# Fixed-Point Arithmetic of $\ell_2$-Norm Approximation for 2-Tuple Arrays with Rotated $\ell_1$-Norm Evaluation

Yuya KODAMA*, Shogo MURAMATSU*, and Hiroyoshi YAMADA*
* Niigata University, Niigata, Japan
E-mail: ykodama@ieee.org, shogo@eng.niigata-u.ac.jp, yamada@ie.niigata-u.ac.jp

***Abstract*—This paper proposes a high-precision fast approximation method for the $\ell_2$-norm evaluation of 2-tuple arrays by means of a rotated $\ell_1$-norm evaluation with fixed-point arithmetic. A considerable number of calculations for 2-tuple $\ell_2$-norm are frequently required in several signal processing applications such as image restoration with isotropic total variation and complex $\ell_1$-norm regularization. Typical embedded applications prefer parallel processing, constant scaling, and fixed-point arithmetic compared with serial processing, variable multiplication, and floating-point arithmetic. To achieve a hardware-friendly calculation, square and square root operations should be adequately approximated. However, several existing techniques are challenged with respect to approximations with all three preferable features. Thus, in this paper, a hardware-friendly approximation algorithm is proposed. The proposed method uses the fact that the upper bound of the surface of a first-order rotational cone traces a second-order cone, i.e., $\ell_2$-cone, requests less variable multiplication, and can easily be implemented in parallel with fixed-point arithmetic. To verify the effectiveness of the proposed method, an image restoration performance and software/hardware co-design report are evaluated.**

***Index Terms*—Sparse modeling, Fixed-point arithmetic, Soft-thresholding, Programmable SoC, Embedded vision**

## I. INTRODUCTION

Internet of Things (IoT) is attracting attention to facilitate the development of information societies [1]. IoT conveys a concept of Internet usage where a lot of things (devices) are connected through the Internet, and physical data are collected via servers. Examples of IoT applications include smart homes, structural health monitoring, economic agriculture, assisted living, and safety vehicles [1]. Since the network should be capable of handling heavy traffic and the servers should manage a considerable amount of versatile data within a limited response time, it is necessary to reduce the communication and computational load throughout the system. One way to solve these challenges are to introduce edge computing technologies [2], [3].

Edge computing is a form of distributed processing that assigns tasks not only to servers but also to edge devices, i.e., sensing terminals or computers near sensors. In general, such edge devices are expected to be small as well as lightweight and to consume low power for long-term sustainable ubiquitous sensing. Therefore, tasks on edge devices place a priority on speed and power consumption over quality. Embedded implementation of processing in hardware (HW) is a typical approach to satisfy the highlighted need. Specific IoT applications adopt imaging devices as sensors to exploit visual information. For example, driver assistance technologies use optical cameras and radars to detect objects [4], [5]. Many visual IoT applications, however, work under severe conditions such as vibration, inclement weather, and darkness, among others. Thus, the acquired images are prone to be contaminated. Edge devices are required to restore such images in real-time while satisfying the conditions of low-power consumption. When implementing such algorithms on HW, reducing the usage of circuit resources is required.

Recent advances in image restoration employ the effectiveness of sparsity. The total variation (TV) and $\ell_1$-norm are typical examples of sparsity-promoting regularizers to solve image restoration problems [6]–[11]. Certain solvers for such problems are implemented in an HW-friendly manner with simple arithmetics such as the iterative shrinkage- thresholding algorithm (ISTA). However, certain cases request operations that consume circuit resources, e.g., variable multiplications and non-linear operations. For example, image restoration by means of isotropic TV regularization and the $\ell_1$-norm regularization for complex-valued data demand the 2-tuple $\ell_2$-norm evaluation. The $\ell_2$-norm evaluation requires square root and square operations. These slightly expensive operations result in costs that cannot be ignored, particularly in cases where large amounts of data are involved. Parallel processing, constant scaling, and fixed-point arithmetic are preferable for HW to serial processing, variable multiplication, and floating-point arithmetic [12]. One approach to tackle the square root is to use the CORDIC (COordinate Rotation DIgital Computer) [13], [14]. However, the calculations for squares inside the square root remain. Piecewise linear approximation also suffers from the same problem.

In this work, we focus on the approximation of the $\ell_2$-norm for 2-tuple arrays, because it appears in solvers used in certain important problem settings such as isotropic TV regularized image restoration and complex $\ell_1$-norm regularization. We propose an algorithm for approximating the 2-tuple $\ell_2$-norm evaluation. The proposed algorithm reduces the $\ell_2$-norm evaluation to simple calculations such as constant scaling by using the fact that the upper bound of a rotated first-order, or $\ell_1$-cone traces a second-order, or $\ell_2$-cone [15]. We assess the performance of the proposed method through an image restoration simulation and software/hardware co-design reports.

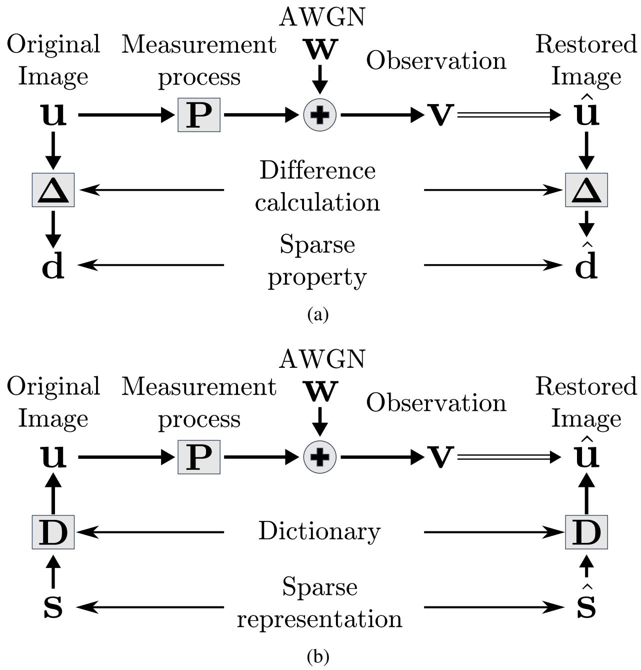



Fig. 1. Typical sparsity-aware image restoration models, where (a) is an analysis dictionary model and (b) is a synthesis dictionary model.

## II. Review of Image Restoration

Let us review the problem settings relating to image restoration that use the $\ell_2$-norm evaluation of 2-tuple elements. For context setting, we briefly describe the isotropic TV regularization and the complex $\ell_1$-norm regularization. In this paper, by the term "2-tuple array," we imply an array of 2-tuple elements. Note that we refer to an element in $\mathbb{R}^2$ and $\mathbb{C}$ as a 2-tuple element in common.

### A. Isotropic TV regularization

Fig. 1 (a) shows a sparsity-aware image restoration model with an analysis dictionary $\boldsymbol{\Delta} \in \mathbb{R}^{n\times m}$. A problem concerning isotropic TV regularization is categorized in the model in Fig. 1 (a), where $\mathbf{u} \in \mathbb{R}^m$ is an unknown original image and $\mathbf{d} \in \mathbb{R}^n$ is a vector consisting of the first-order differences of $\mathbf{u}$ analyzed through $\boldsymbol{\Delta}$ in the vertical and horizontal directions, $\mathbf{v} \in \mathbb{R}^q$ is an observation of $\mathbf{u}$ and assumed to be measured through the process $\mathbf{P} \in \mathbb{R}^{q\times m}$ with additive white Gaussian noise (AWGN) $\mathbf{w} \in \mathbb{R}^q$. Their relation can be formulated as

$$\mathbf{d} = \boldsymbol{\Delta}\mathbf{u} \tag{1}$$

$$\mathbf{v} = \mathbf{P}\mathbf{u} + \mathbf{w}. \tag{2}$$

An image restoration problem based on the isotropic TV regularization is formulated as

$$\hat{\mathbf{u}} = \arg \min_{\mathbf{u}\in[u_{\min},u_{\max}]^n} \frac{1}{2}\|\mathbf{P}\mathbf{u} - \mathbf{v}\|_2^2 + \lambda\|\boldsymbol{\Delta}\mathbf{u}\|_{1,2}, \tag{3}$$

where $\|\cdot\|_2$ and $\|\cdot\|_{1,2}$ denote the $\ell_2$-norm and mixed $\ell_1$-$\ell_2$-norm, respectively, $u_{\min}$, and $u_{\max}$ are the minimum and maximum values of $\mathbf{u}$, and $\lambda$ is a regularization parameter [6]. Note that $\|\boldsymbol{\Delta}\mathbf{u}\|_{1,2}$ is refered to as the isotropic TV of $\mathbf{u}$.

We can adopt the primal-dual splitting (PDS) algorithm to solve the problem in (3) [16], [17]. In the PDS solver, the generalized soft-thresholding expressed by

$$\left[\mathrm{prox}_{\frac{1}{\tau}\|\cdot\|_{1,2}}(\mathbf{x})\right]_i = [\mathbf{x}]_i \odot \max\left(\mathbf{1} - \tau\mathbf{1} \oslash \|[\mathbf{x}]_i\|_2, \mathbf{0}\right) \tag{4}$$

is used as the proximity operator of $\|\cdot\|_{1,2}$, where $[\cdot]_i$ denotes the $i$-th tuple consisting of the vertical and horizontal differences at the $i$-th position, $\tau > 0$ denotes the threshold value, $\odot$ and $\oslash$ denote element-wise multiplication and division, respectively, $\max(\cdot,\cdot)$ denotes a vector consisting of greater elements between the first and second argument elementwisely, $\mathbf{0}$ and $\mathbf{1}$ denote the vectors of zeros and ones, respectively. (4) performs the soft-thresholding for every element in the $i$-th tuple. In the isotropic TV case, each tuple has two elements and consists of differences between adjacent pixels in the vertical and horizontal directions.

### B. Complex $\ell_1$-norm regularization

Fig. 1 (b) shows a sparsity-aware image restoration model with a synthesis dictionary $\mathbf{D} \in \mathbb{K}^{m\times n}$, where $\mathbb{K}$ is a field such as $\mathbb{R}$ and $\mathbb{C}$. The observation image $\mathbf{v} \in \mathbb{K}^q$ is assumed to be

$$\mathbf{v} = \mathbf{P}\mathbf{u} + \mathbf{w}, \tag{5}$$

where $\mathbf{P} \in \mathbb{K}^{q\times m}$ is the observation process, $\mathbf{w} \in \mathbb{K}^q$ is additive white Gaussian noise, and $\mathbf{u} \in \mathbb{K}^m$ is an unknown original image represented by

$$\mathbf{u} = \mathbf{D}\mathbf{s}. \tag{6}$$

The least absolute shrinkage and selection operator (Lasso) problem is categorized in the model in Fig. 1 (b) [10], and the problem setting is formulated as

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}\in\mathbb{K}^n} \frac{1}{2}\|\mathbf{P}\mathbf{D}\mathbf{s} - \mathbf{v}\|_2^2 + \lambda\|\mathbf{s}\|_1, \tag{7}$$

where $\|\cdot\|_1$ denotes the $\ell_1$-norm. To solve (7), we can adopt ISTA.

In this study, we are interested in the complex-valued case when $\mathbf{u} \in \mathbb{C}^m$. In complex ISTA, the complex soft-thresholding expressed by

$$\left[\mathrm{prox}_{\frac{1}{\tau}\|\cdot\|_1}(\mathbf{x})\right]_i = x_i \cdot \max\left(1 - \tau/|x_i|, 0\right) \tag{8}$$

is used as the proximity operator of $\|\cdot\|_1$, where $[\cdot]_i$ denotes the $i$-th element of its argument and $x_i = [\mathbf{x}]_i \in \mathbb{C}$ [18].

### C. Problems for embedded implementation

In the generalized soft-thresholding in (4), it is necessary to calculate the $\ell_2$-norm of the 2-tuple element, i.e.,

$$\|[\mathbf{x}]_i\|_2 = \sqrt{x_{i,1}^2 + x_{i,2}^2}. \tag{9}$$

It is also necessary to take the absolute of complex element in the complex soft-thresholding in (8), i.e.,

$$|x_i| = \sqrt{\Re^2(x_i) + \Im^2(x_i)}. \tag{10}$$

The calculations of two formulas are the same in that both of them take the square root of the sum of the squares of two components. Note that all tuples should be evaluated in every iteration of the restoration algorithms. The $\ell_2$-norm is essentially important to gurantees the phase invariance in the complex plane and the rotation invariance in the 2-D Euclidean space, where the approximation by the $\ell_1$-norm severly violates those characteristics.

In the embedded implementation, it is desirable to use fixed-point arithmetic from the viewpoint of circuit resources and calculation speed. As well, constant scaling is preferable to variable multiplication to reduce circuit resource usage through sophisticated approaches such as canonical signed digit (CSD) representation, reduced adder graph (RAG), and distributed arithmetic (DA) [12]. For the square root function, the calculation can be approximately realized by several existing methods such as CORDIC and piecewise linear approximation(PLA). However, such techniques do not allow us to avoid variable multiplications within the square root function.

## III. Proposed 2-tuple $\ell_2$-norm Approximation

In this section, we propose an approximation method for the 2-tuple $\ell_2$-norm evaluation with fixed-point arithmetic to solve the problems discussed in the previous section.

### A. Second-order cone via rotational first-order cone

Let us first express a key idea of our proposed approximation. We identically represent (9) and (10) in a form without the square root and square operations. We express the real and imaginary part of a complex value $x_i$ as $x_{i,1} = \Re(x_i)$ and $x_{i,2} = \Im(x_i)$, respectively. As a result, the right-hand sides of the two equations became equal. Thus, in the following section, we discuss the two cases in common.

Our proposed method has the basis of the following theorem:

**Theorem 1** (Second-order cone via first-order rotational cone)**.** *Let a rotated first-order cone function* $a_\theta \colon \mathbb{R}^2 \to \mathbb{R}$ *be*

$$a_\theta([\mathbf{x}]_i) := \frac{1}{\sqrt{2}}\Big(|x_{i,1}\cos\theta + x_{i,2}\sin\theta| + |x_{i,1}\sin\theta - x_{i,2}\cos\theta|\Big), \tag{11}$$

*where* $[\mathbf{x}]_i = (x_{i,1}, x_{i,2})^\intercal \in \mathbb{R}^2$. *Then,* $a_\theta(\cdot)$ *relates to the* $\ell_2$*-norm of 2-tuple element* $[\mathbf{x}]_i$ *as*

$$\|[\mathbf{x}]_i\|_2 = \sup_{\theta \in [0, \frac{\pi}{2})} a_\theta([\mathbf{x}]_i). \tag{12}$$

*Proof:* Let $\angle[\mathbf{x}]_i = \tan^{-1}\frac{x_{i,2}}{x_{i,1}}$. Then, we see $a_\theta([\mathbf{x}]_i) = \frac{\|[\mathbf{x}]_i\|_2}{\sqrt{2}}\left(|\cos(\angle[\mathbf{x}]_i - \theta)| + |\sin(\angle[\mathbf{x}]_i - \theta)|\right)$. When $\theta = \angle[\mathbf{x}]_i - \frac{\pi}{4}$, we have the relation

$$a_{\angle[\mathbf{x}]_i - \frac{\pi}{4}}(x) = \frac{\|[\mathbf{x}]_i\|_2}{\sqrt{2}}\left(\left|\cos\frac{\pi}{4}\right| + \left|\sin\frac{\pi}{4}\right|\right) = \|[\mathbf{x}]_i\|_2. \tag{13}$$

Besides, we can equate the derivative to zero as

$$\frac{\partial a_\theta}{\partial \theta} = \frac{\|[\mathbf{x}]_i\|_2}{\sqrt{2}}\left(-\sin(\angle[\mathbf{x}]_i - \theta) + \cos(\angle[\mathbf{x}]_i - \theta)\right) = 0. \tag{14}$$

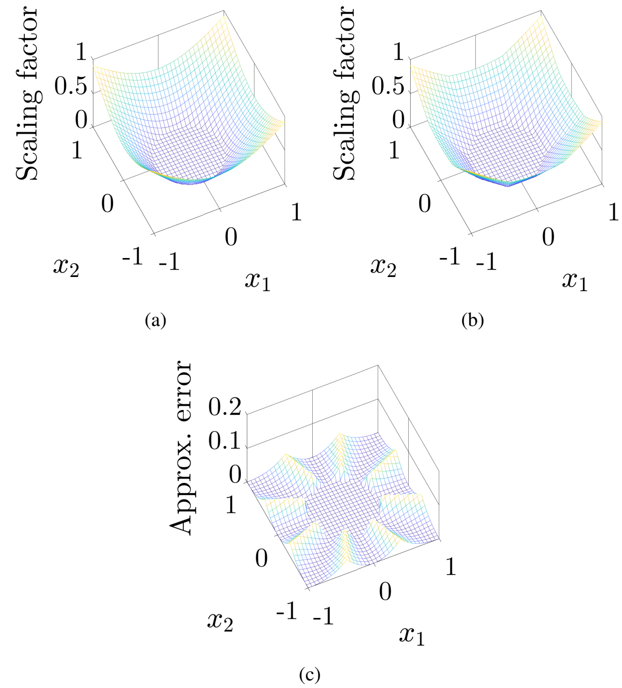



Fig. 2. Soft-thresholding factor of the generalized soft-thresholding with $\tau = 0.5$, where (a) $\max(1 - \tau/\|(x_1, x_2)^\intercal\|_2, 0)$, (b) $\max(1 - \tau/A_N((x_1, x_2)^\intercal), 0)$ with $N = 2$, and (c) approximation error.

As a result, we see that $\theta = \angle[\mathbf{x}]_i - \frac{\pi}{4}$ gives the maximum value for $0 \leq \angle[\mathbf{x}]_i - \theta < \frac{\pi}{2}$, which means that (12) holds from the fact that $a_\theta(\cdot) = a_{\theta + \frac{\pi}{2}k}(\cdot)$ always holds for any $k \in \mathbb{Z}$. ■

### B. Approximation of 2-tuple $\ell_2$-norm evaluation

Let us derive an approximation method for the $\ell_2$-norm evaluation of 2-tuple elements. From Theorem 1, we see that (12) can be approximated by discretizing the angle $\theta$. From this consideration, we propose to approximate (4) as follows:

$$\left[\mathrm{prox}_{\frac{1}{\tau}\|\cdot\|_{1,2}}(\mathbf{x})\right]_i \approx [\mathbf{x}]_i \odot \max\left(\mathbf{1} - \tau\mathbf{1} \oslash A_N([\mathbf{x}]_i), \mathbf{0}\right), \tag{15}$$

where

$$A_N([\mathbf{x}]_i) := \max_{\theta \in \Theta} a_\theta([\mathbf{x}]_i),$$
$$\Theta = \{\theta_0, \theta_1, \ldots, \theta_{N-1}\}, \quad \theta_k = \frac{\pi}{2N}k. \tag{16}$$

Algorithm 1 shows the procedure to realize (15). Fig. 2 (a) and (b) confirm the soft-thresholding factor of the generalized soft-thresholding in (4) and that given by the proposed approximation in (15) with $N = 2$, respectively, where $\tau = 0.5$. The approximation error results in Fig. 2 (c), where it is observed that there is no approximation error in the direction where $\theta = 0$ and $\pi/4$, as indicated by the proof of Theorem 1. The maximum error in the range shown in Fig. 2 (c) is

**Algorithm 1** Approximated 2-tuple generalized soft-thresholding

**Input:** $(x_1, x_2)^\mathsf{T} \in \mathbb{R}^2, \tau > 0, N \in \mathbb{N}$
**Output:** $(y_1, y_2)^\mathsf{T} \in \mathbb{R}^2$
**for each** $k \in \{0, 1, 2, \cdots, N-1\}$ **do**
$c_k, s_k \leftarrow \frac{1}{\sqrt{2}} \cos \frac{\pi}{2N} k, \frac{1}{\sqrt{2}} \sin \frac{\pi}{2N} k$
$a_k \leftarrow |c_k x_1 + s_k x_2| + |s_k x_1 - c_k x_2|$
**end for each**
$a \leftarrow \max(a_0, a_1, a_2, \ldots, a_{N-1}, \tau)$
$y_1, y_2 \leftarrow x_1 \left(1 - \frac{\tau}{a}\right), x_2 \left(1 - \frac{\tau}{a}\right)$

TABLE I
COMPARISON OF EXISTING AND PROPOSED METHODS.

| Method | CORDIC | PLA | Proposal |
|---|---|---|---|
| Fixed-point arithmetic | Feasible | Feasible | Feasible |
| Parallel processing | Difficult | Feasible | Feasible |
| Constant scaling | Impossible | Impossible | Feasible |

approximately 0.0412. One can verify that the error decreases as $N$ increases.

Note that it is possible to calculate $\{c_k\}_k$ and $\{s_k\}_k$ in Algorithm 1 in advance and store them as fixed-point constants. Finally, we obtain a simple approximation procedure for the 2-tuple $\ell_2$-norm evaluation with fixed-point constant scaling.

A comparison of the existing and proposed methods is shown in Table I. In contrast to CORDIC and PLA, the proposed method is feasible, in terms all of the fixed-point arithmetic, parallel processing and constant scaling.

## IV. PERFORMANCE EVALUATION

To evaluate the performance of the proposed method, we first demonstrate the simulation results of image restoration with floating-point arithmetic. Then, we evaluate the precision and resource usage of the fixed-point HW implementation and compare the proposed method to other approximation methods.

### A. Evaluation method

We evaluate the validity of the proposed method through the applications to image restoration. We used monochrome image to evaluate the TV, and millimeter wave range spectrum to evaluate the complex $\ell_1$-norm regularization.

The range spectrum $s_b$ is generated by

$$s_b(x, y, x_u) = \sum_k \sigma_k w_k(x, y, x_u) T_p \exp\left(j \frac{4\pi f_c}{c} r_k(x_u)\right) \cdot \operatorname{sinc}\left(\frac{2\pi B}{c}(r(x, y, x_u) - r_k(x_u))\right), \tag{17}$$

$$w_k(x, y, x_u) = \begin{cases} 1 & \left|\tan^{-1} \frac{x - x_u}{y} - \theta_s\right| \leq \frac{\delta\theta}{2} \cap \\ & \left|\tan^{-1} \frac{x_k - x_u}{y} - \theta_s\right| \leq \frac{\delta\theta}{2} \\ 0 & \text{otherwise} \end{cases}, \tag{18}$$

where $x$ and $y$ and are positions in the along track (azimuth) and range direction, respectively, and $r(x, y, x_u)$ is the range distance between the antenna position and observation point $(x, y)$, $s_b$ is an image created by mapping a range data on the beat-spectrum at $r_k(x_u) := r(x_k, y_k, x_u)$, the range from the antenna to the $k$-th target [19], [20]. In 17, $c$ is the speed of light, $\sigma_k$ is the reflection coefficient of the $k$-th target, $f_c$ is the center frequency, $B$ is the band width, $T_p$ is the sweep time, $\theta_s$ is the squint angle, and $\delta\theta$ is the hone angle. We perform denoising before synthesizing a SAR image. A simulation is conducted with a fixed antenna position $x_u$.

TABLE II
SIMULATION CONDITIONS OF IMAGE RESTORATION MODELED BY THE ISOTROPIC TV REGULARIZATION.

| | |
|---|---|
| Image size | $256 \times 256$ |
| Bit depth | 8bpp |
| Scale | $[0, 1]$ |
| Standard deviation of AWGN | $\sigma$ =10/255 |
| Pixel loss rate | 50% |
| Regularization parameter $\lambda$ | 0.01 |
| $\sharp$ of angles | $N = 2$ |
| $\sharp$ of CORDIC iterations | 5 |
| Piecewise linear Approx. | [0, 2] with nodes set to every 0.3 |

TABLE III
SIMULATION CONDITIONS OF IMAGE RESTORATION MODELED BY THE COMPLEX $\ell_1$-NORM REGULARIZATION.

| | |
|---|---|
| Image size | $2000 \times 600$ |
| Center frequency | $f_c = 76.5 \times 10^9$ |
| Band width | $B = 0.9 \times 10^9$ |
| Sweep time | $T_p = 0.5 \times 10^{-3}$ |
| Squint angle | $\theta_s = 45°$ |
| Hone angle | $\delta\theta = 20°$ |
| Standard deviation of AWGN | $\sigma = 3.16 \times 10^{-6}$ |
| Regularization parameter $\lambda$ | $4 \times 10^{-6}$ |
| Synthesis dictionary $\mathbf{D}$ | 2-Lv. Undecimated Haar Trans. |
| $\sharp$ of angles | $N = 2$ |
| $\sharp$ of CORDIC iterations | 5 |
| Piecewise linear Approx. | [0, 2] with nodes set to every 0.3 |

The variation of Peak Signal-to-Noise Ratio (PSNR) of the restored images is observed by changing the bit width of the fractional part in fixed-point arithmetic. Furthermore, we implement a part of the proposed method with HW and evaluate the resource utilization by changing the bit width. Our simulations are conducted using MATLAB R2019a, and HW resources are estimated by the Xilinx SDSoC 2016.2 for the ZC706 Evaluation Kit used as the target device. Tables II and III summarize the simulated conditions for image restoration modeled by the isotropic TV regularization and complex $\ell_1$-norm regularization, respectively.

### B. Simulation of restoration performance

Let us show the restoration performance of the proposed method with floating-point arithmetic. Figs. 3 and 4 show the image restoration results by the isotropic TV regularization and complex $\ell_1$-norm regularization, respectively, where PSNR for complex images is calculated by using one as the peak value and the average of mean squared error (MSE) values of the real and imaginary part as the total MSE. In Fig. 3, (a) and (b) show the original and observation image, respectively, (c) is a restored image with the original

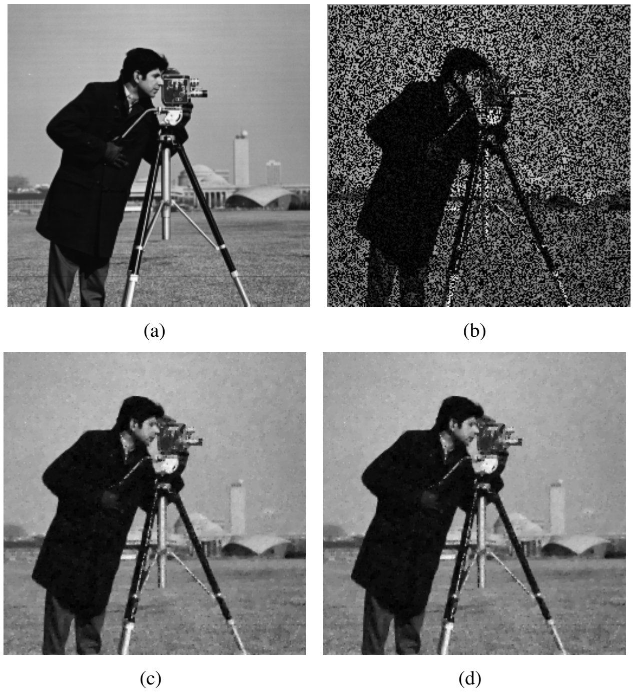



Fig. 3. Simulation results of image restoration by means of isotropic TV regularization. (a) Original image, (b) Observation image (PSNR : 8.60 dB), (c) Restored image without approximation (PSNR : 26.46 dB), and (d) Restored image with the proposed approximation (PSNR : 26.45 dB).

$\ell_2$-norm evaluation, and (d) represents the image with the proposed approximation. In Fig. 4, (a) shows the observation image, (b) is a restored image with the original $\ell_2$-norm evaluation, and (c) represents the image with the proposed approximation. The difference between Fig. 3 (c) and (d), i.e., the approximation error, is approximately 0.01 dB with respect to PSNR.

Let us also compare the restoration performance of the proposed method to that of conventional approximations with fixed-point arithmetic. Fig. 5 (a) and (b) show the variation in PSNR for the isotropic TV regularization and complex $\ell_1$-norm regularization case, respectively, where the integer part is fixed to 6 bits, and the number of bits in the fractional part is changed.

*1) Isotropic TV regularization case:* From Fig. 5 (a), it is observed that the PSNR increases as the width of the fractional part increases and shows a significant rise around 6 bits. This is because the fractional part of the regularization parameter $\lambda$ becomes valid when the number of bits is greater than 6, and the generalized soft-thresholding works properly.

*2) Complex $\ell_1$-norm regularization case:* Fig. 5 (b) compares the influence of the width of the fractional part to the PSNR of the complex $\ell_1$-norm regularization case. The PSNR rises once when the fractional part exceeds 7 bits. This is because the fractional part of the regularization parameter $\lambda$ becomes valid for more than 7 bits, and the complex soft-thresholding begins to work. PSNR, more than 24 dB, requires the fraction to be of width more than 10 bits. Our proposed approximation shows almost the same performance as other conventional techniques for more than 10 bits in the fractional part. Without significant performance loss, we can set the fractional part to 10 bits, and our approximation can be adopted with $N = 2$.

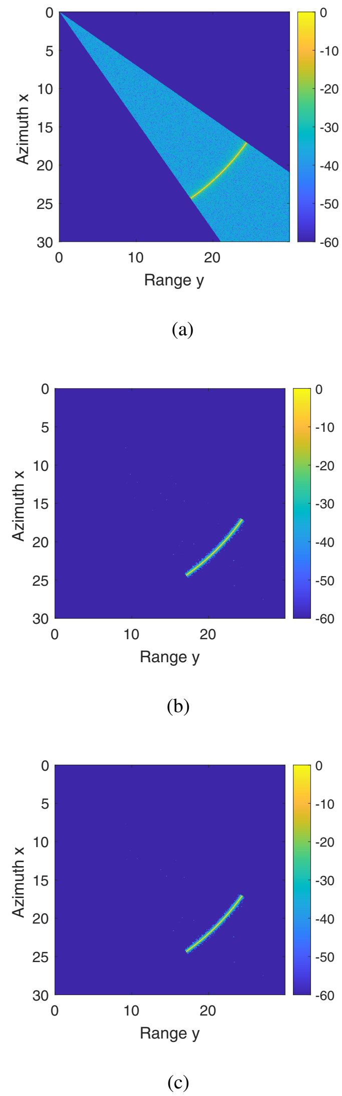
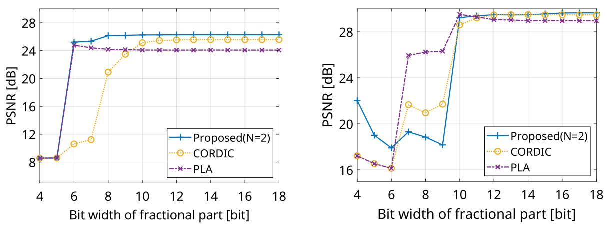



Fig. 4. Simulation results of image restoration by means of complex $\ell_1$-norm regularization. The observation images were generated by superimposition of $s_b$ and AWGN. (a) Observation image, (b) Restored image without approximation, and (c) Restored image with approximation.

### *C. Evaluation of FPGA implementation*

Let us evaluate the FPGA implementation and compare the results among the approximation techniques under discussion. The scope of the FPGA implementation is soft-thresholding. Fig. 6 compares the resource utilization of the proposed method to that of the others, where the integer and fractional parts are fixed to 15 bits and 10 bits, respectively. Fig. 6 shows that the amount of resources utilized in the proposed




(a) (b)

Fig. 5. Image restoration performance of CORDIC, piecewise linear approximation (PLA) and the rotated $\ell_1$-norm approximation (Proposed), where the bit width of the fractional part is changed. (a) PSNR for the isotropic TV regularization and (b) PSNR for the complex $\ell_1$-norm regularization.

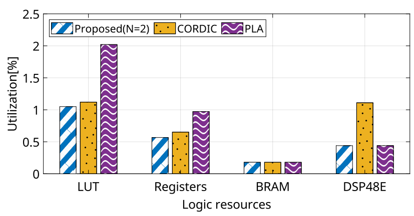



Fig. 6. Logic resource utilization of soft-thresholding with three approximation methods in FPGA, where the integer and fractional parts are fixed to 15 and 10 bits, respectively.The proposed method used the least amount of LUTs and Registers compared to the existing method; the amount of BRAM used was the same; the amount of DSP used was similar between the proposed method and PLA.

method is the lower than or equal to that of the other methods. This is because our proposed method requires fewer variable multiplications than other methods. Table IV shows a comparison of the execution speed on the FPGA. The proposed method is faster than the existing methods as it takes less time for a one-time calculation. Also, the increase in computation time is small because the proposed method can be run in parallel even when the number of $N$ is increased.

## V. Conclusions

In this paper, we proposed an $\ell_2$-norm evaluation method for 2-tuple arrays, which is suitable for fixed-point arithmetic. We assessed the application by applying it to image restoration and evaluated the HW resource utilization through the FPGA design. The degradation of the proposed approximation is verified to be insignificant with lower resource utilization than other conventional techniques. In the future, we will apply the proposed method to an embedded vision system, particularly for millimeter-wave radar image processing.

TABLE IV
Comparison of execution speed between existing and proposed methods on FPGA.

| Method | CORDIC | PLA | Proposal ($N = 2$) | Proposal ($N = 4$) |
|---|---|---|---|---|
| Latency [us] | 22.770 | 23.770 | 22.010 | 22.210 |

## References

[1] Y. Chen, "Challenges and opportunities of internet of things," *17th Asia and South Pacific Design Automation Conference*, pp. 383–388, Jan. 2012.
[2] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
[3] A. Akhtar, B. Kathariya, and Z. Li, "Low latency scalable point cloud communication," *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 2369–2373, Sep. 2019.
[4] W. Zhi-Hao, L. Siou-Wei, C. Yaw-Jong, H. Shih-Hsien, L. Yen-Cheng, and J. Gwo-Jia, "Study on vehicle safety image system optimization," in *2017 5th International Conference on Electrical, Electronics and Information Engineering (ICEEIE)*, Oct. 2017, pp. 96–101.
[5] T. Kobayashi, H. Yamada, Y. Yamaguchi, and Y. Sugiyama, "Simple squint angle compensation for sar with millimeter wave automotive radar," *2017 International Symposium on Antennas and Propagation (ISAP)*, pp. 1–2, Oct. 2017.
[6] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992.
[7] P. L. Combettes and J. C. Pesquet, "Image restoration subject to a total variation constraint," *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp. 1213–1222, Sep. 2004.
[8] T. F. Chan, J. Shen, and H. Zhou, "Total variation wavelet inpainting," *Journal of Mathematical Imaging and Vision*, vol. 25, no. 1, pp. 107–125, 2006.
[9] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–20, 2009.
[10] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.
[11] L. Deng, H. Guo, and T. Huang, "A fast image recovery algorithm based on splitting deblurring and denoising," *Journal of Computational and Applied Mathematics*, vol. 287, pp. 88–97, 2015.
[12] S. A. White, "Applications of distributed arithmetic to digital signal processing: a tutorial review," *IEEE ASSP Magazine*, vol. 6, no. 3, pp. 4–19, July. 1989.
[13] E. Monmasson and M. N. Cirstea, "FPGA design methodology for industrial control systems—a review," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1824–1842, Aug. 2007.
[14] R. Andraka, "A survey of CORDIC algorithms for FPGA based computers," *Proceedings of the 1998 ACM/SIGDA Sixth International Symposium on Field Programmable Gate Arrays*, pp. 191–200, 1998.
[15] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.
[16] L. Condat, "A primal-dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms," *Journal of Optimization Theory and Applications*, vol. 158, no. 2, pp. 460–479, 2013.
[17] S. Ono and I. Yamada, "Hierarchical convex optimization with primal-dual splitting," *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 373–388, Jan. 2015.
[18] S. Nagayama, S. Muramatsu, and H. Yamada, "Single-image super-resolution using complex nonseparable oversampled lapped transforms," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Nov. 2018, pp. 1309–1312.
[19] Akira Oshima, Hiroyoshi Yamada, and Shogo Muramatsu, "Experimental Study on Automotive Millimeter Wave SAR in Curved Tracks," in *2019 International Symposium on Antennas and Propagation (ISAP)*, Oct. 2019, pp. 1–2.
[20] Angel Ribalta, "Time-domain reconstruction algorithms for FMCW-SAR," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 3, pp. 396–400, 2011.