Mahmmod, BM, Abdulhussain, SH, Suk, T, Alsabah, M and Hussain, A

Accelerated and Improved Stabilization for High Order Moments of Racah Polynomials

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 September 2023, accepted 29 September 2023, date of publication 4 October 2023,
date of current version 11 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3321969

RESEARCH ARTICLE

Accelerated and Improved Stabilization for High Order Moments of Racah Polynomials

BASHEERA M. MAHMMOD[1], SADIQ H. ABDULHUSSAIN[1], TOMÁŠ SUK[2], (Member, IEEE), MUNTADHER ALSABAH[3], AND ABIR HUSSAIN[4,5], (Senior Member, IEEE)
[1]Department of Computer Engineering, University of Baghdad, Baghdad 10071, Iraq
[2]Institute of Information Theory and Automation, Czech Academy of Sciences, 182 08 Prague, Czech Republic
[3]Medical Technical College, Al-Farahidi University, Baghdad 10071, Iraq
[4]School of Computer Science and Mathematics, Liverpool John Moores University, L3 3AF Liverpool, U.K.
[5]Department of Electrical Engineering and Computer Science, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates

Corresponding author: Abir Hussain (abir.hussain@sharjah.ac.ae)

This work was supported in part by the Czech Science Foundation under Grant GA21-03921S, and in part by Praemium Academiae.

ABSTRACT Discrete Racah polynomials (DRPs) are highly efficient orthogonal polynomials and used in various scientific fields for signal representation. They find applications in disciplines like image processing and computer vision. Racah polynomials were originally introduced by Wilson and later modified by Zhu to be orthogonal on a discrete set of samples. However, when the degree of the polynomial is high, it encounters numerical instability issues. In this paper, we propose a novel algorithm called Improved Stabilization (ImSt) for computing DRP coefficients. The algorithm partitions the DRP plane into asymmetric parts based on the polynomial size and DRP parameters. We have optimized the use of stabilizing conditions in these partitions. To compute the initial values, we employ the logarithmic gamma function along with a new formula. This combination enables us to compute the initial values efficiently for a wide range of DRP parameter values and large polynomial sizes. Additionally, we have derived a symmetry relation for the case when the Racah polynomial parameters are zero ($a = 0$, $\alpha = 0$, $\beta = 0$). This symmetry makes the Racah polynomials symmetric about the main diagonal, and we present a different algorithm for this specific scenario. We have demonstrated that the ImSt algorithm works for a broader range of parameters and higher degrees compared to existing algorithms. A comprehensive comparison between ImSt and the existing algorithms has been conducted, considering the maximum polynomial degree, computation time, restriction error analysis, and reconstruction error. The results of the comparison indicate that ImSt outperforms the existing algorithms for various values of Racah polynomial parameters.

INDEX TERMS Racah polynomials, recurrence formulas, stabilizing condition, improved stabilization, orthogonal moments.

I. INTRODUCTION

One effective method for representing different objects is to use a set of basic functions and calculate the object's projection onto this basis [1], [2]. When these basic functions are polynomials, the resulting numerical characteristics are referred to as moments. The use of orthogonal polynomials is advantageous because it reduces the correlation between moments and improves the numerical stability of both the polynomials and the moments, particularly at higher degrees and orders [3], [4]. (The order of the moment corresponds to the degree of the polynomial.)

Orthogonal polynomials and the moments derived from them find widespread applications in various fields [5], [6], [7]. For instance, Usman et al. [8] employed Gegenbauer polynomials to solve nonlinear physical models, while Chakraborty and Jung [9] utilized Hermite, Legendre, Laguerre, Jacobi, and generalized Laguerre polynomials to model the impact of continuous random variables described by normal, uniform, exponential, beta, and gamma probability distributions, respectively. Feinberg and Langtangen [10] discussed the application of orthogonal polynomials

The associate editor coordinating the review of this manuscript and approving it for publication was Larbi Boubchir.

for uncertainty quantification. In the context of image classification, Abbas and George [11] utilized discrete Fourier transformation, wavelets, and statistical moments. Additionally, Chebyshev polynomials were employed for solving differential equations, as described in [12]. Interesting way of Zernike moments can be found in [13].

Orthogonal polynomials can be classified into two categories: continuous and discrete. Continuous orthogonal polynomials establish orthogonality based on integrals over a specific interval. However, when computing continuous moments from digital signals that are defined only at discrete samples, there is an inherent error due to the approximated computation of integral definitions. This has led to extensive research on polynomials with discrete orthogonality, which rely on summations over finite sets of discrete samples.

Over time, various types of discrete orthogonal polynomials have been developed. Chebyshev initially derived two types of continuous polynomials. More recently, new variations of Chebyshev polynomials have emerged, such as the three and four [14] and six [15] polynomial families. In addition to his well-known continuous polynomials, Chebyshev also introduced discrete versions. Mukundan devised an efficient algorithm for computing these discrete Chebyshev polynomials [16]. Special approach to the computation of Chebyshev moments based on Z-transformation is applied in [17].

Another example is the Krawtchouk polynomials, which include a parameter $p \in \langle 0, 1 \rangle$. Adjusting this parameter shifts the zeros of the polynomials, allowing for the customization of the region of interest. An efficient algorithm for computing Krawtchouk polynomials can be found in [18], while an alternative method employing filters was published in [19]. Other algorithm is in [20]. Meixner polynomials, which are a generalization of Krawtchouk polynomials, have an efficient algorithm described in [21]. Lastly, the computation of Hahn polynomials can be achieved using the algorithm outlined in [22].

The Racah polynomials, named after physicist and mathematician Giulio Racah, were initially introduced by Wilson in [23]. These polynomials are associated with a non-uniform lattice defined as $x(s) = s(s + \gamma + \delta + 1)$ (refer to [24]). However, utilizing them in practice can be challenging. To address this, Zhu et al. [25] made slight modifications to the definition, using the index s as a coordinate within the object. The Racah moments derived from these polynomials have been applied in various applications, including skeletonization of craft images [26], Chinese character recognition [27], handwritten digit recognition [28], and face recognition [29].

Gasper and Rahman derived q-extension of the Racah polynomials [30]. Benouini et al. proposed the moment invariants to translation, rotation and scaling based on the Racah polynomials in [31]. El Mallahi et al. in [32] used polar coordinates for derivation of 2D and 3D rotation invariants. Lakhili et al. applied neural network on 3D Racah moments computed in Cartesian coordinates in [33]. In [34], Batioua et al. combine Racah polynomials with other types.

In this paper, we present an efficient algorithm for computing Racah polynomials of high degrees. The contribution of this paper can be summarized as follows:

1) The paper introduces a new algorithm called Improved Stabilization (ImSt) which is specifically designed to compute DRP coefficients. This algorithm addresses the numerical instability issues encountered with high-degree DRPs. The proposed algorithm is applicable for a wide range of parameters and higher degrees compared to existing algorithms.
2) The proposed algorithm partitions the DRP plane into asymmetric parts. In addition, the proposed algorithm optimizes the use of stabilizing conditions within these partitions.
3) The paper presents the use of the logarithmic gamma function and a new formula to efficiently compute initial values for a wide range of DRP parameter values and large polynomial sizes.
4) A new symmetry relation for the case when Racah polynomial parameters are set to zeros.
5) We conduct a comprehensive comparison between the ImSt algorithm and existing algorithms. We consider factors such as maximum polynomial degree, computation time, restriction error analysis, and reconstruction error in the comparison.

The structure of the paper is organized as follows: Section II provides a summary of definitions, Section III describes our proposed method, ImSt, and discusses its properties. The applications of ImSt are demonstrated in Section IV, and Section V concludes the paper. The summary of the state-of-the-art algorithms and some proofs are in Appendix.

II. PRELIMINARIES AND RELATED WORK

In this section, the mathematical definitions and fundamentals of the discrete Racah polynomials (DRP) are presented.

The original Wilson's definition [23] can be written as

$$\mathcal{R}_n^{\left(\substack{\alpha,\beta\\ \gamma,\delta}\right)}(\lambda(x); N) = {}_4F_3\left(\left.\begin{matrix} -n, n{+}\alpha{+}\beta{+}1, -x, x{+}\gamma{+}\delta{+}1 \\ \alpha+1, \beta+\delta+1, \gamma+1 \end{matrix}\right| 1\right), \quad (1)$$

where ${}_4F_3(\cdot)$ is the hypergeometric series. It is defined as follows

$${}_4F_3\left(\left.\begin{matrix} a, b, c, d \\ e, f, g \end{matrix}\right| z\right) = \sum_{k=0}^{\infty} \frac{(a)_k\,(b)_k\,(c)_k\,(d)_k}{(e)_k\,(f)_k\,(g)_k k!}\,(z)^k . \quad (2)$$

The symbol $(\cdot)_m$ is the Pochhammer symbol defined as

$$(a)_m = a(a+1)(a+2)\cdots(a+m-1). \quad (3)$$

Zhu et al. in [25] introduced a new variable s and defined as $x = s(s+1)$. Then DRPs $\mathcal{R}_n^{\left(\substack{\alpha,\beta\\ a,b}\right)}(s; N)$ of the nth degree are given by

$$\begin{aligned}\mathcal{R}_n^{\left(\substack{\alpha,\beta\\a,b}\right)}&(s;N)\\ &=\frac{1}{n!}(a+b+\alpha+1)_n(\beta+1)_n(a-b+1)_n\times\\ &\quad\times\ 4F_3\left(\begin{matrix}-n,a-s,a+s+1,\alpha+\beta+n+1\\ \beta+1,a+b+\alpha+1,a-b+1\end{matrix}\,\middle|\,1\right).\end{aligned}\tag{4}$$

The DRPs satisfy the condition of orthogonality

$$\sum_{s=a}^{b-1}\mathcal{R}_n^{\left(\substack{\alpha,\beta\\a,b}\right)}(s;N)\,\mathcal{R}_m^{\left(\substack{\alpha,\beta\\a,b}\right)}(s;N)\,\rho(s)\Delta x\left(s-\frac{1}{2}\right)=d_n^2\delta_{nm},\tag{5}$$

where δ_{nm} is the Kronecker delta, $\Delta x\left(s-\frac{1}{2}\right)$ is the difference of the x shifted by a half, i.e. $\Delta x\left(s-\frac{1}{2}\right)=\left(s+\frac{1}{2}\right)\left(s+\frac{3}{2}\right)-\left(s-\frac{1}{2}\right)\left(s+\frac{1}{2}\right)=(2s+1)$, ρ is the weight function of DRP

$$\rho(s)=\frac{\Gamma(a+s+1)\Gamma(b+s+\alpha+1)\Gamma(b+\alpha-s)\Gamma(s-a+\beta+1)}{\Gamma(b+s+1)\Gamma(b-s)\Gamma(s-a+1)\Gamma(a-\beta+s+1)}\tag{6}$$

and d_n^2 is the norm factor of DRP

$$\begin{aligned}d_n^2&=\frac{\Gamma(\alpha+n+1)\Gamma(\beta+n+1)\Gamma(a+b+\alpha+n+1)}{(\alpha+\beta+2n+1)\Gamma(n+1)\Gamma(b-a-n)}\\ &\quad\times\frac{\Gamma(b-a+\alpha+\beta+n+1)}{\Gamma(\alpha+\beta+n+1)\Gamma(a+b-n-\beta)}.\end{aligned}\tag{7}$$

Noting that the weighted version of DRP can be useful. The nth degree of the proposed method is given by

$$\hat{\mathcal{R}}_n^{\left(\substack{\alpha,\beta\\a,b}\right)}(s;N)=\mathcal{R}_n^{\left(\substack{\alpha,\beta\\a,b}\right)}(s;N)\sqrt{\frac{\rho(s)}{d_n^2}\cdot\Delta x\left(s-\frac{1}{2}\right)}.\tag{8}$$

For convenience, we will use the simplified notation $\hat{\mathcal{R}}_n^{\left(\substack{\alpha,\beta\\a,b}\right)}(s;N)=\hat{\mathcal{R}}_n(s)$ with $b=a+N$ in the following text.

The state-of-the-art methods of DRP computation are in Appendix A.

III. THE PROPOSED METHODOLOGY

This section presents the proposed methodology for computing DRPs. We call it improved stabilization (ImSt). The weighted Racah polynomials are defined

$$\begin{aligned}\hat{\mathcal{R}}_n&(s)\\ &=\sqrt{\frac{\Gamma\left(a+N+\alpha-s\right)\Gamma\left(a+N+\alpha+s+1\right)\Gamma\left(N-n\right)}{\Gamma\left(a+N-s\right)\Gamma\left(a+N+s+1\right)\Gamma\left(N+\alpha+1+n\right)}}\\ &\quad\times\sqrt{\frac{(\alpha+1+2n)\left(2s+1\right)\Gamma\left(2a+N-n\right)}{\Gamma\left(2a+N+\alpha+n+1\right)}}\;(-N+1)_n\\ &\quad\times(2a+N+\alpha+1)_n\,{}_4F_3\left(\begin{matrix}-n,a-s,a+s+1,\alpha+n+1\\ 1,2a+N+\alpha+1,-N+1\end{matrix}\,\middle|\,1\right).\end{aligned}\tag{9}$$

The DRP matrix is partitioned into two parts. They are shown in FIGURE 1 as Part 1 and Part 2. In the following subsections, the detailed steps are given. First of all, we must compute initial values.

A. THE FIRST INITIAL VALUE

The selection of the first initial value, specifically its location and how it is computed, is considered to be crucial because the entire values of the polynomial rely on this initial value. The computation of the initial value in the existing algorithms limits the ability to compute the entire values of DRPs. For example, in [35], the formula for computation of the first initial value is

$$\begin{aligned}\hat{\mathcal{R}}_0&(a)\\ &=\sqrt{(2a+1)F},\\ F&\\ &=\frac{\Gamma(2a+1)\Gamma(\alpha+\beta+2)\Gamma(b-a+\alpha)\Gamma(a+b-\beta)}{\Gamma(a+b+1)\Gamma(\alpha+1)\Gamma(2a+1-\beta)\Gamma(-a+b+\alpha+\beta+1)}.\end{aligned}\tag{10}$$

This formula (Equation (10)) is incomputable for a wide range of parameter values a, α, and β. Thus, in the proposed algorithm, we begin the computation at the last value of the first row, i.e. at $s=a+N-1$ as follows

$$\begin{aligned}\hat{\mathcal{R}}_0&(N-1+a)\\ &=\sqrt{\frac{\Gamma(\alpha+\beta+2)\Gamma(2a+N)\Gamma(\beta+N)\Gamma(2a+2N+\alpha)}{\Gamma(2a+2N-1)\Gamma(\beta+1)\Gamma(\alpha+\beta+N+1)\Gamma(2a+N+\alpha+1)}},\end{aligned}\tag{11}$$

however, the Gamma function ($\Gamma(\cdot)$) make this equation incomputable for high parameter values. To fix this issue, we rewrote Equation (11)

$$\begin{aligned}Y&=\psi(\alpha+\beta+2)+\psi(2a+N)+\psi(\beta+N)\\ &\quad+\psi(2a+2N+\alpha)-(\psi(2a+2N-1)+\psi(\beta+1)\\ &\quad+\psi(\alpha+\beta+N+1)+\psi(2a+N+\alpha+1))\\ &\hat{\mathcal{R}}_0(a+N-1)=\exp(Y/2)\end{aligned}\tag{12}$$

where $\psi(\cdot)$ represents the logarithmic gamma function: $\psi(x)=\log(\Gamma(x))$, $\log(\cdot)$ is natural logarithm. Using (12) the first initial value $\hat{\mathcal{R}}_0(N-1+a)$ is computable for a wide range of the DRP parameters as shown in FIGURE 2b in comparison with values $\hat{\mathcal{R}}_0(a)$ in FIGURE 2a.

B. THE INITIAL SETS

After computing the first initial value, the initial sets in the first two rows $\hat{\mathcal{R}}_0(s)$ and $\hat{\mathcal{R}}_1(s)$ are computed by the two-term recurrence relation. These initial sets will be used for computation of the remaining coefficients of DRPs, i.e. the coefficients in Part 1 and indirectly in Part 2.

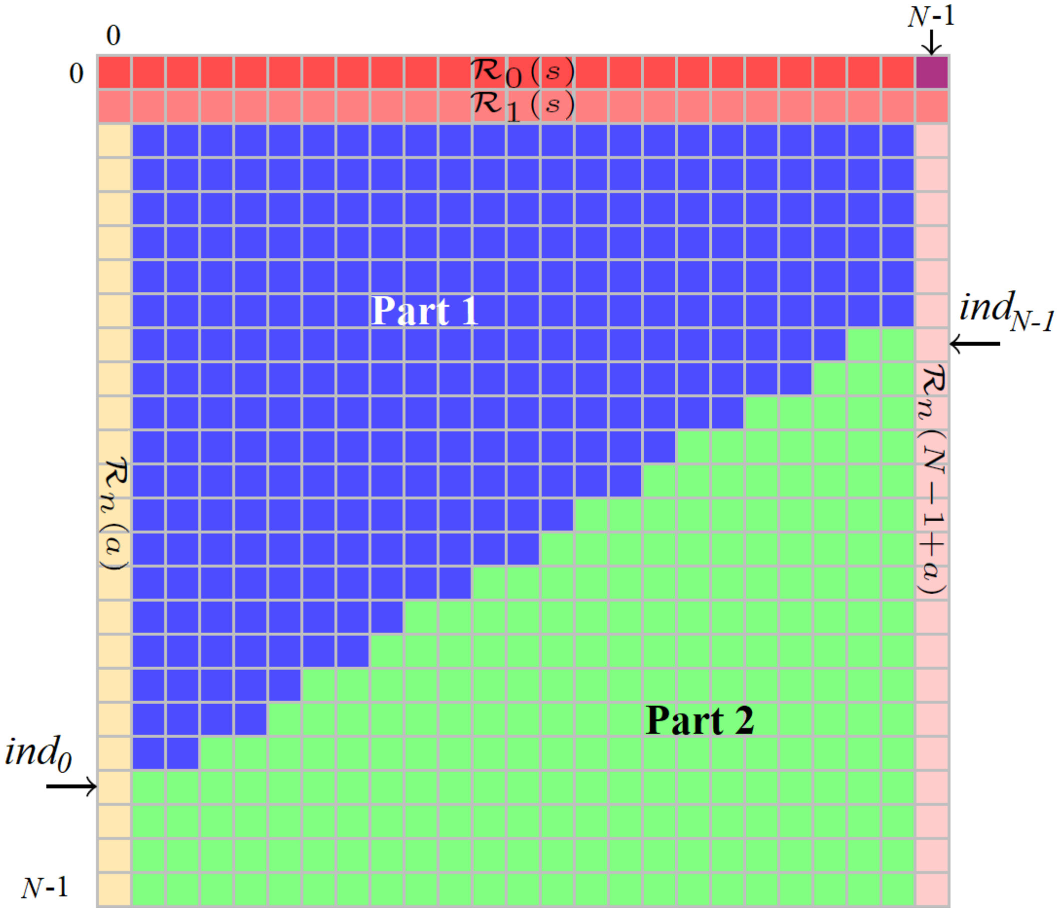
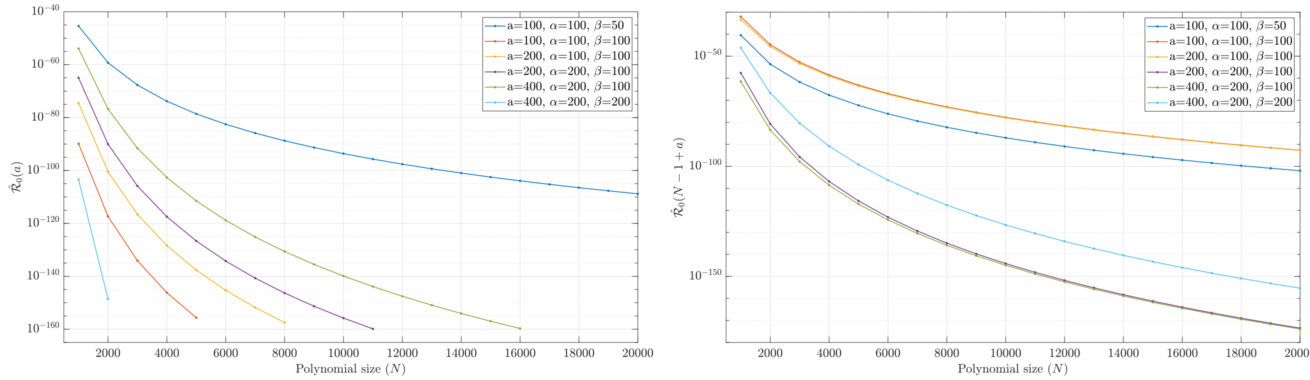


FIGURE 1. The matrix of DRPs. The matrix indices are the degree n vertically and the coordinate i, where $i = s - a$ horizontally. The parameters $a = 6$, $b = 31$, $\alpha = 13$, $\beta = 8$ were chosen as an example.



(a)

(b)

FIGURE 2. The plot of the initial value (a) $\hat{\mathcal{R}}_0(a)$, and (b) $\hat{\mathcal{R}}_0(N-1+a)$.

The values of the coefficients $\hat{\mathcal{R}}_0(s)$ in the first row are calculated

$$\hat{\mathcal{R}}_0(s) = \sqrt{\frac{(2s+1)\,(a-\beta+s+1)\,(b+s+1)\,(b+\alpha-s-1)\,(a-s-1)}{(a+s+1)\,(b+\alpha+s+1)\,(a-\beta-s-1)\,(2s+3)\,(b-s-1)}} \times \hat{\mathcal{R}}_0(s+1)\,,\ \ s = a+N-2, a+N-3, \ldots, a. \tag{13}$$

After computing the values of $\hat{\mathcal{R}}_0(s)$, the values of $\hat{\mathcal{R}}_1(s)$ in the second row are computed using the previously computed coefficients

$$\hat{\mathcal{R}}_1(s) = -\Big(\Big((-a+b-1)\alpha+b^2-s^2-a-s-1\Big)\,\beta + (a^2-s^2+b-s-1)\alpha + a^2+b^2-2(s^2+s)-1\Big)$$

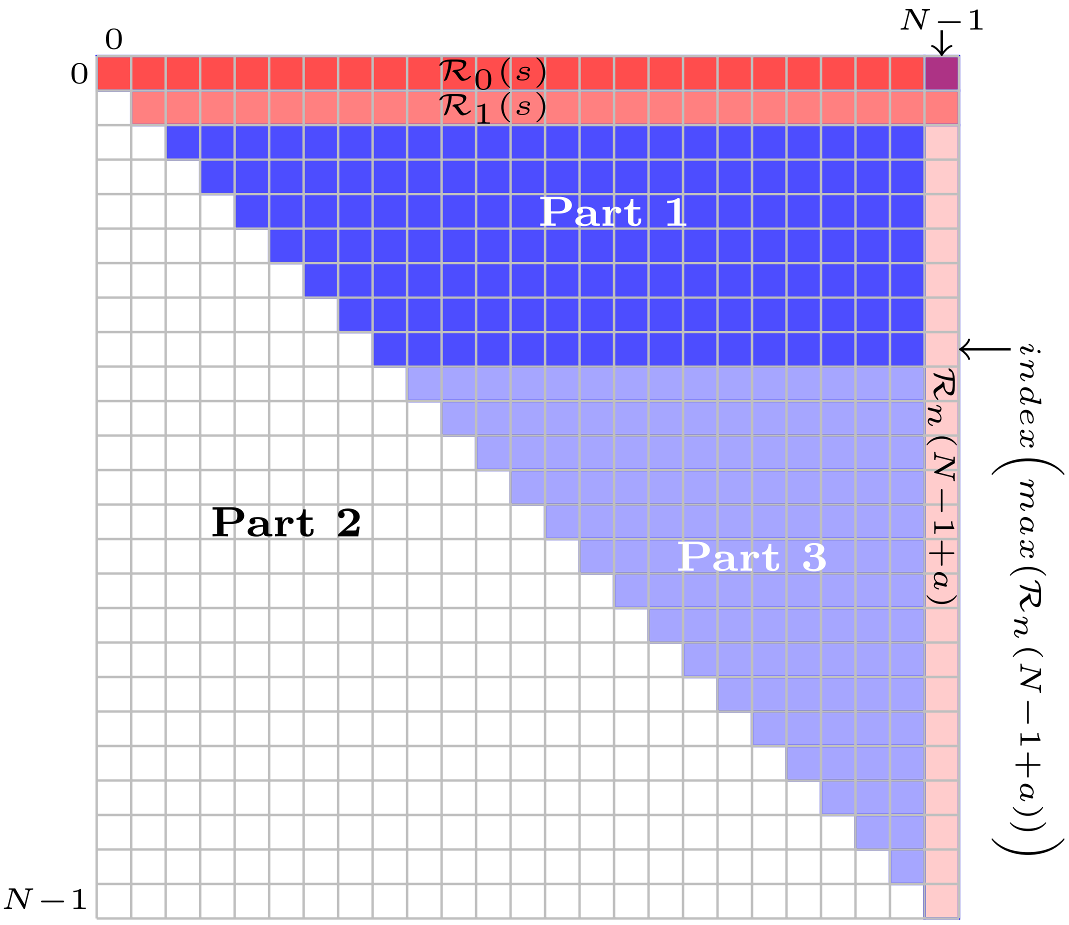


FIGURE 3. The matrix of DRPs for the case $a = \alpha = \beta = 0$.

$$\times\sqrt{\frac{\alpha+\beta+3}{(a-b+1)(a+b-1-\beta)(\alpha+1)(\beta+1)(-b-\alpha+a-\beta-1)}}$$
$$\times\sqrt{\frac{1}{(b+\alpha+a+1)}}\,\hat{\mathcal{R}}_1(s+1)\,,\ s=a,a+1,\ldots,a+N-1. \tag{14}$$

C. THE CONTROLLING INDICES IN THE FIRST AND LAST COLUMNS

To control the stability of the computation of the DRP coefficients, we present a controlling indices that are used to stabilize the computation of the coefficients. We first compute the coefficients $\hat{\mathcal{R}}_n(a)$ and $\hat{\mathcal{R}}_n(a+N-1)$ in the first and last columns. Then, the location, where the peak values occur, are found. To compute the coefficients for $\hat{\mathcal{R}}_n(a)$, the two-term recurrence relation is used

$$\hat{\mathcal{R}}_{n+1}(a)$$
$$=-\sqrt{\frac{(N-n-1)\,(\alpha+\beta+2n+3)\,(\alpha+\beta+n+1)}{(2a+N-\beta-n-1)\,(\alpha+\beta+2\,n+1)\,(\alpha+n+1)}}$$
$$\times\sqrt{\frac{(\beta+n+1)\,(2a+N+\alpha+n+1)}{(N+\alpha+\beta+n+1)\,(n+1)}}\,\hat{\mathcal{R}}_n(a)\,,$$
$$n=1,2,\ldots,N-2. \tag{15}$$

Also, we present a new two-term recurrence relation to compute the coefficients of $\hat{\mathcal{R}}_n(a+N-1)$ in the last column

$$\hat{\mathcal{R}}_{n+1}(a+N-1)$$
$$=\sqrt{\frac{(N-n-1)\,(\alpha+\beta+2n+3)\,(\alpha+\beta+n+1)}{(n+1)\,(2a+N+\alpha+n+1)\,(\alpha+\beta+2n+1)}}$$
$$\times\sqrt{\frac{(\alpha+n+1)\,(2\,a+N-\beta-n-1)}{(N+\alpha+\beta+n+1)\,(n+1)}}\,\hat{\mathcal{R}}_n(a+N-1)\,,$$
$$n=1,2,\ldots,N-2. \tag{16}$$

The peak value at the first column $s=a$ is the index

$$ind_0=\underset{n=0,1,\ldots N-1}{\arg\max}\ \hat{\mathcal{R}}_n(a)\,, \tag{17}$$

while the peak value at the last column $s=a+N-1$ is the index

$$ind_{N-1}=\underset{n=0,1,\ldots N-1}{\arg\max}\ \hat{\mathcal{R}}_n(a+N-1)\,. \tag{18}$$

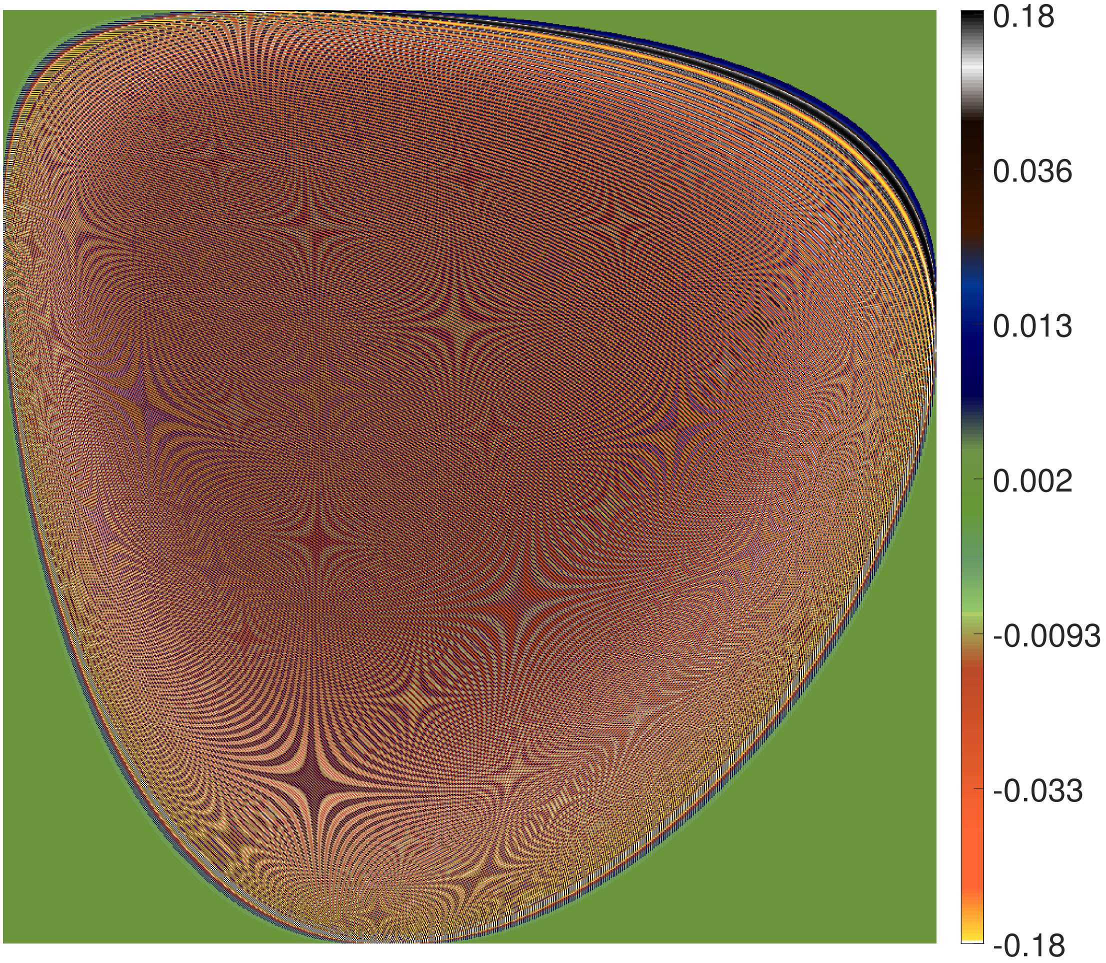


FIGURE 4. The Racah polynomials for $a = 800$, $b = 1800$, $\alpha = 400$, and $\beta = 100$.

The border between Part 1 and Part 2 is the straight line segment connecting the points with the coordinates (a, ind_0) and $(a + N - 1, ind_{N-1})$.

D. THE COEFFICIENTS FOR PART 1

The coefficients in Part 1 are computed using the three-term recurrence algorithm in the n-direction

$$\hat{\mathcal{R}}_n(s) = \Theta_1\,\hat{\mathcal{R}}_{n-1}(s) + \Theta_2\,\hat{\mathcal{R}}_{n-2}(s)\,, \tag{19}$$

where

$$\Theta_1 = \frac{\Theta_{11}}{\Theta_0}\sqrt{\Theta_{12}}, \qquad \Theta_2 = \frac{\Theta_{21}}{\Theta_0}\sqrt{\Theta_{12}\Theta_{22}} \tag{20}$$

and

$$\Theta_0 = \frac{n\,(\alpha+\beta+n)}{(\alpha+\beta+2n-1)\,(\alpha+\beta+2n)} \tag{21}$$

$$\begin{aligned}\Theta_{11} &= s(s+1) - \frac{1}{4}\left(a^2+b^2+(a-\beta)^2+(b+\alpha)^2-2\right) \\ &\quad + \frac{1}{8}\left((\alpha+\beta+2n-2)\,(\alpha+\beta+2n)\right) \\ &\quad - \frac{1}{2}\left(\frac{\left(\beta^2-\alpha^2\right)\left((b+\alpha/2)^2-(a-\beta/2)^2\right)}{(\alpha+\beta+2n-2)\,(\alpha+\beta+2n)}\right)\end{aligned} \tag{22}$$

$$\begin{aligned}\Theta_{21} &= -\frac{(\alpha+n-1)\,(\beta+n-1)}{(\alpha+\beta+2n-2)\,(\alpha+\beta+2n-1)} \\ &\quad \times \left(\left(a+b+\frac{\alpha-\beta}{2}\right)^2-\left(n-1+\frac{\alpha+\beta}{2}\right)^2\right) \\ &\quad \times \left(\left(b-a+\frac{\alpha+\beta}{2}\right)^2-\left(n-1+\frac{\alpha+\beta}{2}\right)^2\right)\end{aligned} \tag{23}$$

$$\begin{aligned}\Theta_{12} &= \frac{n\,(\alpha+\beta+n)\,(\alpha+\beta+2n+1)}{(\alpha+n)\,(\beta+n)\,(\alpha+\beta+2n-1)\,(a-b-\alpha-\beta-n)} \\ &\quad \times \frac{1}{(a+b+\alpha+n)\,(a+b-\beta-n)\,(a-b+n)}\end{aligned} \tag{24}$$

$$\begin{aligned}\Theta_{22} &= \frac{(n-1)\,(\alpha+\beta+n-1)\,(\alpha+\beta+2n-1)}{(\alpha+n-1)\,(\beta+n-1)\,(\alpha+\beta+2n-3)\,(a-b-\alpha-\beta-n+1)} \\ &\quad \times \frac{1}{(a+b+\alpha+n-1)\,(a+b-\beta-n+1)\,(a-b+n-1)}.\end{aligned} \tag{25}$$

The border between Part 1 and Part 2 is created by the index ind_s defined as

$$ind_s = \text{round}\left(ind_0 + \frac{ind_{N-1} - ind_0}{N-1}(s-a) \right), \tag{26}$$

where round$(\cdot)$ is rounding to the nearest integer, i.e. round$(x) = \lfloor x + 0.5 \rfloor$,[1] while the indices ind_0 and ind_{N-1} are defined in Eqs. (17) and (18) respectively. The recurrence algorithm is applied for $s = a, a+1, \ldots, a+N-2$ and $n = 2, 3, \ldots, ind_s - 1$.

E. THE COEFFICIENTS FOR PART 2

The coefficients in Part 2 are computed using the same three term recurrence algorithm in the n-direction as in (19). After computation of each value, the following stabilizing condition is applied for each order n

$$\hat{\mathcal{R}}_n(s) = 0 \text{ if } \left|\hat{\mathcal{R}}_n(s)\right| < 10^{-5} \wedge \left|\hat{\mathcal{R}}_n(s)\right| > \left|\hat{\mathcal{R}}_{n-1}(s)\right|, \tag{27}$$

where $\wedge$ means the logical AND operation. The recurrence algorithm in Part 2 is applied in the range $s = a, a+1, \ldots, a+N-2$ and $n = ind_s, ind_s+1, \ldots, N-1$.

F. SPECIAL CASE OF RACAH POLYNOMIALS

In this section, a special case of DRPs is presented. The parameter β affects on the energy compaction as its value becomes larger than 0. So, the case $\hat{\mathcal{R}}_n^{\binom{0,0}{0,b}}(s)$, where $a = \alpha = \beta = 0$, has special significance. Then, we can derive the the following symmetry relation

$$\hat{\mathcal{R}}_s^{\binom{0,0}{0,b}}(n) = (-1)^{(s-n)}\hat{\mathcal{R}}_n^{\binom{0,0}{0,b}}(s), \tag{28}$$

see Appendix B for the proof.

Thus, from (28), we can compute the coefficients for 50% and the rest of the coefficients using the symmetry relation. In other words, the coefficients are computed in the range $n = 0, 1, \ldots, N-1$ and $s = n, n+1, \ldots, N-1$ (Parts 1 and 3). The rest of the coefficients are computed using the symmetry relation (Part 2) as shown in FIGURE 3.

Then Eq. (19) becomes

$$\hat{\mathcal{R}}_n(s) = \Theta_{10}\,\hat{\mathcal{R}}_{n-1}(s) + \Theta_{20}\,\hat{\mathcal{R}}_{n-2}(s), \tag{29}$$

where

$$\Theta_{10} = \frac{\left(2s(s+1)+n\,(n-1)\,-N^2+1\right)\sqrt{4n^2-1}}{n(N-n)(N+n)} \tag{30}$$

$$\Theta_{20} = -\frac{(n-1)\,(N-n+1)\,(N+n-1)}{n\,(N-n)\,(N+n)}\sqrt{\frac{2n+1}{2n-3}}. \tag{31}$$

G. IMPLEMENTATION OF THE PROPOSED ALGORITHM

Here, the implementation is described by pseudo codes. The pseudo code of the proposed algorithm for the general case is presented in Algorithm 1, while the pseudo code for the special case ($a = \alpha = \beta = 0$) is given in Algorithm 2.

Algorithm 1 Computation of the DRP Coefficients Using the Improved Stabilization

```
Input: $Ord, a, b, \alpha, \beta$
    $Ord$ is the maximum degree of DRP, $Ord < b - a$.
    $a, b, \alpha, \beta$ represents the parameter of DRP.
Output: $\hat{\mathcal{R}}_n(s)$
$N \leftarrow b - a$                          ▷ $N$ represents the size of DRP
$\Theta = 10^{-5}$                       ▷ Threshold for stabilizing condition
Compute $\hat{\mathcal{R}}_0(N-1+a)$ using (12)
for $s = a+N-2 : a$ do
    Compute $\hat{\mathcal{R}}_0(s)$ using (13)
end for
for $s = a : a+N-1$ do
    Compute $\hat{\mathcal{R}}_1(s)$ using (14)
end for
for $n = 1 : Ord-1$ do
    Compute $\hat{\mathcal{R}}_n(a)$ using (15)
    Compute $\hat{\mathcal{R}}_n(N-1+a)$ using (16)
end for
$ind_0 \leftarrow$ index($max(\hat{\mathcal{R}}_n(a))$)       ▷ Find the index of the
    maximum value in $\hat{\mathcal{R}}_n(a)$
$ind_{N-1} \leftarrow$ index($max(\hat{\mathcal{R}}_n(a+N-1))$)      ▷ Find the
    index of the maximum value in $\hat{\mathcal{R}}_n(a+N-1)$
for $s = a : a+N-1$ do                                   ▷ Part 1
    Compute $ind_s$ using (26)          ▷ Border between Part
    1 and Part 2
    for $n = 2 : ind_s - 1$ do
        Compute $\hat{\mathcal{R}}_n(s)$ using (19)
    end for
end for
for $s = a : a+N-1$ do                                   ▷ Part 2
    Compute $ind_s$ using (26)          ▷ Border between Part
    1 and Part 2
    for $n = ind_s : Ord$ do
        Compute $\hat{\mathcal{R}}_n(s)$ using (19)
        if $\left|\hat{\mathcal{R}}_n(s)\right| < \Theta \wedge \left|\hat{\mathcal{R}}_n(s)\right| > \left|\hat{\mathcal{R}}_{n-1}(s)\right|$ then
            $\hat{\mathcal{R}}_n(s) = 0$
            Exit inner loop
        end if
    end for
end for
```

The values of the Racah polynomials for $a = 800$, $b = 1800$, $\alpha = 400$, and $\beta = 100$ (i.e. $N = b - a = 1000$) in artificial colors are shown in FIGURE 4.

IV. APPLICATIONS

This section shows some applications of the Racah polynomials. We also compared the proposed algorithm for DRP with

[1] $\lfloor\cdot\rfloor$ is the function floor, while $\lceil\cdot\rceil$ is the function ceiling.

Algorithm 2 Computation of the DRP Coefficients Using the Improved Stabilization for the Special Case $a = \alpha = \beta = 0$

```
Input: $N$, $Ord$
    $N$ represents the size of the DRP,
    $Ord$ is the maximum degree of the DRP, $Ord < N$.
Output: $\hat{\mathcal{R}}_n(s)$
$\Theta = 10^{-5}$                    ▷ Threshold for stabilizing condition
$\hat{\mathcal{R}}_0(N-1) \leftarrow \frac{\sqrt{2N-1}}{N}$
for $s = N-2 : 0$ do
    $\hat{\mathcal{R}}_0(s) \leftarrow \sqrt{\frac{2s+1}{2s+3}}\hat{\mathcal{R}}_0(s+1)$
end for
for $s = 1 : Ord$ do
    $\hat{\mathcal{R}}_s(0) \leftarrow (-1)^s\hat{\mathcal{R}}_0(s)$
end for
for $s = 0 : N-1$ do
    $\hat{\mathcal{R}}_1(s) \leftarrow -\frac{(N^2-2s^2-s2-1)\sqrt{3}}{N^2-1}\hat{\mathcal{R}}_0(s)$
end for
for $s = 2 : Ord$ do
    $\hat{\mathcal{R}}_s(1) \leftarrow (-1)^{s-1}\hat{\mathcal{R}}_1(s)$
end for
for $n = 1 : Ord-1$ do
    $\hat{\mathcal{R}}_{n+1}(N-1) \leftarrow \frac{N-n-1}{N+n+1}\sqrt{\frac{2n+3}{2n+1}}\hat{\mathcal{R}}_n(N-1)$
end for
$ind_{N-1} \leftarrow$ index($max(\hat{\mathcal{R}}_n(N-1))$)    ▷ position of maximum in $\hat{\mathcal{R}}_n(N-1)$
for $n = 2 : ind_{N-1} - 1$ do                          ▷ Part 1
    for $s = n : N-1$ do
        Compute $\hat{\mathcal{R}}_n(s)$ using (29)
    end for
end for
for $n = ind_{N-1} : Ord$ do                             ▷ Part 3
    for $s = n : N-1$ do
        Compute $\hat{\mathcal{R}}_n(s)$ using (29)
        if $\left|\hat{\mathcal{R}}_n(s)\right| < \Theta \wedge \left|\hat{\mathcal{R}}_n(s)\right| > \left|\hat{\mathcal{R}}_{n-1}(s)\right|$ then
            $\hat{\mathcal{R}}_n(s) = 0$
            Exit inner loop
        end if
    end for
end for
for $s = 3 : Ord$ do                                     ▷ Part 2
    for $n = 2 : s-1$ do
        $\hat{\mathcal{R}}_s(n) \leftarrow (-1)^{s-n}\hat{\mathcal{R}}_n(s)$
    end for
end for
```

the existing algorithms. Three evaluation procedures are carried out to check the performance of the proposed algorithm which are: maximum size generated, computational cost, and signal reconstruction. The experiments were carried out using MATLAB version 2019b on the computer with the processor Intel(R) Core(TM) i9-7940X CPU with frequency 3.10GHz, memory 32 GB, and with 64-bit Windows 10 Pro.

A. THE DEFINITION OF DISCRETE RACAH MOMENTS

The discrete Racah moments (DRMs) represent the projection of a signal (e.g. speech or images) on the basis of DRPs. The computation of the DRMs (ϕ_{nm}) for a 2D signal, $f(x, y)$, with a size of $N_1 \times N_2$ is performed by

$$\phi_{nm} = \sum_{x=0}^{N_1-1}\sum_{y=0}^{N_2-1} f(x,y)\hat{\mathcal{R}}_n^{\binom{\alpha_1,\beta_1}{a_1,b_1}}(x;N_1)\,\hat{\mathcal{R}}_m^{\binom{\alpha_2,\beta_2}{a_2,b_2}}(y;N_2)$$
$$n = 0, 1, \ldots, N_1-1;\quad m = 0, 1, \ldots, N_2-1. \quad (32)$$

The reconstruction of the 2D signal (image) from the Racah domain (moment) into the spatial domain can be carried out by

$$\hat{f}(x,y) = \sum_{n=0}^{N_1-1}\sum_{m=0}^{N_2-1} \phi_{nm}\hat{\mathcal{R}}_n^{\binom{\alpha_1,\beta_1}{a_1,b_1}}(x;N_1)\,\hat{\mathcal{R}}_n^{\binom{\alpha_2,\beta_2}{a_2,b_2}}(y;N_2)$$
$$x = 0, 1, \ldots, N_1-1;\quad y = 0, 1, \ldots, N_2-1. \quad (33)$$

B. MAXIMUM DEGREE

We searched the maximum signal size N, where the orthogonality error E is less than 0.001. We changed the parameter values a, α and β as ratios of N. It has an advantage, that the pattern of non-zero values looks similar and does not moved. The orthogonality error is defined

$$E = \max_{n,m=0,1,\ldots,N-1}\left|\sum_{s=a}^{b-1}\hat{\mathcal{R}}_n(s)\,\hat{\mathcal{R}}_m(s) - \delta_{nm}\right|. \quad (34)$$

The results are shown in TABLE 1.

In the first column, when $a = 0$, $\alpha = 0$, and $\beta = 0$, Algorithm 2 is used, in the other cases, Algorithm 1 is used. The limit $N = 56000$ is not the limit of our algorithm, it is the memory limit of our computer. We are not able to check the orthogonality error because of the "Out of memory" error.

Another problem is the long computation of GSOP. In the case $a = \alpha = \beta = 0$ and $N = 56000$, the error of orthogonality E was also under the threshold 0.001, but the computation of GSOP took 15 days. We cannot test the precise maximum size when the computing times are such long. That is why we added another criterion, the result must be available in the time less than one hour. The sizes of GSOP in the first two columns are limited by this condition. However, to get more explanation about the relation between the initial value calculation and the maximum size of the polynomial, the limit is calculated empirically as shown in FIGURE 6. Different values of DRP parameters have been considered by increasing the size until we reached 10^6 and the initial value is computed until this size. From FIGURE 6, most of the tested parameters are computable up to $N = 10^6$. However, two cases shows less size. The first case when $a = 200, \alpha = 200\ and \beta = 100$, the limit of the algorithm

TABLE 1. Maximum sizes N of the Racah polynomials reachable by various algorithms. Usually, the limit is the algorithm precision, i.e. the orthogonality error $E \leq 10^{-3}$, † the limit is computing time ≤1 hour, ‡ the limit is computer memory 32 GB.

	$a=0$ $\alpha=0$ $\beta=0$	$a=\lceil N/10000+0.5\rceil$ $\alpha=N/10000$ $\beta=N/10000$	$a=\lfloor N/4+0.5\rfloor$ $\alpha=\lfloor N/8+0.5\rfloor$ $\beta=\lfloor N/16+0.5\rfloor$	$a=\lfloor N/2+0.5\rfloor$ $\alpha=\lfloor N/2+0.5\rfloor$ $\beta=\lfloor N/4+0.5\rfloor$
Zhu n	23	25	37	32
Zhu s	21	26	35	32
Daoui	1165	4	65	53
GSOP	9649†	9834†	1075	504
ImSt	56000‡	25580	6770	4659

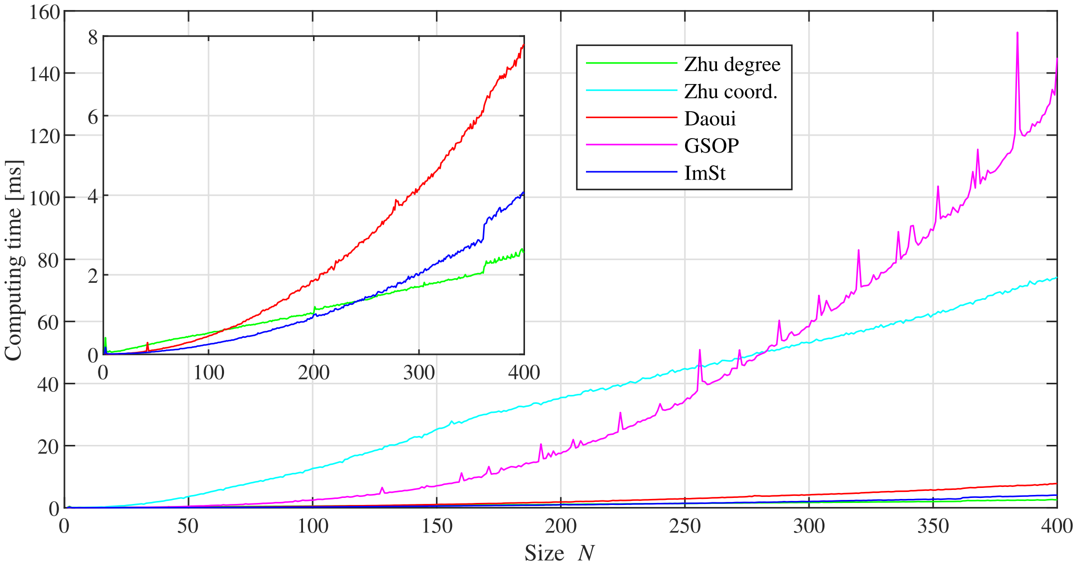


FIGURE 5. The computing times of the Racah polynomials for $n = N - 1$, $a = \max(N/4, 1)$, $b = a + N$, $\alpha = N/8$, and $\beta = N/16$.

is $N = 630000$. And the second case when $a = 200$, $\alpha = 200 and \beta = 200$, the maximum size of the polynomial reached is 970000.

C. COMPUTING TIME

In this section, the computing time has been tested. There is one problem, the maximum sizes of Daoui and particularly Zhu algorithms are low, that sufficient analysis of computing times is not possible. Therefore, we tested these algorithms even if the error of orthogonality was higher than our threshold. We choose these values of the parameters: $n = N - 1$, $a = \max(N/4, 1)$, $b = a + N$, $\alpha = N/8$, and $\beta = N/16$. We repeated each computation ten times, and took average time. The results are in FIGURE 5.

The fastest algorithm is Zhu's recurrence over the degree, our algorithm ImSt is based on the similar principle, it is only a little bit slower. Daoui's algorithm is a little bit slower than ours and Zhu's recurrence over the coordinate is significantly slower, but it has still computing complexity $\mathcal{O}(N^2)$, only with worse constant. The computing complexity $\mathcal{O}(N^3)$ of GSOP is clearly visible in the graph; from beginning, it is fast, but it cannot be used for high values of polynomial size (N). The detailed computational complexity analysis of the proposed algorithm is as follows:

1) The computation of the initial value has a constant time complexity, denoted as $\mathcal{O}(1)$.
2) The first and second loops used to compute the initial set ($\hat{\mathcal{R}}_0(s)$) for N coefficients. The time complexity of this loop is $\mathcal{O}(N)$.
3) The third loop runs from $n = 1$ to $Ord - 1$, iterating $Ord - 1$ times, which has operations for computing $\hat{\mathcal{R}}_n(a)$ and $\hat{\mathcal{R}}_n(N - 1 + a)$. The time complexity of this loop is $\mathcal{O}(Ord)$.
4) The forth loop (starts at step 16 and ends at step 21 in Algorithm 1) computes the coefficients of Part 1. The computational complexity for the coefficients is $\sim \frac{(ind_{N-1}\cdot N+(ind_0-ind_{N-1})\cdot N/2)}{N^2}$ $\mathcal{O}(N \cdot Ord)$.

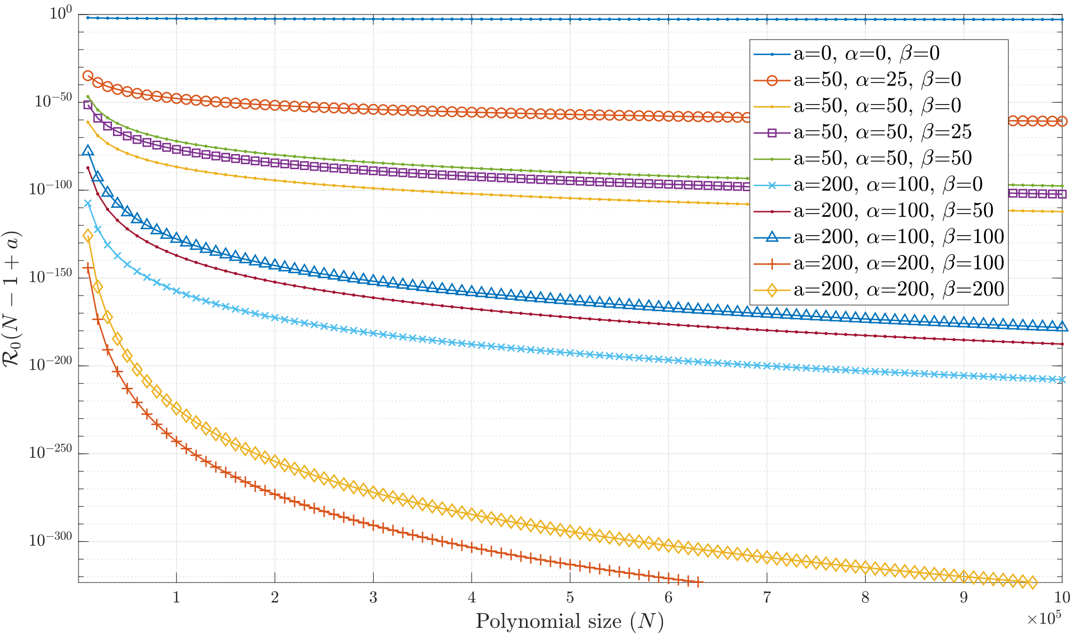


FIGURE 6. **The initial value plot for different values of DRP parameters with maximum $N = 10 \times 10^5 = 10^6$.**

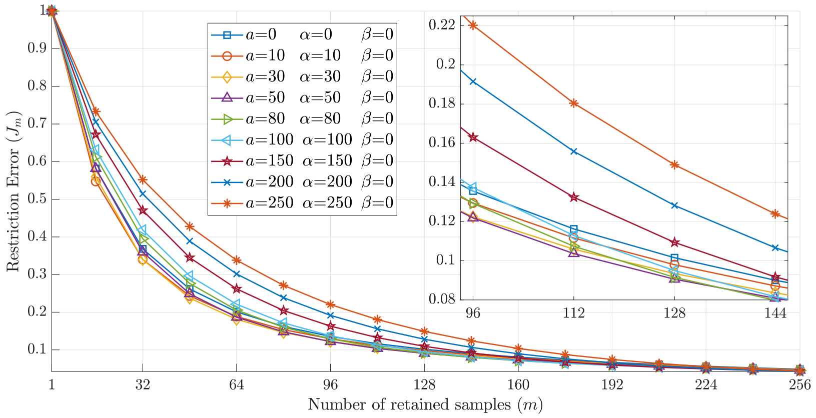


FIGURE 7. **Results of the restriction error for different values of Racah parameters ($a = \alpha$ and $\beta = 0$).**

5) The fifth loop (starts at step 22 and ends at step 31 in Algorithm 1) computes the coefficients of Part 2. The computational complexity for the coefficients is $\sim \frac{((N-ind_0)\cdot N+(ind_0-ind_{N-1})\cdot N/2)}{N^2}$ $\mathcal{O}(N \cdot Ord)$.
6) T o sum up, the overal computational complexity is $\mathcal{O}(N \cdot Ord)$. When the order (Ord) equals to N, the computational complexity is $\mathcal{O}(N^2)$.

D. RESTRICTION ERROR ANALYSIS

The distribution of moments is diverse from each other based on the discrete transforms [36]. To correctly reconstruct the signal information, the sequence of moments is important and should be recognized. Therefore, the moment energy distribution of DRP is examined first; then the signal reconstruction analysis is performed. To acquire the distribution of moments, the procedure presented by Jian [37] is followed. The procedure is given in Algorithm 3.

The covariance matrix Σ is used as the signal. Then the matrix multiplication $R \times \Sigma \times R^T$ can be used for moment computation, R is the matrix of Racah polynomials, $R_{n,s-a} = \hat{\mathcal{R}}_n(s)$. For the covariance coefficients, three values are used, $\rho = 0.90$, $\rho = 0.95$, and $\rho = 0.98$ with length $N = 16$; then, the results are reported in TABLE 2. It can observed from TABLE 2 that the maximum value of DRP is found at $\ell = 0$ and the values are descendingly ordered. It declares that the DRP moment order used for signal reconstruction is $n = 0, 1, \ldots, N-1$.

The energy compaction property of the discrete transformation based on orthogonal polynomials is considered one of the important properties. It is the fraction of the number of coefficients that reflect most of the signal energy to the

TABLE 2. The values of transform coefficient for different values of covariance coefficients.

	$\rho=0.9$				$\rho=0.95$				$\rho=0.98$			
	$a=0$	$a=10$	$a=30$	$a=50$	$a=0$	$a=10$	$a=30$	$a=50$	$a=0$	$a=10$	$a=30$	$a=50$
	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$	$\alpha=a$
ℓ	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$	$\beta=0$
0	9.159	9.832	2.742	2.249	11.325	12.401	2.923	2.362	12.975	14.407	3.039	2.434
1	2.912	2.856	2.388	2.045	2.232	1.907	2.500	2.125	1.527	0.916	2.568	2.174
2	1.278	1.136	2.074	1.854	0.843	0.612	2.137	1.908	0.532	0.249	2.173	1.941
3	0.702	0.591	1.794	1.675	0.440	0.300	1.822	1.708	0.272	0.120	1.836	1.727
4	0.446	0.366	1.543	1.506	0.273	0.182	1.545	1.522	0.168	0.072	1.543	1.530
5	0.311	0.252	1.313	1.345	0.188	0.124	1.298	1.347	0.115	0.049	1.286	1.347
6	0.233	0.188	1.101	1.190	0.139	0.092	1.072	1.180	0.084	0.036	1.053	1.173
7	0.183	0.147	0.900	1.037	0.109	0.072	0.862	1.018	0.065	0.028	0.838	1.005
8	0.149	0.120	0.707	0.884	0.088	0.059	0.663	0.856	0.053	0.023	0.637	0.839
9	0.125	0.101	0.523	0.725	0.074	0.050	0.477	0.691	0.044	0.020	0.450	0.671
10	0.108	0.088	0.357	0.560	0.063	0.043	0.313	0.523	0.037	0.017	0.287	0.500
11	0.095	0.077	0.224	0.396	0.055	0.038	0.183	0.357	0.032	0.015	0.159	0.334
12	0.085	0.070	0.133	0.250	0.049	0.034	0.096	0.213	0.028	0.013	0.075	0.190
13	0.077	0.063	0.083	0.142	0.044	0.031	0.051	0.108	0.025	0.012	0.031	0.088
14	0.071	0.058	0.062	0.082	0.040	0.028	0.032	0.051	0.023	0.011	0.015	0.033
15	0.066	0.054	0.055	0.058	0.037	0.027	0.027	0.030	0.021	0.010	0.011	0.014

```
Algorithm 3 Find the Moment Order of DRM
Input: $\rho$=covariance coefficient
Output: Degree of DRP.
1: Generate the covariance matrix $\Sigma$ with zero mean and length $N$
       $\Sigma = \begin{bmatrix} 1 & \rho & \cdots & \rho^{N-1} \\ \rho & 1 & & \vdots \\ \vdots & & \ddots & \rho \\ \rho^{N-1} & \cdots & \rho & 1 \end{bmatrix}$   (35)
2: Transform the covariance matrix $\Sigma$ into the domain of the discrete Racah moment ($M$) using
       $M = R \times \Sigma \times R^T,$   (36)
   where $R$ is the DRP matrix.
3: Find the diagonal coefficients $\sigma_\ell^2 = M_{\ell\ell}$ of the discrete Racah moments $M$.
4: Find the order of moments according to the values of the diagonal coefficients.
5: return result
```

total number of coefficients. This characteristic is used to assess a DRP's ability to reconstruct a significant portion of the signal information from a very small number of moment coefficients. To examine the impact of the DRP parameters a, α and β on the energy compaction, the restriction error, $\mathcal{J}_m$, is used as follows [37]

$$\mathcal{J}_m = \frac{\sum\limits_{k=m}^{N-1} \sigma_k^2}{\sum\limits_{k=0}^{N-1} \sigma_k^2}; \quad m = 0, 1, 2, \ldots, N-1, \tag{37}$$

where σ_k^2 represents diagonal values of the transform coefficients ordered descendingly. In our case, the coefficients are already ordered, i.e. $k = \ell$. FIGURE 7 shows the restriction error using covariance coefficient ($\rho = 0.95$) with DRP parameters of $a = \alpha$ and $\beta = 0$. From FIGURE 7, it is noticed that the DRP parameters affect the restriction error, which reveals that DRPs with parameters $a = \alpha = 30$ and $\beta = 0$ shows better energy compaction than other parameter values in the range of $m < 96$. However, when $a = \alpha = 50$ and $\beta = 0$ presents better energy compaction compared to other DRP parameters in the range $m > 96$.

FIGURE 8 shows the restriction error of DRP with parameters of a, $\alpha = \{0, a/2\}$ and $\beta = \{0, a/2\}$. It can be observed from FIGURE 8 that DRPs with parameters $a = 50$, $\alpha = 25$ and $\beta = 0$ shows better energy compaction than other parameter values in the range of $m < 96$. However, when $a = 100$, $\alpha = 50$ and $\beta = 0$ presents better energy compaction compared to other DRP parameters in the range $m > 96$.

On the other hand, FIGURE 9 shows the restriction error for DRP with parameters of a, $\alpha = a$, and $\beta = \{0, a/2, a\}$. The best energy compaction occurs at DRP parameters is $a = 50$, $\alpha = 50$, and $\beta = 0$ for the entire range of retained samples m.

E. ANALYSIS OF RECONSTRUCTION ERROR

In this section, the reconstruction error analysis is carried out using real images. The test images, shown in FIGURE 10, are taken from LIVE dataset [38], [39], [40]. The size of the test images are cropped and resized to 512 × 512. Various values of DRP parameters (a, α, β) were considered in the analysis as shown in the following groups :

1) a and $\alpha = \{0, a/2\}$ with $\beta = 0$,
2) a and $\alpha = \{0, a/2\}$ with $\beta = \{0, a/2\}$,

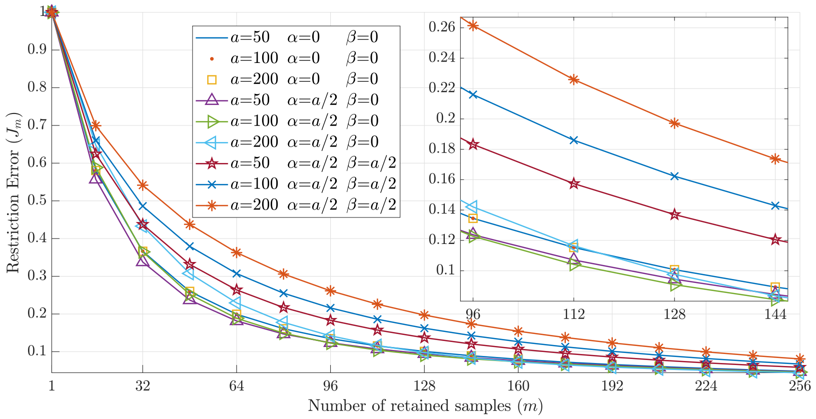


FIGURE 8. Results of the restriction error for different values of Racah parameters (a, $\alpha = \{0, a/2\}$, and $\beta = \{0, a/2\}$).

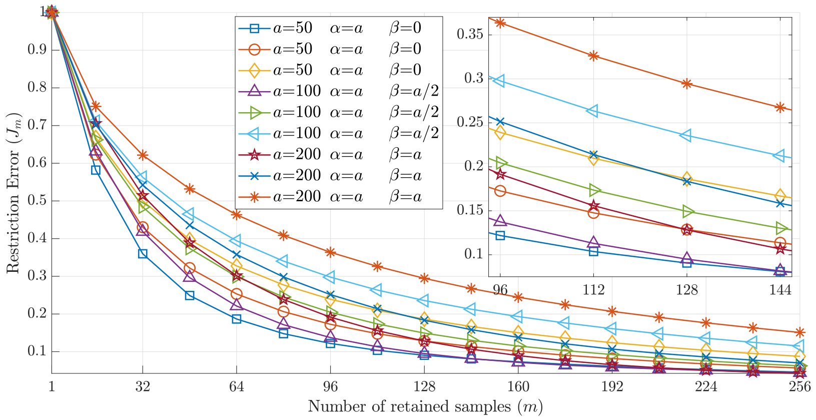


FIGURE 9. Results of the restriction error for different values of Racah parameters (a, $\alpha = a$, and $\beta = \{0, a/2, a\}$).

3) a and $\alpha = a$ with $\beta = \{0, a/2, a\}$.

DRPs (R) are generated first using the proposed sets of parameters. Then, DRMs (M) of the test images are computed. After that, the image is reconstructed using the finite number of calculated moments. The normalized mean square error (NMSE), which compares the input image to the reconstructed version of the image, is then calculated. The NMSE is expressed as

$$NMSE(I, I_r) = \frac{\sum\limits_{x,y} [I(x, y) - I_r(x, y)]^2}{\sum\limits_{x,y} I(x, y)^2}, \tag{38}$$

where I and I_r represent the original grayscale image and the reconstructed grayscale image, respectively. NMSE is defined as the reconstruction error.

First, the reconstruction error analysis is carried out for $\alpha = a$ and $\beta = 0$. The order of moments used to reconstruct the image is varied in the set $\{1, 32, 64, \ldots, 512\}$. The obtained results are depicted in FIGURE 11. The obtained results show that at the moment order of 64, the best NMSE is occurred at DRP parameters of $a = 10, \alpha = a, \beta = 0$ with NMSE of 0.03276. The next three best NMSE are 0.03388, 0.03706, and 0.04627 for DRP parameters $a = 30, \alpha = a, \beta = 0$, $a = 0, \alpha = a, \beta = 0$, and $a = 50, \alpha = a, \beta = 0$, respectively. However, for moment order of 128, the NMSE are 0.0160, 0.0163, 0.01634, and 0.01649 for DRP parameters $a = 80, \alpha = a, \beta = 0$, $a = 50, \alpha = a, \beta = 0$, $a = 100, \alpha = a, \beta = 0$, and $a = 30, \alpha = a, \beta = 0$, respectively.

Moreover, for moment order of 256, the best NMSE is occurred at DRP parameters $a = 10, \alpha = a, \beta = 0$ with NMSE of 0.0052. From these values, the most suitable parameters of the proposed polynomial are determined, and the desired signal can be found by inspecting the lowest NMSE between the reconstructed signal and the original signal. For better inspection, visual reconstruction error and PSNR between the original image and the reconstructed image is computed for different values of DRP parameters as shown in FIGURE 12.

The DRP parameter values in the range (a and $\alpha = \{0, a/2\}$ with $\beta = \{0, a/2\}$) are used to perform the reconstruction error analysis. The same moment orders of the first experiment are used in this experiment. FIGURE 13

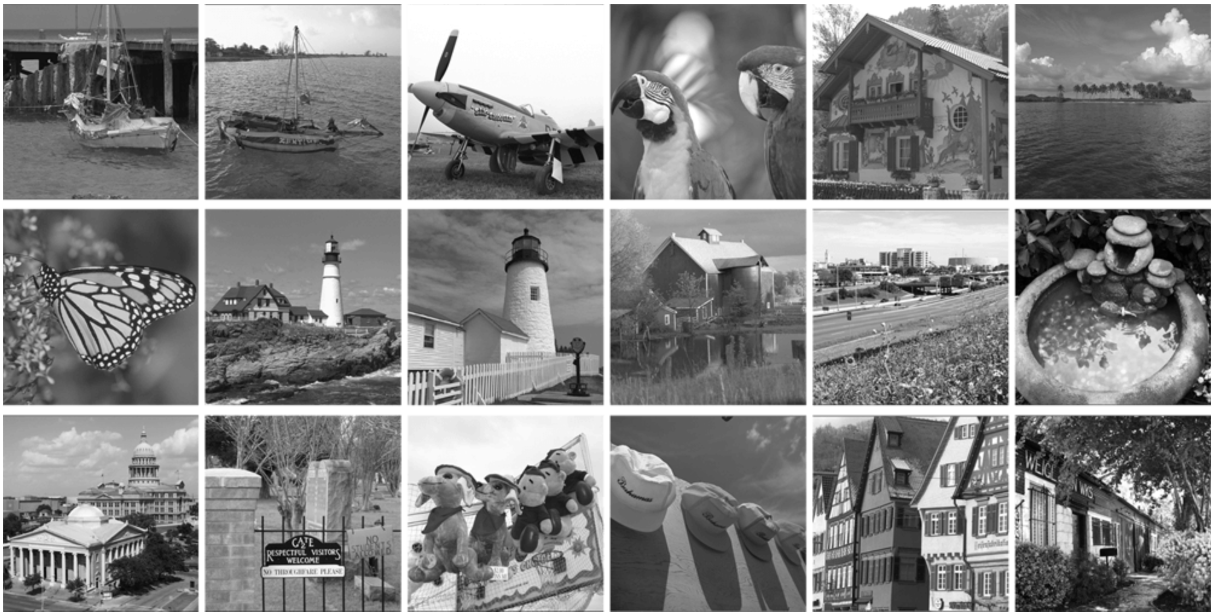


FIGURE 10. Test images used for reconstruction error analysis.

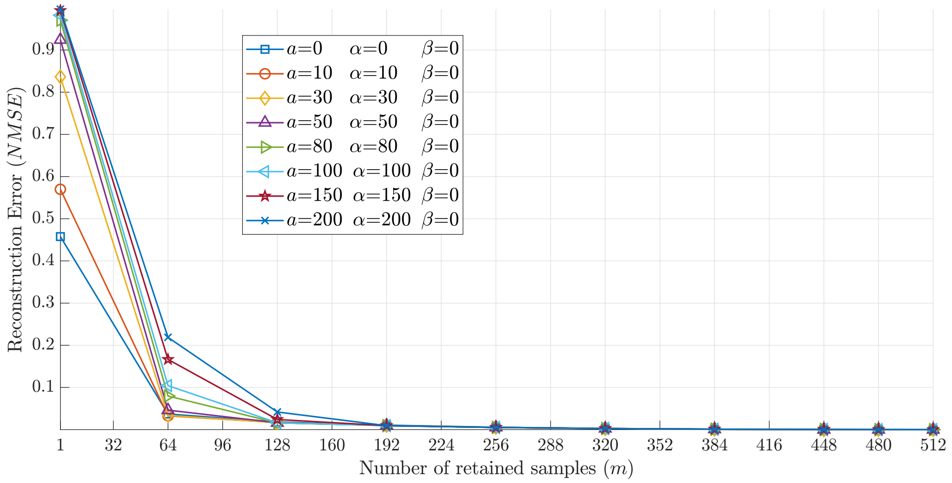


FIGURE 11. Results of the reconstruction error using real image for different values of Racah parameters values (a, $\alpha = a$, and $\beta = 0$).

shows the obtained NMSE results of the second experiment. From FIGURE 13, the results demonstrate that at moment order of 64, the best NMSE is 0.03169 for DRP parameters of $a = 50, \alpha = 25, \beta = 0$. The second best NMSE appears at $a = 50, \alpha = 0, \beta = 0$ with NMSE of 0.03255; while the third best NMSE occurs at $a = 100$, $\alpha = 0$, $\beta = 0$ with NMSE of 0.03263. For the moment order of 128, the best NMSE is 0.01878 for DRP parameters $a = 100$, $\alpha = 50$, $\beta = 0$. The best NMSE for a moment order of 256 occurred at DRP parameters $a = 200, \alpha = 100, \beta = 0$ with NMSE of 0.00786. For the sake of clarity, the visual reconstruction error between the original and the reconstructed image is acquired and the PSNR is reported for different values of DRP parameters as shown in FIGURE 14 to evaluate the performance of the proposed work.

Another range of DRP parameters values has been used to determine the higher performance. The DRP parameter values in the range (a and $\alpha = a$ with $\beta = \{0, a/2, a\}$) are used to carry out the reconstruction error analysis. FIGURE 15 shows the reported NMSE results. From FIGURE 15, the results demonstrate that at moment order of 64, the best NMSE is 0.04446 for DRP parameters of $a = 50$, $\alpha = 50$, $\beta = 0$. The second best NMSE appears at $a = 100, \alpha = 100$, $\beta = 0$ with NMSE of 0.1027; while the third best NMSE occurs at $a = 50$, $\alpha = 50$, $\beta = 25$ with NMSE of 0.1741. For moment order of 128, the best NMSE is 0.0185 for DRP

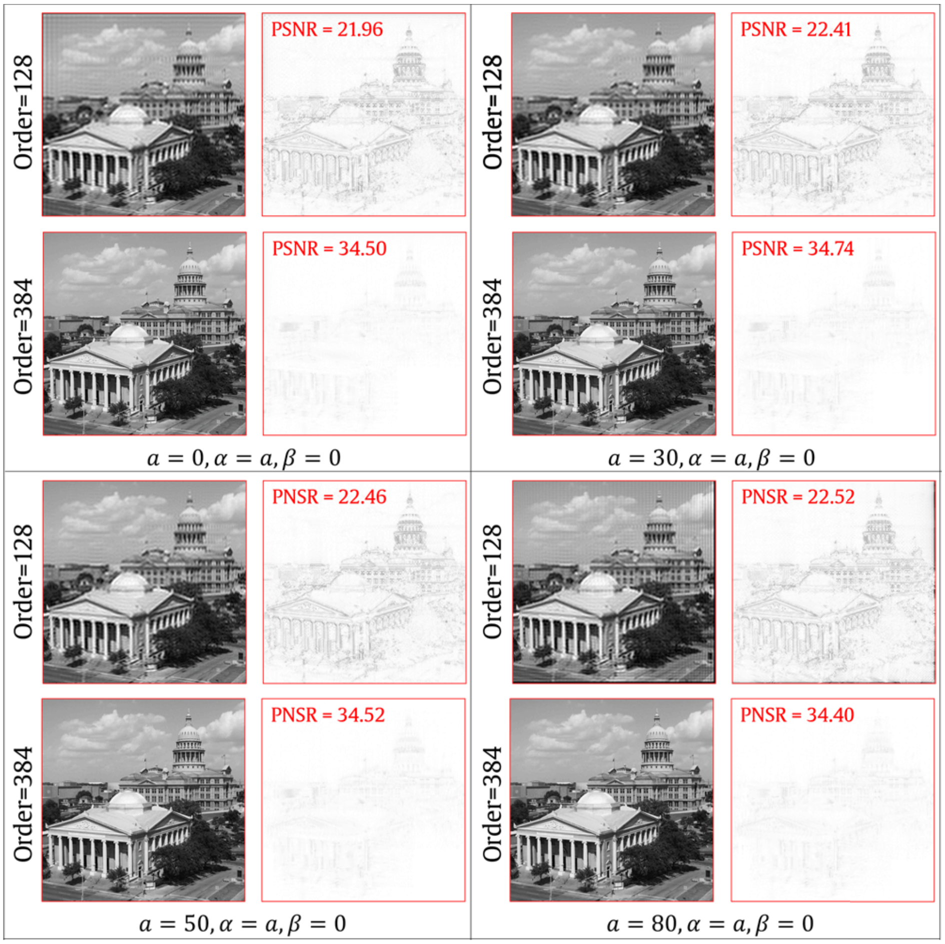


FIGURE 12. **Visual result of the reconstruction error using real image for different values of Racah parameters values (a and $\alpha = (a)$ with $\beta = \{0, a/2\}$).**

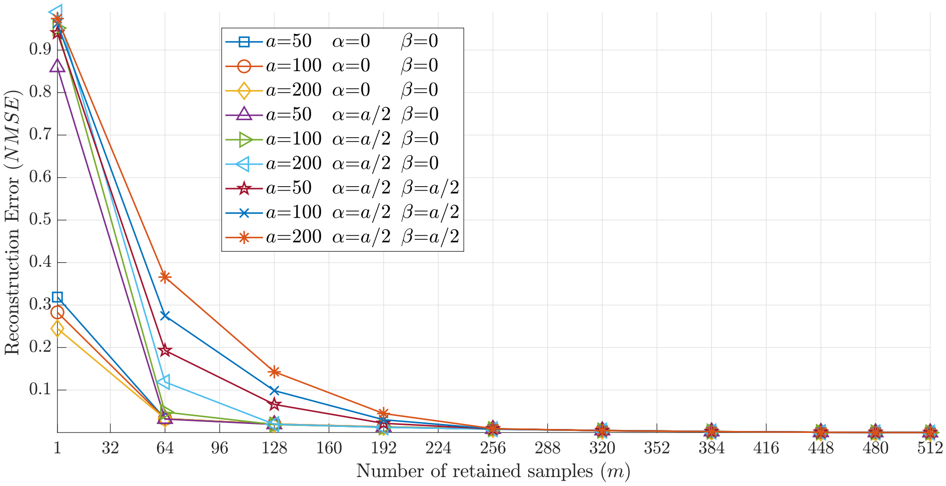


FIGURE 13. **Result of the reconstruction error using a real image for different values of Racah parameters values (a and $\alpha = \{0, a/2\}$ with $\beta = \{0, a/2\}$).**

parameters $a = 100$, $\alpha = 100$, $\beta = 0$ and 0.0186 at $a = 100$, $\alpha = 100$, $\beta = 0$. In addition, the best NMSE for the moment order 256 occurred at DRP parameters $a = 200$, $\alpha = 200$, $\beta = 0$ with NMSE of 0.00751, and at DRP parameters $a = 200$, $\alpha = 200$, $\beta = 100$ with NMSE of 0.00752. The visual reconstruction error between the original and the

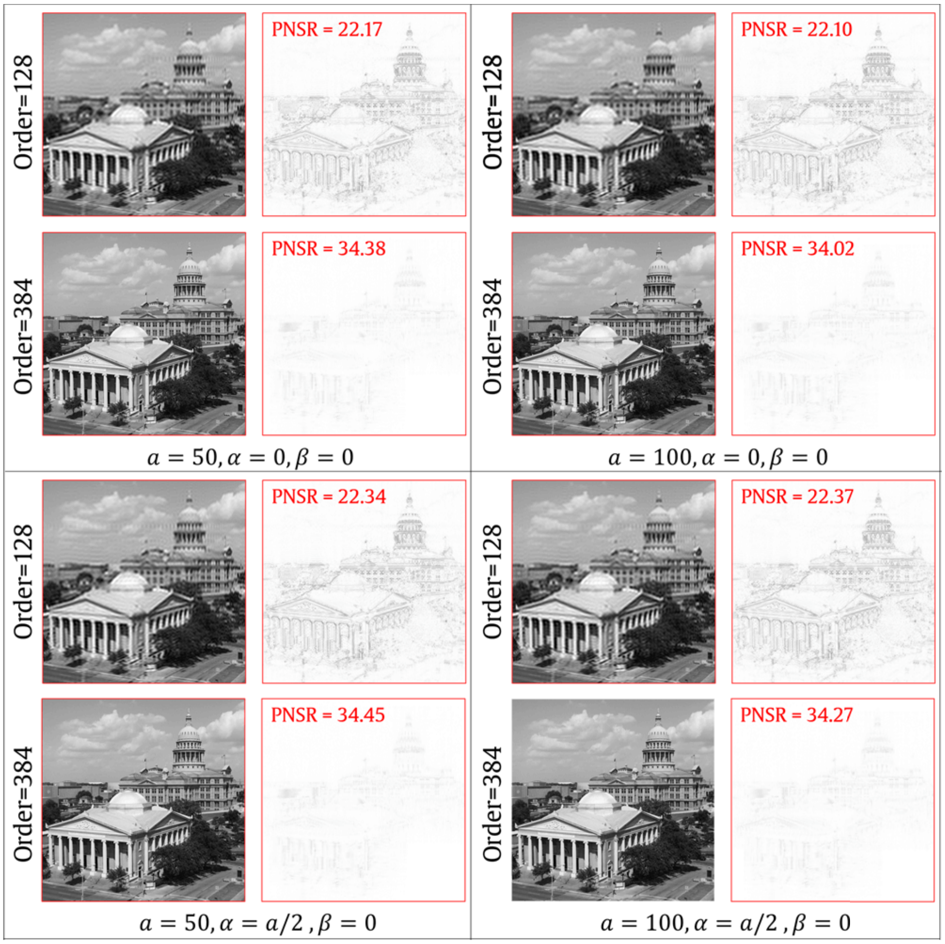


FIGURE 14. Visual result of the reconstruction error using real image for different values of Racah parameters values (a and $\alpha = \{0, a/2\}$ with $\beta = \{0, a/2\}$).

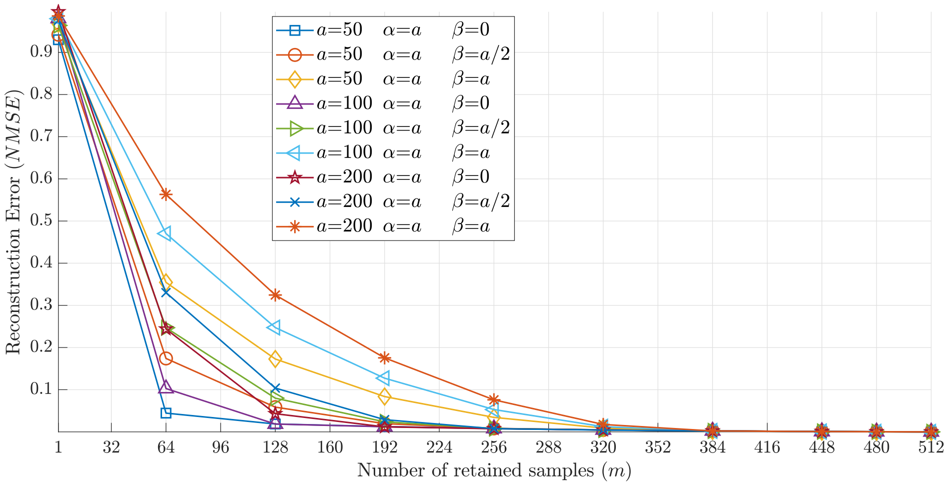


FIGURE 15. Result of the reconstruction error using a real image for different values of Racah parameters values (a and $\alpha = a$ with $\beta = \{0, a/2, a\}$).

reconstructed image is acquired and the PSNR is reported for different values of DRP parameters as shown in FIGURE 16. From the results, the best performance is achieved at different DRP parameters as discussed earlier. In addition, choosing the parameters for best performance depends significantly on the type of used application and its requirements.

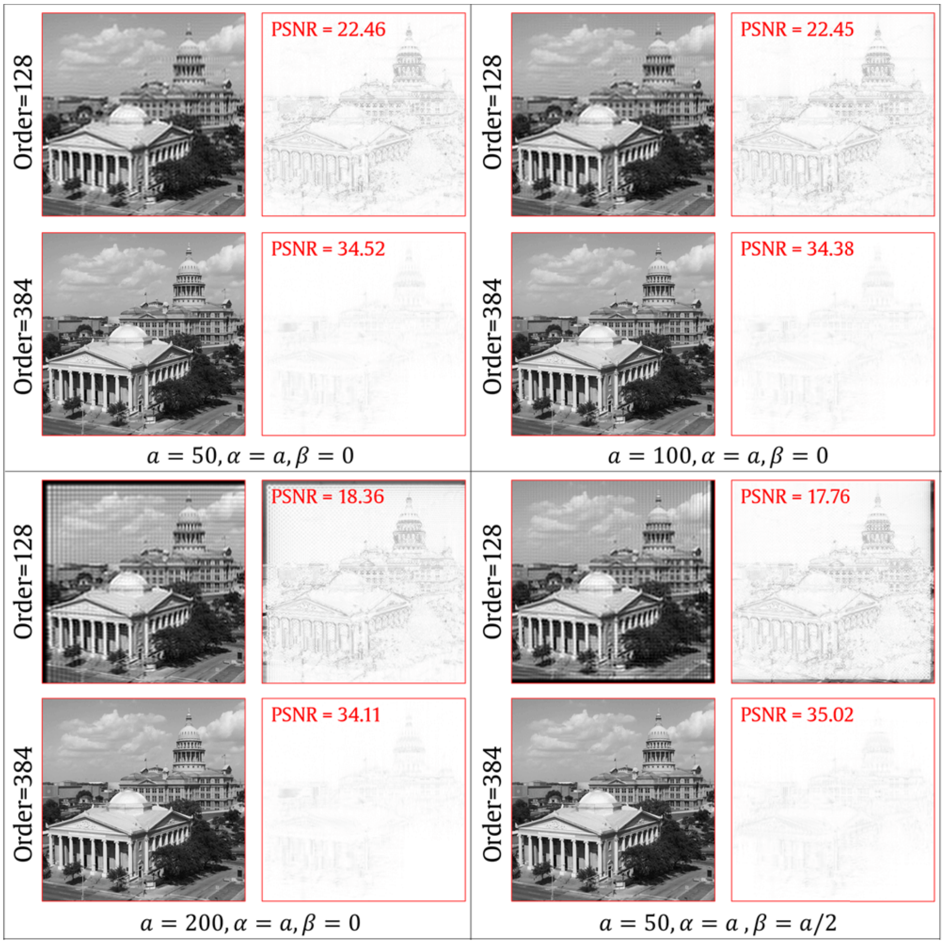


FIGURE 16. Visual result of the reconstruction error using real image for different values of Racah parameters values (a and $\alpha = a$ with $\beta = \{0, a/2\}$).

V. CONCLUSION

This paper presents a novel algorithm for computing the coefficient values of Discrete Racah Polynomials (DRPs). The algorithm utilizes the logarithmic gamma function to compute initial values, enabling efficient computation for a wide range of DRP parameter values and large polynomial sizes. Additionally, a new formula is employed to calculate the values of the initial sets based on the initial value. The remaining DRP coefficients are computed by partitioning the DRP plane into two parts. In the first part, the values are computed using a recurrence relation in the "n" direction. To mitigate propagation errors, a stabilizing condition is enforced in the second part. The performance of the proposed algorithm is evaluated using different DRP parameter values and compared with existing algorithms. The experimental results demonstrate that the proposed algorithm significantly reduces computational costs compared to the existing algorithms. Moreover, the proposed algorithm successfully generates DRPs for large sizes without propagation errors. Furthermore, restriction error and reconstruction error analyses are conducted to assess the impact of the chosen parameter values. These analyses provide insights into the influence of the parameter values on the accuracy and quality of the generated DRPs.

APPENDIX

A. THE STATE-OF-THE-ART ALGORITHMS

We can find significant algorithms of two authors in the literature, the original Zhu's paper and Daoui's approach.

1) ZHU'S ALGORITHMS

Zhu et al. in [25] published two algorithms for Racah polynomial computation: recurrence over the order n and recurrence over the index s. The recurrence formula of weighted Racah polynomials over the order n is

$$\hat{\mathcal{R}}_{n+1}(s) = \left(B\frac{d_n}{d_{n+1}}\hat{\mathcal{R}}_n(s) - C\frac{d_{n-1}}{d_{n+1}}\hat{\mathcal{R}}_{n-1}(s) \right) /A \quad (39)$$

with initial conditions

$$\hat{\mathcal{R}}_0(s) = \sqrt{\frac{\varrho(s)}{d_n^2}(2s+1)},$$

$$
\begin{aligned}
&\hat{\mathcal{R}}_1(s)\\
&= \Bigg(-\frac{\varrho(s+1)(s+1-a)(s+1+b)(s+1+a-\beta)(b+\alpha-s-1)}{\varrho(s)(2s+1)}\\
&\quad+\frac{\varrho(s)(s-a)(s+b)(s+a-\beta)(b+\alpha-s)}{\varrho(s)(2s+1)}\Bigg)\sqrt{\frac{\varrho(s)}{d_n^2}(2s+1)},
\end{aligned} \tag{40}
$$

where

$$
\begin{aligned}
A &= \frac{(n+1)(\alpha+\beta+n)}{(\alpha+\beta+2n+1)(\alpha+\beta+2n+2)},\\
B &= s(s+1) - \frac{a^2+b^2+(a-\beta)^2+(b+\alpha)^2}{4} + \frac{(\alpha+\beta+2n)(\alpha+\beta+2n+2)}{8}\\
&\quad - \frac{(\beta^2-\alpha^2)[(b+\alpha/2)^2-(a-\beta/2)^2]}{2(\alpha+\beta+2n)(\alpha+\beta+2n+2)},\\
C &= \frac{(\alpha+n)(\beta+n)}{(\alpha+\beta+2n)(\alpha+\beta+2n+1)}\\
&\quad\times\left[\left(a+b+\frac{\alpha-\beta}{2}\right)^2-\left(n+\frac{\alpha+\beta}{2}\right)^2\right]\\
&\quad\times\left[\left(b-a+\frac{\alpha+\beta}{2}\right)^2-\left(n+\frac{\alpha+\beta}{2}\right)^2\right].
\end{aligned} \tag{41}
$$

The second algorithm is recurrence over the index s

$$
\begin{aligned}
&\hat{\mathcal{R}}_n(s)\\
&= \frac{(2s-1)[\sigma(s-1)+(s-1)\tau(s-1)-2\lambda s(s-1)]}{(s-1)[\sigma(s-1)+(2s-1)\tau(s-1)]}\\
&\quad\times\sqrt{\frac{\rho(s)(2s+1)}{\rho(s-1)(2s-1)}}\hat{\mathcal{R}}_n(s-1)\\
&\quad-\frac{2\sigma(s-1)}{(s-1)[\sigma(s-1)+(2s-1)\tau(s-1)]}\\
&\quad\times\sqrt{\frac{\rho(s)(2s+1)}{\rho(s-2)(2s-3)}}\hat{\mathcal{R}}_n(s-2),
\end{aligned} \tag{42}
$$

where

$$
\begin{aligned}
\sigma(s) &= (s+a-\beta(b+\alpha-s))(s-a)(s+b)\\
\tau(s) &= a(\alpha+1)(a-\beta)+b(b+\alpha)(\beta+1)\\
&\quad-(\alpha+1)(\beta+1)-s(s+1)(\alpha+\beta+2)\\
\lambda &= n(n+1+\alpha+\beta).
\end{aligned} \tag{43}
$$

The initial values declared in the original paper [25] does not work. We can use either the recurrence over n for $s=a$ and $s=a+1$ or one of the following algorithms can be used.

2) DAOUI'S ALGORITHMS

Daoui et al. in [35] proposed more stable algorithm for DRP computation with two modifications. One problem is overflow of the initial value $\hat{\mathcal{R}}_0(a)$ for high values of the parameter β. When β is integer, we can compute $\hat{\mathcal{R}}_0(a)$ by recurrence

$$
\begin{aligned}
F(0) &= \frac{\alpha+1}{(a+b)(\alpha+b-a)}\\
F(k) &= \frac{(\alpha+k+1)(2a-k+1)}{(a+b-k)(b-a+\alpha+k)}F(k-1),\\
&\quad k=1,2,\ldots,\beta\\
\hat{\mathcal{R}}_0(a) &= \sqrt{F(\beta)(2a+1)}.
\end{aligned} \tag{44}
$$

The other values are obtained by the recurrence relation over n as in Eq. (39). It is called Algorithm 1 in the paper [35].

Another algorithm is based on the recurrence over s. It begins by the same way, computation of $\hat{\mathcal{R}}_0(a)$ by Eq. (44). The initial values of higher degrees follow

$$
\begin{aligned}
&\hat{\mathcal{R}}_n(a)\\
&= \frac{(a-b+n)(\beta+n)(a+b+\alpha+n)}{n}\sqrt{D}\,\hat{\mathcal{R}}_{n-1}(a),\\
&D\\
&= \frac{n(\alpha+\beta+2n+1)(\alpha+\beta+n)}{(\alpha+n)(\beta+n)(b-a+\alpha+\beta+n)(a+b+\alpha+n)}\\
&\quad\times\frac{1}{(\alpha+\beta+2n-1)(a+b-\beta-n)(b-a-n)}.
\end{aligned} \tag{45}
$$

The rest of the initial values is computed as

$$
\hat{\mathcal{R}}_n(a+1) = E\sqrt{\frac{\rho(a+1)}{\rho(a)}\cdot\frac{2a+3}{2a+1}}\,\hat{\mathcal{R}}_n(a), \tag{46}
$$

where (47), as shown at the bottom of the next page, and

$$
\frac{\rho(a+1)}{\rho(a)} = \frac{(2a+1)(\beta+1)(b+\alpha+a+1)(b-a-1)}{(b+\alpha-a-1)(2a-\beta+1)(a+b+1)}. \tag{48}
$$

In the paper [35], these initial conditions are not written precisely.

Finally, Daoui et al. use the stabilizing condition. When $\hat{\mathcal{R}}_n(s)$ is computed by Eq. (42), the new value is tested. When

$$
n > N/6\;\&\;\left|\hat{\mathcal{R}}_n(s)\right| < 10^{-6}\;\&\;\left|\hat{\mathcal{R}}_n(s)\right| > \left|\hat{\mathcal{R}}_n(s-1)\right|, \tag{49}
$$

the value of $\hat{\mathcal{R}}_n(s)$ is substituted by zero. It erases senselessly high values distorted by propagated error. It is called Algorithm 3 in the paper [35]. We will use it, after the error correction, as the reference algorithm.

3) GRAM-SCHMIDT ORTHOGONALIZATION

Gram-Schmidt orthogonalization process (GSOP) is a way, how to change a set of functions to another set of orthogonal functions. It can be used for derivation of completely new orthogonal polynomials, e.g. GSOP applied on a set $\{1, x, x^2, \ldots\}$ in the interval $\langle -1, 1\rangle$ gives Legendre polynomials, see e.g. [41]. We can use GSOP also for increasing precision of orthogonal polynomials computed by another method. Here we have computed $\hat{\mathcal{R}}_n(s)$, but we are not sure, if it is sufficiently precise. We can compute correction

$$
\begin{aligned}
\mathcal{T}(s) &= \sum_{k=0}^{n-1}\hat{\mathcal{R}}_k(s)\left(\sum_{i=a}^{a+N-1}\hat{\mathcal{R}}_n(i)\,\hat{\mathcal{R}}_k(i)\right),\\
&\quad s=a,a+1,\ldots,a+N-1.
\end{aligned} \tag{50}
$$

This correction is then subtracted from the original value

$$\check{\mathcal{R}}_n(s) = \hat{\mathcal{R}}_n(s) - \mathcal{T}(s), \;\; s = a, a+1, \ldots, a+N-1. \tag{51}$$

Then we must correct also the norm

$$\tilde{\mathcal{R}}_n(s) = \check{\mathcal{R}}_n(s) \,/\, \left(\sqrt{\sum_{i=a}^{a+N-1} \check{\mathcal{R}}_n(s)^2} + \varepsilon\right), \quad s{=}a, a{+}1, \ldots, a{+}N{-}1, \tag{52}$$

where ε is some small value preventing division by zero. In Matlab $\varepsilon = 2.2204 \cdot 10^{-16}$. $\tilde{\mathcal{R}}_n(s)$ is now version of $\hat{\mathcal{R}}_n(s)$ with increased precision.

GSOP works well, its main disadvantage is a high computing complexity $\mathcal{O}(N^3)$ (if we compute all the degrees up to $n = N-1$), while the computing complexity of all other algorithms mentioned in this paper is $\mathcal{O}(N^2)$. It is big limitation of this method, the computing time may not be acceptable for very high N.

B. PROOF OF THE SYMMETRY RELATION

The proof of the symmetry relation (28) in the case $a = \alpha = \beta = 0$ is here. The $\hat{\mathcal{R}}_n^{\binom{0,0}{0,b}}(s)$ is then given as follows

$$\begin{aligned}
&\hat{\mathcal{R}}_n^{\binom{0,0}{0,b}}(s) \\
&= \frac{(b{+}1)_n(1)_n(-b{+}1)_n}{n!} \\
&\quad \times {}_4F_3\left(\begin{matrix}-n, -s, s{+}1, n{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right) \\
&\quad \times \sqrt{\frac{\frac{\Gamma(s+1)\Gamma(b+s+1)\Gamma(b-s)\Gamma(s+1)}{\Gamma(b+s+1)\Gamma(b-s)\Gamma(s+1)\Gamma(s+1)}}{\frac{\Gamma(n+1)\Gamma(n+1)\Gamma(b+n+1)\Gamma(b+n+1)}{(2n+1)\Gamma(n+1)\Gamma(b-n)\Gamma(n+1)\Gamma(b-n)}}(2s{+}1)} \\
&= \frac{(b{+}1)_n(1)_n(-b{+}1)_n}{n!} \times {}_4F_3\left(\begin{matrix}-n, -s, s{+}1, n{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right) \\
&\quad \times \sqrt{\frac{(2n+1)\Gamma(b-n)\Gamma(b-n)}{\Gamma(b+n+1)\Gamma(b+n+1)}(2s{+}1)} \\
&= \frac{(b{+}1)_n(1)_n(-b{+}1)_n}{n!} \times {}_4F_3\left(\begin{matrix}-n, -s, s{+}1, n{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right) \\
&\quad \times \frac{\Gamma(b-n)}{\Gamma(b+n+1)}\sqrt{(2n+1)(2s{+}1)} \\
&= \frac{\Gamma(b{+}n{+}1)n!\Gamma(-b{+}1{+}n)}{n!\Gamma(b{+}1)\Gamma(-b{+}1)} \\
&\quad \times {}_4F_3\left(\begin{matrix}-n, -s, s{+}1, n{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right)
\end{aligned}$$

$$\begin{aligned}
&\quad \times \frac{\Gamma(b-n)}{\Gamma(b+n+1)}\sqrt{(2n+1)(2s{+}1)} \\
&= \frac{\Gamma(b-n)\Gamma(-b+1+n)}{\Gamma(b+1)\Gamma(-b+1)} \\
&\quad \times {}_4F_3\left(\begin{matrix}-n, -s, s{+}1, n{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right) \times \sqrt{(2n+1)(2s{+}1)}.
\end{aligned} \tag{53}$$

Roman [42] shows the property of factorial

$$c!(-c-1)! = (-1)^{c+(c<0)}, \tag{54}$$

where

$$(c<0) = \begin{cases} 1 & \text{if } c < 0 \\ 0 & \text{if } c \geq 0. \end{cases} \tag{55}$$

It is well known that $c! = \Gamma(c{+}1)$; thus (54) can be written by this way

$$\Gamma(c+1)\Gamma(-c) = (-1)^{c+(c<0)}. \tag{56}$$

Using (56), the term $\Gamma(b-n)\Gamma(-b+1+n)$ from (53) can be expressed as follows

$$\begin{aligned}
\Gamma(b-n)\Gamma(-b+1+n) &= \Gamma(b-n)\Gamma(-(b-n)+1) \\
&= (-1)^{-(b-n)+1} = -(-1)^{-b}(-1)^n.
\end{aligned} \tag{57}$$

Also, the term $\Gamma(b+1)\Gamma(-b+1)$ from (53) can be expressed as follows

$$\begin{aligned}
\Gamma(b+1)\Gamma(-b+1) &= \Gamma(b+1)\Gamma(-b)(-b) \\
&= (-b)(-1)^{b+0} = -b(-1)^{-b}.
\end{aligned} \tag{58}$$

From (57), (58), and (53), $\hat{\mathcal{R}}_n^{\binom{0,0}{0,b}}(s)$ can be expressed as follows

$$\begin{aligned}
\hat{\mathcal{R}}_n^{\binom{0,0}{0,b}}(s) &= \frac{-(-1)^{-b}(-1)^n}{-b(-1)^{-b}}\sqrt{(2n{+}1)(2s{+}1)} \\
&\quad \times {}_4F_3\left(\begin{matrix}-n, -s, s{+}1, n{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right) \\
&= \frac{(-1)^n\sqrt{(2n+1)(2s{+}1)}}{b} \\
&\quad \times 4F_3\left(\begin{matrix}-n, -s, s{+}1, n{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right).
\end{aligned} \tag{59}$$

For (59), replacing n by s, we obtain:

$$\begin{aligned}
\hat{\mathcal{R}}_s^{\binom{0,0}{0,b}}(n) &= \frac{(-1)^s\sqrt{(2s{+}1)(2n{+}1)}}{b} \\
&\quad \times 4F_3\left(\begin{matrix}-s, -n, n{+}1, s{+}1\\ 1, b{+}1, -b{+}1\end{matrix}\,\middle|\, 1\right).
\end{aligned} \tag{60}$$

$$\begin{aligned}
E &= \left(1 - \frac{2\lambda(a+1)}{\tau(a)}\right) \\
&= \left(1 + \frac{2n(\alpha+\beta+n+1)(a+1)}{(\alpha+1)(\beta+1) + a(a+1)(\alpha+\beta+2) - a(\alpha+1)(a-\beta) - b(\beta+1)(b+\alpha)}\right),
\end{aligned} \tag{47}$$

By comparing (59) with (60), we obtain the symmetry relation

$$\hat{\mathcal{R}}_s^{\binom{0,0}{0,b}}(n) = (-1)^{(s-n)}\hat{\mathcal{R}}_n^{\binom{0,0}{0,b}}(s)\,. \square \tag{61}$$

REFERENCES

[1] D. A. Abdulqader, M. S. Hathal, B. M. Mahmmod, S. H. Abdulhussain, and D. Al-Jumeily, "Plain, edge, and texture detection based on orthogonal moment," *IEEE Access*, vol. 10, pp. 114455–114468, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9930344

[2] S. H. Abdulhussain, B. M. Mahmmod, J. Flusser, K. A. Al-Utaibi, and S. M. Sait, "Fast overlapping block processing algorithm for feature extraction," *Symmetry*, vol. 14, no. 4, p. 715, Apr. 2022. [Online]. Available: https://www.mdpi.com/2073-8994/14/4/715

[3] A. Daoui, H. Karmouni, M. Yamni, M. Sayyouri, and H. Qjidaa, "On computational aspects of high-order dual Hahn moments," *Pattern Recognit.*, vol. 127, Jul. 2022, Art. no. 108596.

[4] A. Daoui, H. Karmouni, M. Sayyouri, and H. Qjidaa, "Fast and stable computation of higher-order Hahn polynomials and Hahn moment invariants for signal and image analysis," *Multimedia Tools Appl.*, vol. 80, pp. 32947–32973, Sep. 2021.

[5] A. M. Salih and S. H. Mahmood, "Digital color image watermarking using encoded frequent mark," *J. Eng.*, vol. 25, no. 3, pp. 81–88, Feb. 2019.

[6] S. H. Abdulhussain, A. R. Ramli, A. J. Hussain, B. M. Mahmmod, and W. A. Jassim, "Orthogonal polynomial embedded image kernel," in *Proc. Int. Conf. Inf. Commun. Technol.*, New York, NY, USA, Apr. 2019, pp. 215–221, doi: 10.1145/3321289.3321310.

[7] Z. I. Abood, "Composite techniques based color image compression," *J. Eng.*, vol. 23, no. 3, pp. 80–93, Feb. 2017.

[8] M. Usman, W. Alhejaili, M. Hamid, and N. Khan, "Fractional analysis of Jeffrey fluid over a vertical plate with time-dependent conductivity and diffusivity: A low-cost spectral approach," *J. Comput. Sci.*, vol. 63, Sep. 2022, Art. no. 101769.

[9] D. Chakraborty and J.-H. Jung, "Efficient determination of the critical parameters and the statistical quantities for Klein–Gordon and sine-Gordon equations with a singular potential using generalized polynomial chaos methods," *J. Comput. Sci.*, vol. 4, nos. 1–2, pp. 46–61, Jan. 2013.

[10] J. Feinberg and H. P. Langtangen, "Chaospy: An open source tool for designing methods of uncertainty quantification," *J. Comput. Sci.*, vol. 11, pp. 46–57, Nov. 2015.

[11] H. Abbas and L. E. George, "Image classification schemes based on sliced radial energy distribution of DFT and the statistical moments of Haar wavelet," *Iraqi J. Sci.*, vol. 61, no. 3, pp. 687–712, Mar. 2020.

[12] B. S. Journal, "Orthogonal functions solving linear functional differential Equations Using Chebyshev polynomial," *Baghdad Sci. J.*, vol. 5, no. 1, pp. 143–148, Mar. 2008.

[13] C.-L. Lim, B. Honarvar, K.-H. Thung, and R. Paramesran, "Fast computation of exact Zernike moments using cascaded digital filters," *Inf. Sci.*, vol. 181, no. 17, pp. 3638–3651, Sep. 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025511002039

[14] M. R. Eslahchi, M. Dehghan, and S. Amani, "The third and fourth kinds of Chebyshev polynomials and best uniform approximation," *Math. Comput. Model.*, vol. 55, nos. 5–6, pp. 1746–1762, Mar. 2012.

[15] X. Xu, L. Xiong, and F. Zhou, "Solving fractional optimal control problems with inequality constraints by a new kind of Chebyshev wavelets method," *J. Comput. Sci.*, vol. 54, Sep. 2021, Art. no. 101412.

[16] R. Mukundan, "Some computational aspects of discrete orthonormal moments," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1055–1059, Aug. 2004.

[17] B. Honarvar Shakibaei Asli, R. Paramesran, and C.-L. Lim, "The fast recursive computation of tchebichef moment and its inverse transform based on Z-transform," *Digit. Signal Process.*, vol. 23, no. 5, pp. 1738–1746, Sep. 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1051200413001188

[18] S. H. Abdulhussain, A. R. Ramli, S. A. R. Al-Haddad, B. M. Mahmmod, and W. A. Jassim, "Fast recursive computation of Krawtchouk polynomials," *J. Math. Imag. Vis.*, vol. 60, no. 3, pp. 285–303, Mar. 2018.

[19] B. Honarvar Shakibaei Asli and J. Flusser, "Fast computation of Krawtchouk moments," *Inf. Sci.*, vol. 288, pp. 73–86, Dec. 2014.

[20] B. H. S. Asli and M. H. Rezaei, "Four-term recurrence for fast Krawtchouk moments using Clenshaw algorithm," *Electronics*, vol. 12, no. 8, p. 1834, Apr. 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/8/1834

[21] S. H. Abdulhussain and B. M. Mahmmod, "Fast and efficient recursive algorithm of Meixner polynomials," *J. Real-Time Image Process.*, vol. 18, no. 6, pp. 2225–2237, Dec. 2021.

[22] B. M. Mahmmod, S. H. Abdulhussain, T. Suk, and A. Hussain, "Fast computation of Hahn polynomials for high order moments," *IEEE Access*, vol. 10, pp. 48719–48732, 2022.

[23] J. A. Wilson, "Hypergeometric series recurrence relations and some new orthogonal functions," Ph.D. dissertation, Univ. Wisconsin, Madison, WI, USA, 1978.

[24] R. Koekoek and R. F. Swarttouw, "The Askey-scheme of hypergeometric orthogonal polynomials and its q-analogue," 1996, *arXiv:math/9602214*.

[25] H. Zhu, H. Shu, J. Liang, L. Luo, and J.-L. Coatrieux, "Image analysis by discrete orthogonal Racah moments," *Signal Process.*, vol. 87, no. 4, pp. 687–708, Apr. 2007.

[26] K. Fardousse, A. El Affar, H. Qjidaa, and A. Cherkaoui, "Robust skeletonization of hand written craft motives of 'Zellij' using Racah moments," *Int. J. Comput. Sci. Netw. Secur.*, vol. 9, no. 8, pp. 216–221, 2009

[27] Y. Wu and S. Liao, "Chinese characters recognition via Racah moments," in *Proc. Int. Conf. Audio, Lang. Image Process.*, New York, NY, USA, 2014, pp. 691–694.

[28] R. Salouan, S. Safi, and B. Bouikhalene, "Handwritten Arabic characters recognition using methods based on Racah, Gegenbauer, Hahn, tchebychev and orthogonal Fourier–Mellin moments," *Int. J. Adv. Sci. Technol.*, vol. 78, pp. 13–28, May 2015.

[29] J. P. Ananth and V. S. Bharathi, "Face image retrieval system using discrete orthogonal moments," in *Proc. 4th Int. Conf. Bioinf. Biomed. Technol.*, vol. 29. Singapore: IACSIT, 2012, pp. 218–223.

[30] G. Gasper and M. Rahman, "Some systems of multivariable orthogonal q-Racah polynomials," *Ramanujan J.*, vol. 13, nos. 1–3, pp. 389–405, Jun. 2007, doi: 10.1007/s11139-006-0259-8.

[31] R. Benouini, I. Batioua, K. Zenkouar, A. Zahi, H. E. Fadili, and H. Qjidaa, "Fast and accurate computation of Racah moment invariants for image classification," *Pattern Recognit.*, vol. 91, pp. 100–110, Jul. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320319300780

[32] M. El Mallahi, A. Zouhri, A. Mesbah, A. Berrahou, I. El Affar, and H. Qjidaa, "Radial invariant of 2D and 3D Racah moments," *Multimedia Tools Appl.*, vol. 77, no. 6, pp. 6583–6604, Mar. 2018, doi: 10.1007/s11042-017-4573-5.

[33] Z. Lakhili, A. El Alami, A. Mesbah, A. Berrahou, and H. Qjidaa, "Deformable 3D shape classification using 3D Racah moments and deep neural networks," *Proc. Comput. Sci.*, vol. 148, pp. 12–20, Jan. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S187705091930002X

[34] I. Batioua, R. Benouini, K. Zenkouar, and H. E. Fadili, "Image analysis using new set of separable two-dimensional discrete orthogonal moments based on Racah polynomials," *EURASIP J. Image Video Process.*, vol. 2017, no. 1, p. 20, Dec. 2017, doi: 10.1186/s13640-017-0172-7.

[35] A. Daoui, H. Karmouni, M. Sayyouri, and H. Qjidaa, "Stable analysis of large-size signals and images by Racah's discrete orthogonal moments," *J. Comput. Appl. Math.*, vol. 403, Mar. 2022, Art. no. 113830.

[36] A. M. Abdul-Hadi, S. H. Abdulhussain, and B. M. Mahmmod, "On the computational aspects of Charlier polynomials," *Cogent Eng.*, vol. 7, no. 1, Jan. 2020, Art. no. 1763553.

[37] A. K. Jain, *Fundamentals of Digital Image Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 1989.

[38] H. Sheikh, Z. Wang, L. Cormack, and A. Bovik. *LIVE Image Quality Assessment Database Release 2*. Accessed: Feb. 10, 2022. [Online]. Available: http://live.ece.utexas.edu/research/quality

[39] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.

[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[41] W. H. Thomas II, ‘‘Introduction to real orthogonal polynomials,’’ M.S. thesis, Naval Postgraduate School, Monterey, CA, USA, Jun. 1992.

[42] S. Roman, ‘‘The logarithmic binomial formula,’’ *Amer. Math. Monthly*, vol. 99, no. 7, p. 641, Aug. 1992.

BASHEERA M. MAHMMOD received the Ph.D. degree in computer and embedded system engineering from Universiti Putra Malaysia (UPM), in 2018. She is currently a Lecturer with the Department of Computer Engineering, University of Baghdad. Her research interests include signal processing, speech enhancement, and computer vision.

SADIQ H. ABDULHUSSAIN received the Ph.D. degree in computer and embedded system engineering from Universiti Putra Malaysia (UPM), in 2018. Since 2008, he has been a Faculty Member with the University of Baghdad. His research interests include image processing, computer vision, and signal processing.

TOMÁŠ SUK (Member, IEEE) received the M.Sc. degree in electrical engineering from the Faculty of Electrical Engineering, Czech Technical University, Prague, Czech Republic, in 1987, the Ph.D. degree in computer science from the Czechoslovak Academy of Sciences, in 1992, and the D.Sc. degree from the Czech Academy of Sciences, Prague, in 2018. Since 1991, he has been a Researcher with the Institute of Information Theory and Automation, Czech Academy of Sciences. He has authored more than 35 journal articles and more than 50 conference papers, and coauthored the monographs *Moments and Moment Invariants in Pattern Recognition*. His research interests include invariant features, pattern recognition, image filtering, digital image processing, and moment-based and point-based invariants.

MUNTADHER ALSABAH received the B.S. degree in electrical engineering from the University of Baghdad, Iraq, in 2007, the M.S. degree in wireless communication engineering from Universiti Putra Malaysia (UPM), Malaysia, in 2014, and the Ph.D. degree in wireless communication from The University of Sheffield, Sheffield, U.K., in 2020. His current research interests include wireless communications, multiple-input–multiple-output (MIMO) systems, mmWave frequencies, image processing, signal processing and in applying the artificial intelligence, and machine learning techniques in wireless communications systems.

ABIR HUSSAIN (Senior Member, IEEE) is currently a Professor of machine learning with the Department of Computer Science, Liverpool John Moores University. Her research interests include machine learning algorithms and their applications to medical, image and signal processing, and data analysis.