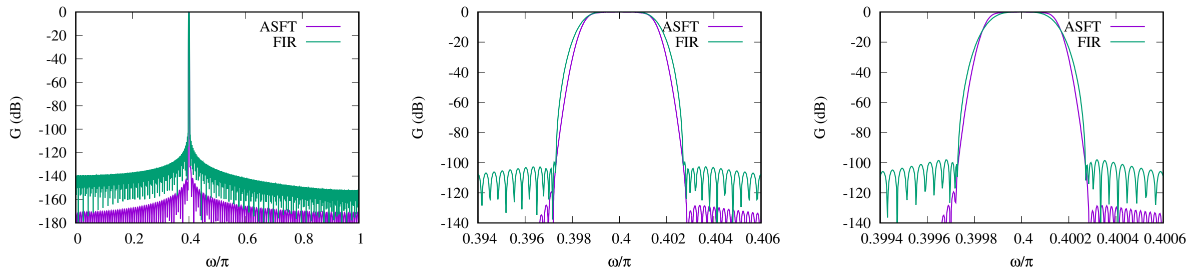
$$h[n] = Ae^{-\gamma n^2}\sum_{l=0}^{L-1}\cos\omega_l n, \tag{14}$$

Table 1: Cutoff and stop band widths (rad/s) of GMBPF.

P	σ	Cutoff	Stop (attenuation)
16	100	0.0271π	0.0455π (-60 dB)
	1000	0.00271π	0.00455π (-60 dB)
	10000	0.000271π	0.000455π (-60 dB)
27	100	0.0271π	0.0471π (-100 dB)
	1000	0.00271π	0.00471π (-100 dB)
	10000	0.000271π	0.000471π (-100 dB)

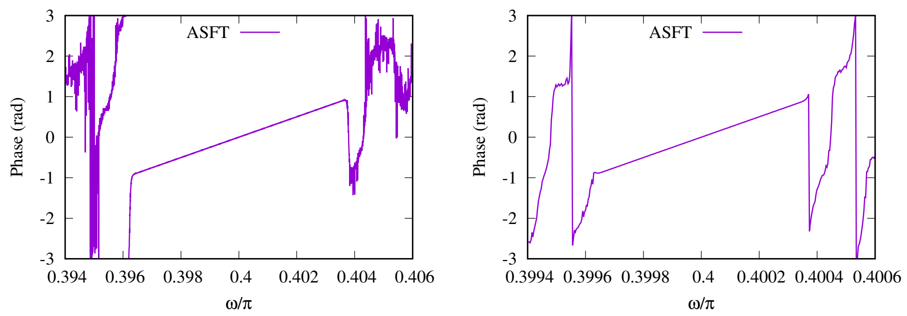


(a) $P = 16$, $\sigma = 1000$ (c) $P = 16$, $\sigma = 1000$ (magnified) (d) $P = 16$, $\sigma = 10000$ (magnified)



(a) $P = 27$, $\sigma = 1000$ (c) $P = 27$, $\sigma = 1000$ (magnified) (d) $P = 27$, $\sigma = 10000$ (magnified)

Figure 2: Amplitudes of the transfer functions of GMBPF with ASFT and FIR for different values of P and σ.



(a) $P = 16$, $\sigma = 1000$ (b) $P = 16$, $\sigma = 10000$

Figure 3: Phase of the transfer functions of GMBPF with ASFT

where A is a constant to normalize the impulse response. The domain of $h[n]$ is the set of all integers, that is not restricted between $-K$ and K. Because the Fourier transform of a Gaussian function is a Gaussian function, the Fourier transform of $h[n]$ can be considered a Gaussian mixture function. We can decide ω_l by

$$\omega_{q+Q} = \omega_C + \omega_\Delta q, \tag{15}$$

for an integer $Q(\geq 0)$, $q = 0, \pm 1, \pm 2, \ldots, \pm Q$, where $L = 2Q + 1$, ω_C is the center of passband, and ω_Δ is a step between Gaussian functions in frequency domain. For an input $x[n]$ and an integer M, GMBPF is defined as a filter using $h[n]$ that outputs the following $y_M[n]$.

$$y_M[n] = \sum_{k=-M}^{M} h[k]x[n-k]. \tag{16}$$

Larger the M is, more precise is the approximation of the GMBPF according to Eq.(16).

4 Realization of GMBPF by FIR recursive filter

We approximate $h[k - n_0]$ by $\tilde{h}[k]$ that is defined as

$$\tilde{h}[k] \equiv \sum_{p=P_0}^{P_0+P-1} \{d_p \cos(\beta pk) + f_p \sin(\beta pk)\}, \tag{17}$$

for $k \in [-K, K]$ and integers P_0 and P. We set $\tilde{h}[k] = 0$ for $k < -K$ or $k > K$.

For a fixed P, parameters P_0, d_p, and f_p are defined as to minimize the following criterion based on MSE.

$$\sum_{k=-M}^{M} |h[k-n_0] - \tilde{h}[k]|^2 + \lambda(\tilde{h}[-K]^2 + \tilde{h}[K]^2), \tag{18}$$

for the parameter $\lambda > 0$. The second term is added to reduce the discontinuities at the edges that cause undesirable high frequency components. The parameter λ is decided to satisfy the following condition

$$\left|\tilde{h}[K]/(h[K] - h[K-1])\right| \leq 1. \tag{19}$$

This condition makes the discontinuity of $\tilde{h}[k]$ at edges small. Then, the filtered signal $y_M[n]$ is approximately given by

$$\tilde{y}[n] \equiv \sum_{p=P_0}^{P_0+P-1} \{d_p \tilde{c}_p[n+n_0] + f_p \tilde{s}_p[n+n_0]\}. \tag{20}$$

5 Experimental results

In this section, we compare the transfer function of the proposed GMBPF and FIR bandpass filters. We also evaluate the errors caused by floating-point calculations and calculation times.

5.1 Transfer function

For design examples of GMBPF, we assume $Q = 2$, $w_C = 0.4\pi$ (rad/s), and $\omega_\Delta = 0.6\pi/\sigma$ (rad/s). We change σ as 100, 1000, and 10000. We also assume $n_0 = 0.08\sigma$ for ASFT. K, d_p, and f_p are selected such that the criterion given by Eq.(18) becomes minimum.

The FIR bandpass filters are designed using 'designfilt', which is a function of the signal processing toolbox in MATLAB. We apply 'kaiserwin' and set the parameters such that the bandwidths for cutoff (-3dB) and stopband (-60dB or -100dB) are the same as that of GMBPF. The bandwidths by GMBPF using ASFT are shown in Table 1.

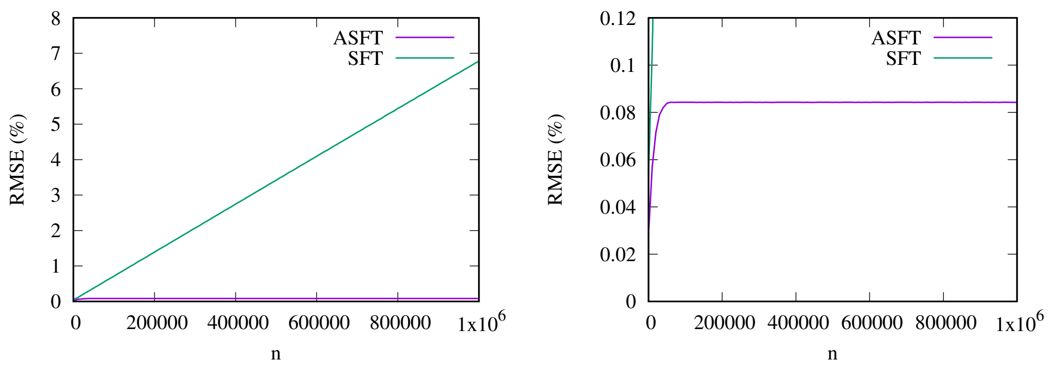
Fig. 2 shows the amplitudes of the transfer functions of GMBPFs and FIR bandpass filters for different values of P and σ. It is observed that GMBPF can realize bandpass filters of stopband attenuations of -60dB and -100dB by $P = 16$ and $P = 27$, respectively.

Fig. 3 shows the phase of the transfer function of GMBPFs. It is observed that GMBPFs have linear phase property.

5.2 Error by floating-point calculation

Fig. 4 shows the accumulated calculation errors by single-precision floating-point calculation. Input signal is a cosine function of which frequency is close to the center frequency of the passband, and the same as one of the frequencies of SFT/ASFT. When both the frequencies are same, the calculation errors are constantly accumulated by SFT. The normalized root mean square error (RMSE) between the output data by the GMBPFs with SFT/ASFT and $y_{5K}[n]$, which is given by Eq.(16) with $M = 5K$, is given by Eq.(21).

$$\mathrm{RMSE}[n] = \sqrt{\sum_{m=n}^{n+9999} |\tilde{y}[m] - y_{5K}[m]|^2 / \sum_{m=n}^{n+9999} |x[m]|^2}. \tag{21}$$



(a) RMSE (b) RMSE (magnified)

Figure 4: Accumulation of floating-point calculation error. ($\sigma = 250$, $\omega_C = 0.4\pi$, and $Q = 2$.)

Table 2: Calculation time (sec). ($N = 1,020,000$)

P	σ	SFT	ASFT	FIR (# of taps)
16	100	0.0160	0.0076	0.0846 (490)
	1000	0.0166	0.0078	0.8075 (4937)
	10000	0.0164	0.0086	7.9891 (45287)
27	100	0.0152	0.0159	0.1326 (762)
	1000	0.0150	0.0169	0.9853 (5597)
	10000	0.0154	0.0173	9.9503 (55211)

The parameters of the GMBPF used in this experiment are $\omega_C = 0.4\pi$, $Q = 2$, $\omega_\Delta = 0.6\pi/\sigma$ (rad/s), and $P = 16$.

From Fig. 4, it is observed that the calculation errors are accumulated constantly by SFT but not by ASFT. We cannot observe such phenomena when we use the double-precision floating-point. Since many economical GPUs do not have double-precision floating-point units, the ASFT has an advantage for such GPUs.

5.3 Calculation time

Table 2 shows the calculation times to apply the input data of length $N = 1,020,000$ to the GMBPFs with $P = 16$ and $\sigma = 100$, 1000, and 10000. It also shows the calculation times by the FIR bandpass filters with the numbers of taps as well. CPU and C compiler used in the calculation are AMD Ryzen 9 3950X and GCC version 9.3.0 with options '-O3 -lm', respectively. Only a core of the CPU is used for this calculation. From Table 2, it can be inferred that the proposed method is much faster than the FIR bandpass filter and calculation time of the proposed method does not depend significantly on σ that corresponds to the passband width. Hence, the proposed GMBPF is suitable to realize narrow passpand BPFs.

6 Conclusion

In this study, we proposed GMBPF and its implementation by FIR recursive filters. Since it has been realized by a symmetric impulse response approximation and its calculation complexity does not depend on its passband width, it is suitable for narrow passed filtering applications. Especially, we used the same σ for impulse responses in the GMBPF. For future work, it is very challenging to use different values of σ to realize a BPF efficiently and implement the proposed algorithm to GPU.

Acknowledgment

A part of this study was supported by JSPS KAKENHI Grant No. 20H04206.

Conflict of interest

The authors declare no potential conflict of interests.

Author contributions

YY conceived of the presented idea, planned and carried out the experiments, and wrote the manuscript. TW made the problem, managed the whole project, and edited the manuscripts from the viewpoint of signal processing.

References

[1] R. J. Wenzel, "Solving the approximation problem for narrowband bandpass filters with equal-ripple passband response and arbitrary phase response," in *Proceedings of 1975 IEEE-MTT-S International Microwave Symposium*, May 1975, p. 50.

[2] C. Britton Rorabaugh, *Digital Filter Designer's Handbook: Featuring C Routines*, McGraw-Hill, Blue Ridge Summit, Pennsylvania, 1993.

[3] K. Martin, "Approximation of complex IIR bandpass filters without arithmetic symmetry," *IEEE Transactions on Circuits and Systems I*, vol. 52, no. 4, pp. 794 – 803, April 2005.

[4] E. Domazet, M. Gusev, and S. Ristov, "Optimizing high-performance CUDA DSP filter for ECG signals," in *Proceedings of the 27th International DAAAM Symposium 2016*, Oct. 2016, pp. 0623–0632.

[5] J. Gunther, H. Gunther, and T. Moon, "GPU acceleration of dsp for communication receivers," *Proceedings of the GNU Radio Conference*, vol. 2, no. 1, pp. 9, 2017.

[6] R. Allred, *Digital Filters for Everyone: Third Edition*, Creative Arts & Sciences House, Indian Harbour Beach, Florida, 2015.

[7] V. Kober, "Fast algorithms for the computation of sliding discrete sinusoidal transforms," *IEEE Transactions on Signal Processing*, vol. 52, no. 6, pp. 1704–1710, Jun. 2004.

[8] E. Elboher and M. Werman, "Cosine integral images for fast spatial and range filtering," in *Proceedings of 2011 18th IEEE International Conference on Image Processing*, Sept. 2011, pp. 89–92.

[9] K. Sugimoto and S. Kamata, "Compressive bilateral filtering," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3357–3369, Nov. 2015.

[10] N. Fukushima, Y. Maeda, Y. Kawasaki, M. Nakamura, T. Tsumura, K. Sugimoto, and S. Kamata, "Efficient computational scheduling of box and Gaussian FIR filtering for CPU microarchitecture," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Nov. 2018, pp. 875–879.

[11] T. Yano, K. Sugimoto, Y. Kuroki, and S. Kamata, "Acceleration of Gaussian filter with short window length using DCT-1," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Nov. 2018, pp. 129–132.

[12] K. Sugimoto, S. Kyochi, and S. Kamata, "Universal approach for DCT-based constant-time Gaussian filter with moment preservation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2018, pp. 1498–1502.

[13] Y. Yamashita and T. Wakahara, "Stabilized calculation of Gaussian smoothing and its differentials using attenuated sliding Fourier transform," in *Proceedings of the 25th International Conference on Pattern Recognition*, Jan. 2021.

[14] Y. Yamashita and T. Wakahara, "Initial location search method using target region features for region-based image matching," *IEICE Transactions on Information and Systems (Japanese Edition)*, vol. J103-D, no. 4, pp. 330–340, Apr. 2020.

[15] Y. Yamashita and T. Wakahara, "Morlet wavelet transform using attenuated sliding fourier transform and kernel integral for graphic processing unit," 2021, `https://arxiv.org/abs/2110.11866v2`.

[16] D. Gabor, "Theory of communication. part 1: The analysis of information," *Journal of Institution of Electrical Engineers*, vol. 93, no. 3, pp. 429–457, 1946.

[17] G. Capizzi, S. Coco, G. L. Sciuto, and C. Napoli, "A new iterative fir filter design approach using a Gaussian approximation," *IEEE Signal Processing Letters*, vol. 25, no. 11, pp. 1615–1619, Aug. 2018.

[18] G. Capizzi and G. Lo Sciuto, "A novel 2-d fir filter design methodology based on a Gaussian-based approximation," *IEEE Signal Processing Letters*, vol. 26, no. 2, pp. 1–1, Jan. 2019.