Efficient and Stable Voxel-Based Algorithm for Computing 3D Zernike Moments and Shape Reconstruction

An-Wen Deng and Chih-Ying Gwo

ABSTRACT

3D Zernike moments based on 3D Zernike polynomials have been successfully applied to the field of voxelized 3D shape retrieval and have attracted more attention in biomedical image processing. As the order of 3D Zernike moments increases, both computational efficiency and numerical accuracy decrease. Due to this phenomenon, a more efficient and stable method for computing high-order 3D Zernike moments was proposed in this study. The proposed recursive formula for computing 3D Zernike radial polynomials combines the recursive calculation of spherical harmonics to develop a voxel-based algorithm for the calculation of 3D Zernike moments. The algorithm was applied to the 3D shape Michelangelo's David with a size of $150 \times 150 \times 150$ voxels. As compared to the method without additional acceleration, the proposed method uses a group action of order sixteen orthogonal group and saving unnecessary iterations, the factor of speed-up is 56.783 ± 3.999 when the order of Zernike moments is between 10 and 450. The proposed method also obtained an accurate reconstructed shape with the error rate (normalized mean square error) of 0.00 (4.17×10^{-3}) when the reconstruction was computed for all moments up to order 450.

Keywords: 3D Zernike moments; 3D Zernike radial polynomials; 3D Zernike polynomials; Spherical harmonics; Recurrence formula; Matrix Lie Group; Group action

The authors are with the Department of Information Management, Chien Hsin University of Science and Technology, Taoyuan City 32097, Taiwan.

E-mail: awdeng@uch.edu.tw, ericgwo@uch.edu.tw.

1. Introduction

The 3D Zernike moments as shape descriptor based on 3D Zernike polynomials have been widely applied in the many fields of 3D image processing [1, 2], such as terrain matching [3, 4], shape retrievals [1, 2, 5], human motion analysis [6, 7], and protein-protein interface prediction [8], etc. Due to the orthogonal property of 3D Zernike polynomials, the resultant Zernike moments do not contain redundancy or overlap of information between the moments, and the magnitudes of these moments are invariant to the rotation of the represented 3D shape.

The 3D Zernike moments can be traced back to the work of Canterakis [9, 10]. The 3D moments are expressed by a complex number linear combination of geometric moments, which are widely used in image processing and pattern recognition. In [11-13], the algorithms for calculating geometric moments was proposed by using exact integration in each voxel. Based on the theory in [1, 2], Novotni and Klein developed the algorithm for calculating Zernike-Canterakis moments. The algorithm has been commonly adopted to calculate 3D Zernike moments. A reformulation of Zernike-Canterakis moments and its implementation were provided in the article by Callahan and De Graef [11]. The coefficients of geometric moments range from 10^0 to 10^9 for the maximal order 20 and the geometric moments also have a great dynamic range [12]. This method of computing 3D Zernike moments is subject to severe instability of the geometric moment in the case of higher orders [13]. On the other hand, [14, 15] provided the algorithms for calculating geometric moments based on the exact volumetric integrations for surface representations. The reconstructed images with high resolution from the moments up to order 40 were shown in [14]. These surface-based methods calculate 3D Zernike moments more effectively and accurately than Novotni and Klein's method.

Geometric moments using exact integration in each pixel are also used for computing 2D Zernike moments. However, when the moment order is greater than 45, this method of calculating Zernike moments cannot provide numerical stability [16]. Another approach is to recursively calculate 2D Zernike radial polynomials. Even if the zeroth-order approximation is used for the integration, these methods are stable and efficient [16, 17]. Based on the recursive approach of 3D Zernike radial polynomials, we proposed an algorithm for calculating 3D Zernike moments.

3D Zernike polynomials are the product of 3D Zernike radial polynomials and the spherical harmonics. Spherical harmonics are generally suitable for surface modeling [18], which are the base functions of the L^2 functions on the unit sphere S^2 [19]. If the surface of the 3D shape is compact and can be one-to-one projected onto a sphere

(e.g., a star shaped surface), it is sufficient to describe it using the Fourier coefficients of spherical harmonics. However, the surface of such a 3D object is topologically identical to the unit sphere S^2 [20]. In the real world, not all surfaces of 3D objects are homeomorphic to S^2 , for example, the torus. As described in [1], multiple concentric shells were used to define spherical functions, and spherical harmonics are not suitable for the torus.

In this study, we introduced a series of stable algorithms to compute high-order 3D Zernike moments, which is an extension of our work in [21]. A recursive formula analogue to Kintner's p -method [22] was proposed for calculating 3D Zernike radial polynomials. Based on the recursion, in combination with the calculation of the spherical harmonics, a voxel-based algorithms for computing 3D Zernike moments were developed. The use of the symmetries generated by group actions effectively reduces the calculations in the proposed algorithm. The group theory was comprehensively introduced in [23], where groups of order 4 or 8 were used to calculate 2D Zernike moments [16, 17], and an abelian group was suggested in the computation of 3D Zernike moments [24]. Orthogonal groups were also used to further reduce the amount of calculation. The concept of group actions was used to introduce the definition of shape symmetry coefficient for a 3D shape, which resulted in better acceleration effects for larger coefficients. The proposed methods not only provide an accurate reconstructed voxelized 3D shape but is also very effective and stable when calculating using group action under a matrix Lie group of order sixteen.

The 3D Zernike polynomials form an orthogonal basis for the Hilbert space $L^2(B)$ of square-integrable complex functions over the unit ball B . The projection coefficients of the 3D image function onto the 3D Zernike polynomials are called 3D Zernike moments. The low-order Zernike moments describe the global contour of a shape and the higher-order moments describe the local detail. Low-order shape descriptor provides sufficient 3D shape information for the search and classification of objects; however, the more detailed shape features are required to differentiate the similar shape objects. The proposed method was applied to voxelized 3D shapes having a size of $150 \times 150 \times 150$ voxels with the maximal moment order n_{max} up to 450. The simulation results have shown that the proposed methods are fast and stable as compared to the methods provided in the literature.

The rest of this article is organized as follows. Section 2, a brief introduction to 2D Zernike and 3D Zernike moments were given. Some theories for calculating 3D Zernike moments have also been proposed with proofs, including a summary of spherical harmonic calculations. In Section 3, two kinds of error measurements for the reconstructed voxelized 3D shape were presented. The proposed algorithms for

computing 3D Zernike moments were listed in Section 4. Based on the group action of some specific orthogonal groups, the group theory was applied to accelerate the calculation of 3D Zernike moments in Section 5. A finite group of order sixteen was used to speed up the algorithm by a factor of sixteen. To understand the relationship more clearly between acceleration effects and voxelized 3D shapes, the definition of acceleration factor and the shape symmetry coefficient of 3D shape were given in Section 6. In Section 7, we summarized the experimental simulations with numerical results. In section 8, a brief conclusion was given.

2. 2D Zernike and 3D Zernike Moments

3D Shape analysis is getting more and more attention in the medical community because it can potentially evaluate morphological changes between healthy and pathological structures. Before introducing the 3D Zernike transformation, let us briefly review the case of 2D. The 2D version of Zernike polynomials were introduced by the Nobel prize winner F. Zernike, which form an orthogonal basis for the Hilbert space $L^2(D)$ of square integrable complex functions over the unit disc D [25]. By the definition of Teague [26], the coefficients of the image function projected onto the Zernike polynomials are called Zernike moments. The unit disc D is the topological product of the closed unit interval $I = [0, 1]$ and the unit circle S^1 . For a non-negative integer n (called order) and an integer m (called repetition) for which $|m| \leq n$ and $n \equiv m \pmod{2}$, the 2D Zernike polynomials $V_{nm}(z)$ is the product of 2D Zernike radial polynomial $R_{nm}^{2D}(r)$ defined over I and the complex exponential function $e^{im\theta} = \cos m\theta + i \sin m\theta$ defined over the unit circle S^1 , where $z = re^{i\theta} \in D$ stands for a complex number, r is its distance to the origin $(0, 0)$ and θ is the angle from the x -axis. The computations of 2D Zernike polynomials, Zernike moments and their variants were comprehensively studied in [17, 27-30].

For the 3D Zernike polynomial, the unit disc D of 2D Zernike polynomial is replaced by the unit ball B which can be regarded as the topological product of the closed unit interval I and the unit sphere S^2 . Every point (x, y, z) in the unit ball B can be represented by a spherical coordinates (r, θ, ϕ) as shown in Eq. (1), where $r \in [0, 1]$ is the distance from the point to the origin $(0, 0, 0)$, $\theta \in [0, \pi]$ is the colatitude angle, i.e. the angle from the z -axis, and $\phi \in [0, 2\pi)$ is the angle from the x -axis on plane $z = r \sin \theta$.

$$(x, y, z) = (r \sin \theta \cos \phi, r \sin \theta \sin \phi, r \cos \theta) \quad (1)$$

The complex exponentials functions $e^{im\theta}$ ($m = \dots, -3, -2, -1, 0, 1, 2, 3, \dots$) build an orthonormal basis for the Hilbert space $L^2(S^1)$. In the case of S^2 , the spherical

harmonics $Y_\ell^m(\cdot)$ form an orthonormal basis for the Hilbert space $L^2(S^2)$ of the square-integrable functions over the unit sphere S^2 [31, 32]. Any function f of $L^2(S^2)$ can be expressed as follows:

$$f(\theta, \phi) = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} C_\ell^m Y_\ell^m(\theta, \phi) \quad (2)$$

where C_ℓ^m are the coefficients. For a non-negative integer ℓ and an arbitrary integer m with $|m| \leq \ell$, the spherical harmonic $Y_\ell^m(\cdot)$ of degree ℓ with order m is given by

$$Y_\ell^m(\theta, \phi) = Y_\ell^m(x, y, z) = K_\ell^m P_\ell^m(\cos \theta) e^{im\phi} \quad (3)$$

where $P_\ell^m(\cdot)$ is the associated Legendre polynomial of degree ℓ , K_ℓ^m is the normalizing factor given by

$$K_\ell^m = \sqrt{\frac{(2\ell+1)(\ell-m)!}{(\ell+m)!}} \quad (4)$$

The orthogonal properties hold in Eq. (5) and (6), thus the spherical harmonics $Y_\ell^m(\cdot)$ are mutually orthogonal.

$$\int_0^{2\pi} e^{im\phi} e^{im'\phi} d\phi = 2\pi \delta_{mm'} \quad (5)$$

$$\int_0^\pi P_\ell^m(\cos \theta) P_{\ell'}^m(\cos \theta) \sin \theta d\theta = \frac{2(\ell+m)!}{(2\ell+1)(\ell-m)!} \delta_{\ell\ell'} \quad (6)$$

where $\delta_{mm'}$ is the Kronecker delta.

Let $\widetilde{P}_\ell^m(\cos \theta) = K_\ell^m P_\ell^m(\cos \theta)$ be the normalized associated Legendre polynomial. The calculation of the normalized associated Legendre polynomials $\widetilde{P}_\ell^m(\cos \theta)$ of degree $\ell \leq \ell_{max}$ is presented as follows [33]:

Theorem A. The normalized associated Legendre polynomials can be recursively computed as follows:

1. $\widetilde{P}_0^0(\cos \theta) = 1$
2. $\widetilde{P}_\ell^\ell(\cos \theta) = -C_3 \sin \theta \widetilde{P}_{\ell-1}^{\ell-1}(\cos \theta)$ for $\ell = 1, 2, 3, \dots, \ell_{max}$
3. $\widetilde{P}_\ell^m(\cos \theta) = C_1 \cos \theta \widetilde{P}_{\ell-1}^m(\cos \theta) - C_2 \widetilde{P}_{\ell-2}^m(\cos \theta)$ for $m = 0, 1, \dots, \ell-1$
(set $\widetilde{P}_{\ell-2}^{\ell-1}(\cos \theta) = 0$).

where

$$\begin{aligned} C &= \sqrt{\frac{2\ell+1}{(\ell+m)(\ell-m)}}, \quad C_1 = C\sqrt{2\ell-1} \\ C_2 &= C\sqrt{\frac{(\ell+m-1)(\ell-m-1)}{2\ell-3}}, \quad C_3 = \sqrt{\frac{2\ell+1}{2\ell}} \end{aligned} \quad (7)$$

The skew symmetric property holds for the normalized associated Legendre polynomials

$$\widetilde{P}_\ell^m(-\cos \theta) = (-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta) \quad (8)$$

The spherical harmonics have the following property:

$$\overline{Y_\ell^m(\theta, \phi)} = (-1)^m Y_\ell^{-m}(\theta, \phi) \quad (9)$$

Furthermore, for the pair of antipodal points on S^2 ,

$$Y_\ell^m(-x, -y, -z) = (-1)^\ell Y_\ell^m(x, y, z) \quad (10)$$

For a non-negative integer n (called order), a non-negative integer ℓ for which $\ell \leq n$ and $n \equiv \ell \pmod{2}$ and an integer m with $|m| \leq \ell$, the 3D Zernike polynomial $V_{n\ell}^m(x, y, z)$ is defined as the 3D Zernike radial polynomial $R_{n\ell}(r)$ and the spherical harmonic given by

$$\begin{aligned} V_{n\ell}^m(x, y, z) &= R_{n\ell}(r) Y_\ell^m(\theta, \phi) \\ &= R_{n\ell}(r) \widetilde{P}_\ell^m(\cos \theta) e^{im\phi} \end{aligned} \quad (11)$$

where $(x, y, z) \in B$ with its spherical coordinate system (r, θ, ϕ) . The 3D Zernike radial polynomial is given in terms of Jacobi polynomials as follows [31]:

$$R_{n\ell}(r) = r^\ell P_{\frac{n-\ell}{2}}^{(0, \ell+\frac{1}{2})}(2r^2 - 1) \quad (12)$$

The orthogonal property of Jacobi polynomials $P_n^{(\alpha, \beta)}(x)$ leads to 3D Zernike radial polynomials being mutually orthogonal.

Theorem B. The notations as stated above. It holds

$$\int_0^1 R_{n\ell}(r) R_{n'\ell}(r) r^2 dr = \frac{\delta_{nn'}}{2n+3} \quad (13)$$

where $\delta_{nn'}$ is the Kronecker delta.

Proof. The Jacobi polynomials $P_n^{(\alpha, \beta)}(x)$ are orthogonal with respect to the weight

$(1-x)^\alpha(1+x)^\beta$ over the closed interval $[-1, 1]$ whenever $\alpha > -1$ and $\beta > -1$ as follows:

$$\begin{aligned} \int_0^1 (1-x)^\alpha(1+x)^\beta P_{n'}^{(\alpha,\beta)}(x) P_n^{(\alpha,\beta)}(x) dx \\ = \frac{2^{\alpha+\beta+1}}{(2n+\alpha+\beta+1)} \frac{\Gamma(n+\alpha+1)\Gamma(n+\beta+1)}{\Gamma(n+\alpha+\beta+1)n!} \delta_{nn'} \end{aligned} \quad (14)$$

where $\Gamma(x)$ denotes the Gamma function and $\delta_{nn'}$ denotes the Kronecker delta (Cf.[31]). Substitute

$$x \leftarrow 2r^2 - 1, \alpha \leftarrow 0, \beta \leftarrow \ell + \frac{1}{2}, n \leftarrow \frac{n-\ell}{2} \text{ and } n' \leftarrow \frac{n'-\ell}{2} \quad (15)$$

into Eq. (14), it yields the orthogonal property of 3D Zernike radial polynomials. Q.E.D.

It is worth noting that due to the orthogonal properties stated in Eq. (6) and Eq. (13), a classic argument for the orthogonal polynomials (Cf. Theorem 3.2.1 in [31]) can be used to derive the three-term recurrence formulas of the associated Legendre polynomials and the one for 3D Zernike radial polynomials described in Theorem A and Theorem D, respectively. The definition of the radial polynomial given in (12) looks different from that of Canterakis [9, 10], but these two definitions are equivalent [34].

After expanding the factors in Eq. (12) and performing some algebraic calculations, a representation of the 3D Zernike polynomial is obtained as shown in Eq. (16).

Theorem C. The 3D Zernike polynomial $R_{n\ell}(r)$ has the following polynomial form:

$$R_{n\ell}(r) = \sum_{s=0}^{\frac{n-\ell}{2}} (-1)^s \left(\frac{n-\ell}{2} \right)_s \left(\frac{n-s+\frac{1}{2}}{n-\ell} \right) r^{n-2s} \quad (16)$$

where $\left(\frac{x}{s} \right)$ denotes the generalized binomial coefficient [35].

Due to the numerical accuracy consideration, it is recommended to use a recursive approach instead of computing the Zernike polynomial stated in Eq. (16) directly. Based on the recurrence of the Jacobi polynomials [31], the recursive approach is provided. This 3D recursive formula can be regarded as similar to the Kintner's p -method used in the case of 2D Zernike polynomials [22, 30]. For the completeness, we provide a proof.

Theorem D. For non-negative integers n, ℓ with $n \equiv \ell \pmod{2}$, the following recurrence of generalized Zernike radial polynomials holds:

$$R_{n\ell}(r) = (K_1 r^2 + K_2) R_{n-2,\ell}(r) + K_3 R_{n-4,\ell}(r) \quad (17)$$

for $n = \ell + 4, \ell + 6, \dots, n_{\max}$ where the coefficients K_i are given as follows:

$$\begin{aligned} k_0 &= (n - \ell)(n + \ell + 1)(2n - 3) \\ k_1 &= (2n - 1)(2n + 1)(2n - 3) \\ k_2 &= (-2n + 1) \left(\frac{4\ell^2 + 4\ell + 1}{2} \right) - \frac{k_1}{2} \\ k_3 &= -(n - \ell - 2)(n + \ell + 1)(2n + 1) \end{aligned} \quad (18)$$

$$K_1 = \frac{k_1}{k_0}, K_2 = \frac{k_2}{k_0}, K_3 = \frac{k_3}{k_0}.$$

Proof. This comes directly from the recurrence for Jacobi polynomials (Cf. [31]):

$$\begin{aligned} &2n(n + \alpha + \beta)(2n + \alpha + \beta - 2)P_n^{(\alpha, \beta)}(z) \\ &= (2n + \alpha + \beta - 1) \left(\frac{(2n + \alpha + \beta)(2n + \alpha + \beta - 2)z}{+\alpha^2 - \beta^2} \right) P_{n-1}^{(\alpha, \beta)}(z) \\ &\quad - 2(n + \alpha - 1)(n + \beta - 1)(2n + \alpha + \beta)P_{n-2}^{(\alpha, \beta)}(z) \end{aligned} \quad (19)$$

by assigning $z \leftarrow 2r^2 - 1, \beta \leftarrow \ell + \frac{1}{2}, n \leftarrow \frac{n-\ell}{2}$ and $\alpha \leftarrow 0$. We obtain

$$\begin{aligned} &2 \left(\frac{n - \ell}{2} \right) \left(\frac{n - \ell}{2} + \ell + \frac{1}{2} \right) \left(2 \left(\frac{n - \ell}{2} \right) + \ell + \frac{1}{2} - 2 \right) \frac{R_{n\ell}(r)}{r^\ell} \\ &= \left(2 \left(\frac{n - \ell}{2} \right) + \ell + \frac{1}{2} - 1 \right) \\ &\quad \times \left(\left(2 \left(\frac{n - \ell}{2} \right) + \ell + \frac{1}{2} \right) \left(2 \left(\frac{n - \ell}{2} \right) + \ell + \frac{1}{2} - 2 \right) (2r^2 \right. \\ &\quad \left. - 1) - \left(\ell + \frac{1}{2} \right)^2 \right) \frac{R_{n-2,\ell}(r)}{r^\ell} \end{aligned} \quad (20)$$

$$-2 \left(\left(\frac{n-\ell}{2} \right) - 1 \right) \left(\left(\frac{n-\ell}{2} \right) + \ell + \frac{1}{2} - 1 \right) \left(2 \left(\frac{n-\ell}{2} \right) + \ell + \frac{1}{2} \right) \frac{R_{n-4,\ell}(r)}{r^\ell}$$

After multiplying both sides of the equality by $4r^\ell$ and performing some calculations, we have the recurrence

$$\begin{aligned} (n-\ell)(n+\ell+1)(2n-3)R_{n\ell}(r) &= \\ &= \left((2n-1)(2n+1)(2n-3)r^2 \right. \\ &\quad \left. - (2n-1) \left(\frac{(2n+1)(2n-3) + 4\ell^2 + 4\ell + 1}{2} \right) \right) R_{n-2,\ell} \\ &\quad - (n-\ell-2)(n+\ell-1)(2n+1)R_{n-4,\ell}(r) \end{aligned} \quad (21)$$

Q.E.D.

To apply this recursive formula, the following initial relations are also required:

$$R_{nn}(r) = r^n \quad \text{for } n = 0, 1, 2, \dots \quad (22)$$

and

$$R_{n,n-2}(r) = \left(n + \frac{1}{2} \right) r^n - \left(n - \frac{1}{2} \right) r^{n-2} \quad \text{for } n = 2, 3, 4, \dots \quad (23)$$

Let $f(x, y, z)$ be a 3D image function within the unit ball B whose values lie over the closed interval $[0, 1]$. The 3D Zernike moment $Z_{n\ell}^m$ can be regarded as the inner product of the image function $f(x, y, z)$ with the basis function $V_{n\ell}^m(x, y, z)$, which is given by Eq. (24).

$$\begin{aligned} Z_{n\ell}^m &= \frac{2n+3}{4\pi} \iiint_{(x,y,z) \in B} f(x, y, z) \overline{V_{n\ell}^m(x, y, z)} dx dy dz \\ &= \frac{2n+3}{4\pi} \int_0^{2\pi} \int_0^\pi \int_0^1 f(r, \theta, \phi) R_{n\ell}(r) r^2 dr \widetilde{P}_\ell^m(\cos \theta) \sin \theta d\theta e^{-im\phi} d\phi \end{aligned} \quad (24)$$

where $\bar{V}$ denotes the complex conjugate of V . Since the skew symmetry property of 3D Zernike polynomials $V_{n\ell}^m(x, y, z)$ is inherited from spherical harmonics as Eq. (9), so is the 3D Zernike moments

$$Z_{n\ell}^m = (-1)^m \overline{Z_{n\ell}^{-m}} \quad (25)$$

All moments with the same order n and same degree ℓ build $(2\ell+1)$ -dimensional vector $\vec{Z}_{n\ell}$ given by

$$\vec{Z}_{n\ell} = (Z_{n\ell}^{-\ell}, Z_{n\ell}^{-\ell+1}, \dots, Z_{n\ell}^0, \dots, Z_{n\ell}^{\ell-1}, Z_{n\ell}^{\ell}) \quad (26)$$

The ℓ^2 -norm of $\vec{Z}_{n\ell}$, denoted by

$$\|\vec{Z}_{n\ell}\| = \sqrt{\sum_{m=-\ell}^{\ell} |Z_{n\ell}^m|^2} \quad (27)$$

is rotationally invariant and represents the shape descriptor of a voxelized 3D image.

3. Error Measurements for Reconstruction

Let the function f_M be defined as

$$f_M(x, y, z) = \sum_{n=0}^M \sum_{\ell} \sum_{m=-\ell}^{\ell} Z_{n\ell}^m V_{n\ell}^m(x, y, z) \quad (28)$$

Since the 3D image function f is piecewise constant, and hence squared integrable. The function f_M is convergent to f in the sense of L^2 -norm as M tends to infinity. For large M , f_M is considered to be the reconstructed function $\hat{f}$ of the image function f . The normalized mean square error (NMSE) between the original image and its reconstructed image, shown in Eq. (29), is used to measure the computational stability and accuracy of the proposed method in calculation 3D Zernike moments.

$$\varepsilon^2 = \frac{\iiint_B |f(x, y, z) - \hat{f}(x, y, z)|^2 dx dy dz}{\iiint_B |f(x, y, z)|^2 dx dy dz} \quad (29)$$

This measurement is well suited for the image function f carrying different intensity information. However, the voxelized image is the characteristic function for a 3D object $A \subset B$, the value of $(x, y, z) \in A$ is **1**, otherwise **0**. For a given moment order M , the reconstructed function $\bar{f}$ is defined as follows:

$$\bar{f}(x, y, z) = \begin{cases} 1, & \text{if } f_M(x, y, z) \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (30)$$

So, the error measurement called the relative recovery error rate $\bar{\varepsilon}$ is defined by

$$\bar{\varepsilon} = \frac{\iiint_B |f(x, y, z) - \bar{f}(x, y, z)| dx dy dz}{\iiint_B f(x, y, z) dx dy dz} \quad (31)$$

4. Algorithms for computing 3D Zernike Moments

The algorithms here we proposed are all voxel-based. For simplicity, we assume that N is even, if N is odd, the following discussion needs a slight modification with N replaced by $N + 1$ in (32). Suppose that the entire 3D image is inside the cube $C: [-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}]^3 \subseteq B$ which consists of N^3 voxels. Each voxel $v(\alpha, \beta, \gamma)$ (for $0 \leq \alpha, \beta, \gamma \leq N - 1$) whose center is

$$(x_\alpha, y_\beta, z_\gamma) = \left(\frac{2\alpha + 1 - N}{\sqrt{3}N}, \frac{2\beta + 1 - N}{\sqrt{3}N}, \frac{2\gamma + 1 - N}{\sqrt{3}N} \right) \quad (32)$$

with edge length $\delta = 2/(\sqrt{3}N)$, and $(r_{\alpha\beta\gamma}, \theta_{\alpha\beta\gamma}, \phi_{\alpha\beta\gamma})$ denotes its corresponding spherical coordinates. Of note that for different $\gamma \neq \gamma'$, on the plane $z = z_\gamma$ (or $z = z_{\gamma'}$), the angles from the x-axis $\phi_{\alpha\beta\gamma}$ and $\phi_{\alpha\beta\gamma'}$ are the same which can be obtained by the following equation,

$$\text{atan2}(y_\beta, x_\alpha) \quad (33)$$

Therefore, the spherical coordinate system can be rewritten as $(r_{\alpha\beta\gamma}, \theta_{\alpha\beta\gamma}, \phi_{\alpha\beta})$. Let the image function f be piecewise constant for which f has the same value on the same voxel. Hence, the 3D Zernike moment $Z_{n\ell}^m$ can be approximated by the following discrete form:

$$\begin{aligned} Z_{n\ell}^m &\approx \widehat{Z}_{n\ell}^m \\ &= \frac{(2n+3)\delta^3}{4\pi} \sum_{\alpha} \sum_{\beta} \sum_{\gamma} f(x_\alpha, y_\beta, z_\gamma) R_{n\ell}(r_{\alpha\beta\gamma}) \widetilde{P}_\ell^m(\cos \theta_{\alpha\beta\gamma}) e^{-im\phi_{\alpha\beta}} \end{aligned} \quad (34)$$

$$= \frac{2(2n+3)}{\pi 3\sqrt{3}N^3} \sum_{\alpha} \sum_{\beta} \sum_{\gamma} f(x_\alpha, y_\beta, z_\gamma) R_{n\ell}(r_{\alpha\beta\gamma}) \widetilde{P}_\ell^m(\cos \theta_{\alpha\beta\gamma}) e^{-im\phi_{\alpha\beta}}$$

where $0 \leq \alpha, \beta, \gamma \leq N - 1$. The algorithms for computing all 3D Zernike moments $Z_{n\ell}^m$ up to order $n \leq n_{\max}$ for the entire 3D image inside the cube $C: [-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}]^3 \subseteq B$ which consists of N^3 voxels are summarized in the following:

Algorithm A:

1. Compute the coefficients of the associated Legendre polynomials (Eq. (7)) and keep

the values in an array.

2. Compute the coefficients of the recursive formula in Eq. (18) and keep the values in an array.
3. For each voxel $v(\alpha, \beta, \gamma)$ in $C: [-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}]^3 \subseteq B$ (where $0 \leq \alpha, \beta, \gamma \leq N-1$):
 - 3.1. Compute $(x_\alpha, y_\beta, z_\gamma) = (\frac{2\alpha+1-N}{\sqrt{3}N}, \frac{2\beta+1-N}{\sqrt{3}N}, \frac{2\gamma+1-N}{\sqrt{3}N})$, vector length $r_{\alpha\beta\gamma}$, and the trigonometric function $\cos \theta_{\alpha\beta\gamma}$.
 - 3.2. Compute the trigonometric functions $\cos m\phi_{\alpha\beta}$, $\sin m\phi_{\alpha\beta}$ (for $m = 0, 1, \dots, n_{max}$) in Algorithm B and store the values in arrays.
 - 3.3. Compute the exponential functions $r_{\alpha\beta\gamma}^k$ (for $k = 0, 1, \dots, n_{max}$) in Algorithm C and store the values into arrays.
 - 3.4. For $\ell = 0, 1, \dots, n_{max}$ and $m = 0, 1, \dots, n_{max}$, compute the normalized associated Legendre polynomials $\widehat{P}_\ell^m(\cos \theta_{\alpha\beta\gamma})$ as in Theorem A.
 - 3.5. For $n = 0, 1, 2, \dots, n_{max}$, $\ell = n, n-2, \dots$ (whenever $\ell \geq 0$) compute the radial polynomials $R_{n\ell}(r_{\alpha\beta\gamma})$ in Theorem D with the initial conditions stated in Eq. (22)-(23).
 - 3.5.1. For $m = 0, 1, 2, \dots, \ell$, use the formula in Eq. (34) to calculate the 3D Zernike moment $\widehat{Z}_{n\ell}^m$.

Due to the conjugation between $Z_{n\ell}^m$ and $Z_{n\ell}^{-m}$ in Eq. (25), it only needs to compute the moments $Z_{n\ell}^m$ with the non-negative repetition m to save computation time and memory requirements since a method based on the Farey sequence can be utilized to reduce the calculation of trigonometric functions $\cos m\phi_{\alpha\beta}$ and $\sin m\phi_{\alpha\beta}$ (for $m = 0, 1, \dots, n_{max}$) [17].

Algorithm B:

Compute the complex exponentials $e^{im\phi} = \cos m\phi + i \sin m\phi$ (where $\phi = \phi_{\alpha\beta}$) as follows

1. Compute the values:

$$e^0 = 1, e^{i\phi} = \cos \phi + i \sin \phi = \frac{x}{r} + i \frac{y}{r}, \tan \phi = \frac{2\beta+1-N}{2\alpha+1-N}$$

$$\text{and } e^{i2\phi} = \cos 2\phi + i \sin 2\phi = \frac{1-\tan^2 \phi}{1+\tan^2 \phi} + i \frac{2 \tan \phi}{1-\tan^2 \phi}$$

2. For $m = 3$ to n_{max} :

$$e^{im\phi} = e^{i2\phi} \cdot e^{i(m-2)\phi}$$

Algorithm C:

Compute the real number exponentials

1. Compute $r^0 = 1$, r and $r^2 = r \cdot r$
2. For $k = 3$ to n_{max} :

$$r^k = r^2 \cdot r^{k-2}$$

Part (3) in Algorithm A is a nested loop where the number of the iterations of outer loop control (three variables α , β , and γ) is N^3 . The complexity of (3.1) is $\mathcal{O}(1)$ while the complexity of (3.2) and (3.2) is $\mathcal{O}(n_{max})$. Since part (3.4) calculates spherical harmonics with a complexity of $\mathcal{O}(n_{max}^2)$, and the computational complexity of the inner loop (3.5) (three variables n , ℓ , and m) is $\mathcal{O}(n_{max}^3)$, the total calculation complexity of (3) in Algorithm A is

$$N^3(\mathcal{O}(1) + \mathcal{O}(n_{max}) + \mathcal{O}(n_{max}^2) + \mathcal{O}(n_{max}^3)) = \mathcal{O}(N^3 n_{max}^3) \quad (35)$$

The calculation complexity of (3) in Algorithm A is $\mathcal{O}(N^3 n_{max}^3)$ and takes the most computational time, which executes sequentially. In the above algorithms, the look-up tables for coefficients, trigonometric functions and exponentials can be built in advance, and the algorithm can be revised slightly to execute in parallel.

5. Computation Speed-up via Group Action

In this section, we introduced the method to speed up the algorithms described previously by using group action. In [23], some concepts of group theory for this purpose were well covered and are valuable for further reference. To compute 3D Zernike moments up to the maximal order n_{max} for a voxelized image with a size of N^3 voxels, Algorithm A needs a computation complexity of $\mathcal{O}(N^3 n_{max}^3)$. To reduce the computational CPU time, we applied the symmetry property to the proposed method on the cube $C: \left[-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right]^3 \subseteq B$ by using finite subgroups of the orthogonal matrix group. The Lie matrix group is defined by

$$O(3, \mathbb{R}) = \{M \in M_{3 \times 3}(\mathbb{R}) : M^t M = \text{Id}\} \quad (36)$$

where $M_{3 \times 3}(\mathbb{R})$ represents a set of 3×3 real number matrices, M^t denotes the transpose of M and Id is the identity matrix. The Lie matrix group $O(3, \mathbb{R})$ consists of rotations and reflections. By the symmetry property, we can replace the set C by a smaller subset via its fundamental domain operated by a suitable group; thereby the computation time can be reduced to some extent.

Let $G \subset O(3, \mathbb{R})$ be a matrix group acting on the set $C: \left[-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right]^3$. For $\sigma \in G$ and $(x, y, z) \in C$, the linear transform σ acts as the matrix multiplication at (x, y, z) :

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \sigma \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (37)$$

Let us consider a non-abelian finite group $G_{16} \subset O(3, \mathbb{R})$ of group order 16 generated by the following three matrices:

$$\tau = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \sigma_x = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \sigma_z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (38)$$

which is denoted by $G_{16} = \langle \tau, \sigma_x, \sigma_z \rangle$. For any voxel $(x, y, z) \in C$ with its spherical coordinates (r, θ, φ) , we have:

$$\begin{aligned} \tau \begin{bmatrix} x \\ y \\ z \end{bmatrix} &= \begin{bmatrix} -y \\ x \\ z \end{bmatrix} \text{ i.e. } \tau(r, \theta, \varphi) = \left(r, \theta, \varphi + \frac{\pi}{2}\right) \\ \sigma_x \begin{bmatrix} x \\ y \\ z \end{bmatrix} &= \begin{bmatrix} -x \\ y \\ z \end{bmatrix} \text{ i.e. } \sigma_x(r, \theta, \varphi) = (r, \theta, \pi - \varphi) \\ \sigma_z \begin{bmatrix} x \\ y \\ z \end{bmatrix} &= \begin{bmatrix} x \\ y \\ -z \end{bmatrix} \text{ i.e. } \sigma_z(r, \theta, \varphi) = (r, \pi - \theta, \varphi) \end{aligned} \quad (39)$$

To explicit the use of group action by the matrix group G_{16} , we summarize some useful properties in the following theorem; most of them can be verified directly by matrix multiplication calculation.

Theorem E. The notations as stated above. The following properties hold:

1. $G_{16}(C) \subset C$, i.e. for any $\sigma \in G_{16}$ and for any voxel P in C , it holds $\sigma(P) \in C$.
2. The linear transforms

$$\sigma_y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \iota_{xy} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (40)$$

are two elements of G_{16} . In fact, $\sigma_y = \tau^2 \sigma_x = \sigma_x \tau^2$ maps (x, y, z) to $(x, -y, z)$ and $\iota_{xy} = \sigma_x \tau = \tau^3 \sigma_x$ maps (x, y, z) to (y, x, z) . That is, $\sigma_y(r, \theta, \varphi) = (r, \theta, 2\pi - \varphi)$ and $\iota_{xy}(r, \theta, \varphi) = \left(r, \theta, \frac{\pi}{2} - \varphi\right)$ in spherical coordinates.

3. The matrices ι_{xy} , σ_x , σ_y , and σ_z are involutions. Furthermore, every element $\sigma \in G_{16}$ can be expressed as the product of these involutions. i.e.

$$\sigma = \iota_{xy}^{\epsilon_{\iota}} \sigma_x^{\epsilon_x} \sigma_y^{\epsilon_y} \sigma_z^{\epsilon_z} \text{ where } \epsilon_{\iota}, \epsilon_x, \epsilon_y, \epsilon_z \in \{0, 1\} \quad (41)$$

Figure 1. The fundamental domain H for the action of the group G_{16}

Proof. A proof sketch for (3) of Theorem E was given. By direct matrix multiplication, it follows that

$$\iota_{xy}^2 = \sigma_x^2 = \sigma_y^2 = \sigma_z^2 = Id \quad (42)$$

Let G' be the group generated by those four involutions. By (2) of Theorem E, it concludes that these four involutions are elements of G_{16} , so $G' \subseteq G_{16}$. Since both groups have the same group order, $G' = G_{16}$.

Q.E.D.

The cube C can be equally divided into sixteen parts. In Figure 1, the H is defined as follows:

$$H = \{(x, y, z) \in C \mid 0 \leq y \leq x \text{ and } z \geq 0\} \quad (43)$$

which is the triangular prism bounded by the following five planes:

$$z = 0, z = \frac{1}{\sqrt{3}}, x = \frac{1}{\sqrt{3}}, y = 0 \text{ and } x = y \quad (44)$$

Let H be the fundamental domain for the action of the group G_{16} . Thus, the cube C can be expressed by the union of 16 subsets as follows:

$$C = \bigcup_{\sigma \in G_{16}} \sigma H \quad (45)$$

To calculate the 3D Zernike moments of the object inside the cube C , we only need to compute all voxels inside the H , and the calculation of voxels inside other than the H can be saved by the use of the symmetries generated by the G_{16} group action. For any voxel $v(\alpha, \beta, \gamma)$ with center $(x_\alpha, y_\beta, z_\gamma)$ inside the H , they can be indexed by the following set:

$$I_H = \left\{ (\alpha, \beta, \gamma) \mid \frac{N}{2} \leq \alpha, \beta, \gamma \leq N-1 \text{ and } \alpha \geq \beta \right\} \quad (46)$$

For a voxel p in C , the orbit of p under the finite group action by G is denoted by $G(p)$, which is the set of all its G conjugations, i.e.

$$G(P) = \{\sigma(p) \mid \sigma \in G\} \quad (47)$$

The stabilizer of p in G is denoted by $\text{Stab}_G(p)$, which is the subgroup of G consisting of the transformation σ such that $\sigma(p) = p$, i.e.

$$\text{Stab}_G(p) = \{\sigma \in G \mid \sigma(p) = p\} \quad (48)$$

Let $|\cdot|$ denote the cardinality of a set, it holds that

$$|\text{Stab}_G(p)| |G(p)| = |G| \quad (49)$$

The visual representation of the G_{16} -orbit of a voxel inside the H is shown in Figure 2.

Figure 2. The G_{16} -orbit of a voxel inside the H has 16 elements. There are eight red voxels in the upper half generated by the group $\langle \tau, \sigma_x \rangle$. The eight green voxels in the lower half are acted by σ_z , which are the reflections of the eight red points in the upper half with regarding $z = 0$.

Of note, there exists some voxels p in C such that $\text{Stab}_{G_{16}}(p) \neq \{\text{Id}\}$. In that case, each voxel in the orbit $G_{16}(p)$ would be repeatedly calculated $|\text{Stab}_{G_{16}}(p)|$ times, when every $\sigma \in G_{16}$ is applied to the voxel p . For instance, the origin has stabilizer $\text{Stab}_{G_{16}}(0,0,0) = G_{16}$ and its G_{16} -orbit is $G_{16}(0,0,0) = \{(0,0,0)\}$. For another example, one face of the fundamental domain H excluding the line $\{(0,0,z)\}$ is described by the following set:

$$\{(x, y, z) \in H \mid x = y, x \neq 0\} \quad (50)$$

Each voxel in the set has its associated stabilizer $\langle \iota_{xy} \rangle$ that is a group of order two, and its G_{16} -orbit is composed of 8 distinct points:

$$G_{16}(x, x, z) = \{(\pm x, \pm x, \pm z)\} \quad (51)$$

Due to the diversity, we need more caution when using G_{16} group actions to speed up Algorithm A. All conjugations in $G_{16}(x_\alpha, y_\beta, z_\gamma)$ have the same radial function $R_{n\ell}(r_{\alpha\beta\gamma}) = R_{n\ell} \circ \sigma(r_{\alpha\beta\gamma})$ for any $\sigma \in G_{16}$. With the property in Eq. (49), the calculation of 3D Zernike moments in Eq. (34) can be rewritten as follows:

$$Z_{n\ell}^m \approx \widehat{Z}_{n\ell}^m = \frac{2(2n+3)}{\pi 3\sqrt{3}N^3} \sum_{(\alpha,\beta,\gamma) \in I_H} \frac{R_{n\ell}(r_{\alpha\beta\gamma})}{|\text{Stab}(x_\alpha, y_\beta, z_\gamma)|} \sum_{\sigma \in G_{16}} f \circ \sigma(x_\alpha, y_\beta, z_\gamma) \widehat{P}_\ell^m(\cos \sigma \theta_{\alpha\beta\gamma}) e^{-im\sigma\phi_{\alpha\beta}} \quad (52)$$

The most of voxels $v(\alpha, \beta, \gamma)$ with centers $(x_\alpha, y_\beta, z_\gamma)$ inside H have the identity stabilizer in G_{16} , except for those voxels $v(\alpha, \beta, \gamma)$ with $\alpha = \beta$. Therefore, the index set I_H can be represented by the union of two index sets I_{H_0} and I_D where

$$I_{H_0} = \left\{ (\alpha, \beta, \gamma) \mid \frac{N}{2} \leq \alpha, \beta, \gamma \leq N-1 \text{ and } \alpha > \beta \right\} \quad (53)$$

$$I_D = \left\{ (\alpha, \beta, \gamma) \mid \frac{N}{2} \leq \alpha, \beta, \gamma \leq N-1 \text{ and } \alpha = \beta \right\}$$

The stabilizer $\text{Stab}(x_\alpha, y_\beta, z_\gamma)$ of any voxel with center $(x_\alpha, y_\beta, z_\gamma)$ for $(\alpha, \beta, \gamma) \in I_D$ is the cyclic group of order two $\langle t_{xy} \rangle$ (Cf. Eq. (50)-(51)). Therefore, the formula in Eq. (52) can be furthermore rewritten as follows:

$$\begin{aligned} \widehat{Z}_{n\ell}^m = & \frac{2(2n+3)}{\pi 3\sqrt{3}N^3} \left(\left(\sum_{(\alpha,\beta,\gamma) \in I_{H_0}} R_{n\ell}(r_{\alpha\beta\gamma}) \sum_{\sigma \in G_{16}} f \circ \sigma(x_\alpha, y_\beta, z_\gamma) \widehat{P}_\ell^m(\cos \sigma \theta_{\alpha\beta\gamma}) e^{-im\sigma\phi_{\alpha\beta}} \right) \right. \\ & + \frac{1}{2} \left(\sum_{(\alpha,\beta,\gamma) \in I_D} R_{n\ell}(r_{\alpha\beta\gamma}) \sum_{\sigma \in G_{16}} f \circ \sigma(x_\alpha, y_\beta, z_\gamma) \widehat{P}_\ell^m(\cos \sigma \theta_{\alpha\beta\gamma}) e^{-im\sigma\phi_{\alpha\beta}} \right) \left. \right) \end{aligned} \quad (54)$$

To calculate Eq. (54), we use Eq. (8) and Theorem E (especially Eq. (41)) and then represent $f \circ \sigma(x, y, z)$, $\widehat{P}_\ell^m(\cos \sigma \theta)$, and $e^{-im\sigma\phi}$ (for $\sigma \in G_{16}$) as the explicit computational forms listed in Table 1. For achieving the whole computation for $\widehat{Z}_{n\ell}^m$,

we need to consider the following exponential sum

$$S = \sum_{\sigma \in G_{16}} f \circ \sigma(x, y, z) \widetilde{P}_\ell^m(\cos \sigma\theta) e^{-im\sigma\phi} \quad (55)$$

The computation of the exponential sum S has only four cases to be considered with $m \equiv 0, 1, 2, 3 \pmod{4}$. By the use of Table 1, the calculations are listed in Table 2.

All the calculations of applying the G_{16} group actions to Algorithm A have been presented. The execution time of the proposed procedure can theoretically accelerate the algorithm by a factor of 16. The speed-up results of Algorithm A in the experiments we achieved are more or less consistent with this theoretical number.

Table 1. The explicit form of $f \circ \sigma(x, y, z)$, $\widetilde{P}_\ell^m(\cos \sigma\theta)$ and $e^{-im\sigma\phi}$ for any $\sigma \in G_{16}$

σ	$f \circ \sigma(x, y, z)$	$\widetilde{P}_\ell^m(\cos \sigma\theta)$	$e^{-im\sigma\phi}$
Id	$f_0 = f(x, y, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$\cos m\phi - i \sin m\phi$
σ_x	$f_1 = f(-x, y, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$(-1)^m(\cos m\phi + i \sin m\phi)$
$\sigma_x \sigma_y$	$f_2 = f(-x, -y, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$(-1)^m(\cos m\phi - i \sin m\phi)$
σ_y	$f_3 = f(x, -y, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$\cos m\phi + i \sin m\phi$
σ_z	$f_4 = f(x, y, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$\cos m\phi - i \sin m\phi$
$\sigma_x \sigma_z$	$f_5 = f(-x, y, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$(-1)^m(\cos m\phi + i \sin m\phi)$
$\sigma_x \sigma_y \sigma_z$	$f_6 = f(-x, -y, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$(-1)^m(\cos m\phi - i \sin m\phi)$
$\sigma_y \sigma_z$	$f_7 = f(x, -y, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$\cos m\phi + i \sin m\phi$
ι_{xy}	$f_8 = f(y, x, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$(-i)^m(\cos m\phi + i \sin m\phi)$
$\iota_{xy} \sigma_x$	$f_9 = f(-y, -x, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$i^m(\cos m\phi - i \sin m\phi)$
$\iota_{xy} \sigma_x \sigma_y$	$f_{10} = f(-y, -x, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$i^m(\cos m\phi + i \sin m\phi)$
$\iota_{xy} \sigma_y$	$f_{11} = f(-y, x, z)$	$\widetilde{P}_\ell^m(\cos \theta)$	$(-i)^m(\cos m\phi - i \sin m\phi)$
$\iota_{xy} \sigma_z$	$f_4 = f(y, x, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$(-i)^m(\cos m\phi + i \sin m\phi)$
$\iota_{xy} \sigma_x \sigma_z$	$f_5 = f(y, -x, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$i^m(\cos m\phi - i \sin m\phi)$
$\iota_{xy} \sigma_x \sigma_y \sigma_z$	$f_6 = f(-y, -x, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$i^m(\cos m\phi + i \sin m\phi)$
$\iota_{xy} \sigma_y \sigma_z$	$f_7 = f(-y, x, -z)$	$(-1)^{\ell-m} \widetilde{P}_\ell^m(\cos \theta)$	$(-i)^m(\cos m\phi - i \sin m\phi)$

Table 2. The computation of the exponential sum S for $m \equiv 0, 1, 2, 3 \pmod{4}$

$m \pmod{4}$	$S = \sum_{\sigma \in G_{16}} f \circ \sigma(x, y, z) \widetilde{P}_\ell^m(\cos \sigma\theta) e^{-im\sigma\phi}$
--------------	--

$m \equiv 0$	$\begin{aligned} \widehat{P}_\ell^m(\cos \theta) & \left(\cos m\phi \left(f_0 + f_1 + f_2 + f_3 + f_8 + f_9 + f_{10} + f_{11} \right. \right. \\ & \quad \left. \left. + (-1)^\ell (f_4 + f_5 + f_6 + f_7 + f_{12} + f_{13} + f_{14} + f_{15}) \right) \right. \\ & \quad \left. + i \sin m\phi \left(-f_0 + f_1 - f_2 + f_3 + f_8 - f_9 + f_{10} - f_{11} \right. \right. \\ & \quad \left. \left. + (-1)^\ell (-f_4 + f_5 - f_6 + f_7 + f_{12} - f_{13} + f_{14} - f_{15}) \right) \right) \end{aligned}$
$m \equiv 1$	$\begin{aligned} \widehat{P}_\ell^m(\cos \theta) & \left(\cos m\phi \left(f_0 - f_1 - f_2 + f_3 + (-1)^\ell (-f_4 + f_5 + f_6 - f_7) \right) \right. \\ & \quad \left. + \sin m\phi \left(f_8 + f_9 - f_{10} - f_{11} + (-1)^\ell (-f_{12} - f_{13} + f_{14} + f_{15}) \right) \right. \\ & \quad \left. + i \left(\sin m\phi \left(-f_0 - f_1 + f_2 + f_3 + (-1)^\ell (f_4 + f_5 - f_6 - f_7) \right) \right. \right. \\ & \quad \left. \left. + \cos m\phi \left(-f_8 + f_9 + f_{10} - f_{11} \right. \right. \right. \\ & \quad \left. \left. \left. + (-1)^\ell (f_{12} - f_{13} - f_{14} + f_{15}) \right) \right) \right) \end{aligned}$
$m \equiv 2$	$\begin{aligned} \widehat{P}_\ell^m(\cos \theta) & \left(\cos m\phi \left(f_0 + f_1 + f_2 + f_3 - f_8 - f_9 - f_{10} - f_{11} \right) \right. \\ & \quad \left. + (-1)^\ell (f_4 + f_5 + f_6 + f_7 - f_{12} - f_{13} - f_{14} - f_{15}) \right) \\ & \quad + i \sin m\phi \left(-f_0 + f_1 - f_2 + f_3 - f_8 + f_9 - f_{10} + f_{11} \right. \\ & \quad \left. + (-1)^\ell (-f_4 + f_5 - f_6 + f_7 - f_{12} + f_{13} - f_{14} + f_{15}) \right) \end{aligned}$
$m \equiv 3$	$\begin{aligned} \widehat{P}_\ell^m(\cos \theta) & \left(\cos m\phi \left(f_0 - f_1 - f_2 + f_3 + (-1)^\ell (-f_4 + f_5 + f_6 - f_7) \right) \right. \\ & \quad \left. + \sin m\phi \left(-f_8 - f_9 + f_{10} + f_{11} + (-1)^\ell (f_{12} + f_{13} - f_{14} - f_{15}) \right) \right. \\ & \quad \left. + i \left(\sin m\phi \left(-f_0 - f_1 + f_2 + f_3 + (-1)^\ell (f_4 + f_5 - f_6 - f_7) \right) \right. \right. \\ & \quad \left. \left. + \cos m\phi \left(f_8 - f_9 - f_{10} + f_{11} \right. \right. \right. \\ & \quad \left. \left. \left. + (-1)^\ell (-f_{12} + f_{13} + f_{14} + f_{15}) \right) \right) \right) \end{aligned}$

6. Shape Symmetry Coefficient and Acceleration Factor

In this section, the 3D shape symmetry coefficient is used to measure the further acceleration of the Zernike moment calculation; the acceleration factor is used to estimate the speed-up performance of algorithms. In Algorithm A, we computed all voxels inside the cube $C: [-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}]^3 \subseteq B$. It saves the unnecessary iterations when only the non-zero value voxels are calculated. The modification is called Algorithm A'. Let V_{nz} be the set of non-zero value voxels for which $\bigcup_{p \in V_{nz}} p \subseteq C \subseteq B$. For a finite

orthogonal subgroup $G \subseteq O(3, \mathbb{R})$, there is the G -closure of V_{nz} given by

$$G(V_{nz}) = \{\sigma(P) \mid \sigma \in G \text{ and } P \in V_{nz}\}. \quad (56)$$

It is larger than the set V_{nz} . To measure the degree of shape symmetry under the group action by G , we define the G -symmetry coefficient as follows:

$$SC(G, V_{nz}) = \frac{|V_{nz}|}{|G(V_{nz})|}. \quad (57)$$

The shape V_{nz} is intrinsic symmetry when $SC(G, V_{nz})=1$. However, the set V_{nz} does not have G -symmetry when $SC(G, V_{nz})=1/|G|$. The index sets I_{H_0} and I_D in Eq. (53) can be replaced with the smaller sets J_{H_0} and J_D :

$$J_{H_0} = \left\{ (\alpha, \beta, \gamma) \in G_{16}(V_{nz}) \mid \frac{N}{2} \leq \alpha, \beta, \gamma \leq N-1 \text{ and } \alpha > \beta \right\} \quad (58)$$

$$J_D = \left\{ (\alpha, \beta, \gamma) \in G_{16}(V_{nz}) \mid \frac{N}{2} \leq \alpha, \beta, \gamma \leq N-1 \text{ and } \alpha = \beta \right\}$$

When applying G_{16} -conjugation for speeding up the calculation of Algorithm A', the computation of $\widehat{Z}_{n\ell}^m$ in Eq. (54) is furthermore rewritten as follows:

$$\begin{aligned} \widehat{Z}_{n\ell}^m = & \frac{2(2n+3)}{\pi 3\sqrt{3}N^3} \left(\left(\sum_{(\alpha, \beta, \gamma) \in J_{H_0}} R_{n\ell}(r_{\alpha\beta\gamma}) \sum_{\sigma \in G_{16}} f \right. \right. \\ & \left. \left. \circ \sigma(x_\alpha, y_\beta, z_\gamma) \widehat{P}_\ell^m(\cos \sigma\theta_{\alpha\beta\gamma}) e^{-im\sigma\phi_{\alpha\beta}} \right) \right. \\ & \left. + \frac{1}{2} \left(\sum_{(\alpha, \beta, \gamma) \in J_D} R_{n\ell}(r_{\alpha\beta\gamma}) \sum_{\sigma \in G_{16}} f \right. \right. \\ & \left. \left. \circ \sigma(x_\alpha, y_\beta, z_\gamma) \widehat{P}_\ell^m(\cos \sigma\theta_{\alpha\beta\gamma}) e^{-im\sigma\phi_{\alpha\beta}} \right) \right) \end{aligned} \quad (59)$$

There is one-to-one correspondence between the set J ($\equiv J_{H_0} \cup J_D$) and the set of G_{16} -orbits in $G_{16}(V_{nz})$. Thus, the acceleration factor under G_{16} is given by

$$ACC(V_{nz}) = \frac{16|V_{nz}|}{|G_{16}(V_{nz})|} \quad (60)$$

The $ACC(V_{nz})$ value depends on the shape of 3D image and is slightly smaller than the $16 \times SC(G_{16}, V_{nz})$ value that is the theoretical value of the acceleration factor of the

G_{16} -conjugation applied to the Algorithm A'.

7. Experimental Results and Discussions

7.1. Experimental Setup

These experiments were performed on a personal computer with 32 GB of RAM, 3 leveled 8 MB cache and Intel i7-6700K quad-core 4.0 GHz central processors. The proposed methods were implemented in C/C++ codes with the TDM-GCC 4.9.2 64-bit release compiler with setting of O2 optimization under the Windows 10 operating system. The automatic parallelism is disabled. The program uses IEEE 754 64-bit double-precision floating-point format for all real and complex computations and runs in serial. We used the isosurface function in MatLab to compute isosurface data from the volume data to form a displayable image.

Three voxelized 3D images Angel, Buddha, and Michelangelo's David (the STL format files in <https://www.thingiverse.com>) with a size of $150 \times 150 \times 150$ voxels are shown in Table 3 with parameters whose definition were described in the previous sections. In the implementation of Algorithm A' with the acceleration of G_{16} -conjugation, the index set J, or equivalently the set of G_{16} -orbits in $G_{16}(V_{nz})$, is constructed before calculating Eq. (59) to run the algorithm smoothly.

Table 3. Three voxelized 3D images Angel, Buddha, and Michelangelo's David with a size of $150 \times 150 \times 150$ voxels were shown with their parameters on the bottom half of the table. NZV denotes the number of non-zero voxels, $|G_{16}|$ is the number of $G_{16}(V_{nz})$, SC is the G_{16} -symmetry coefficient, $|J|$ is the number of G_{16} -orbits of $G_{16}(V_{nz})$, and ACC represents the acceleration factor under G_{16} .

	Angel	Buddha	Michelangelo's David
Voxelized image
	NZV	606492	630579
	$ G_{16} $	1537448	1158168
	SC	0.3945	0.5445
	$ J $	97591	73580
			53357

ACC	6.2146	8.5700	8.0803
-----	--------	--------	--------

7.2. Numerical Accuracy

There are four methods for calculating 3D Zernike moments with different settings for the performance comparison are listed in Table 4.

Table 4. *Methods used in the simulation.*

Notation	Method
A_0	Algorithm A
A_{nz}	Algorithm A'
A_{16}	Algorithm A and using G_{16} -conjugation
$A_{nz,16}$	Algorithm A' and using G_{16} -conjugation

With the proposed method, when the maximum moment order n_{max} becomes larger (up to 450), the *NMSE* (recovery error rate) of the shape reconstruction becomes smaller rapidly for the 3D test images as show in Figure 4. We also used algorithm provided by Novotni and Klein in [1, 2] and applied it to the image Michelangelo's David. As reported in [12], when the maximum order is 20, the absolute value of the geometric moment coefficient ranges from 10^0 to 10^9 , and the geometric moments also have a great dynamic range. The calculation error accumulates very quickly and is directly reflected in the shape reconstruction at $n_{max} = 20$, as shown in Figure 3. The result of the shape reconstruction has many connected components, most of which are caused by computational errors except for the middle one. When the order is greater than 20, the algorithm reconstructs the shape is very time-consuming.

Figure 3. Three different angle views for the reconstruction of Michelangelo's David at $n_{max} = 20$ by using the algorithm of Novotni and Klein.

To evaluate the performance of numerical precision, we first calculated the 3D Zernike moments for the 3D voxelized images in the experiments and then used those moments to reconstruct the 3D shapes. The two types of differences between the original shape and the reconstructed, ε^2 (Eq. (29)) and $\bar{\varepsilon}$ (Eq. (30)), are calculated to

assess the numerical accuracy of the reconstruction. Figure 4 shows that the $NMSE \ \varepsilon^2$ for the shape reconstructions of Angel, Buddha, and Michelangelo's David 3D test images have similar error behaviors. The values of ε^2 of the shape reconstructions keep decreasing up to the maximum order = 450, which are 0.003831291, 0.003683659, and 0.0041664, respectively.

Figure 4. The Log-plots of $NMSE \ \varepsilon^2$ of the shape reconstructions for the three test images at the different order $n_{max} = 0, 10, 20, \dots, 150, 200, 250, \dots, 450$.

Table 5 lists the recovery error rate $\bar{\varepsilon}$ of the proposed method applying to these three test images. The value $\bar{\varepsilon}$ of every image has similar behavior and is zero at order $n_{max} = 450$. That is, the reconstructed shape of each test image at the maximal order 450 is identical to the original image.

Table 5. The list of recovery error rate $\bar{\varepsilon}$ of the reconstructed shape for the three test images at the different order $n_{max} = 10, 50, 100, 150, \dots, 450$.

Order	David	Buddha	Angel
10	0.281100	0.261818	0.256663
50	0.053646	0.033509	0.044719
100	0.025119	0.016826	0.018284
150	0.013678	0.008308	0.009435
200	0.007295	0.003367	0.004693
250	0.003229	0.001354	0.002008
300	0.000844	0.000333	0.000613
350	0.000141	5.71E-05	0.000104
400	2.32E-06	4.76E-06	8.24E-06
450	0	0	0

Table 6 shows three different angle views of reconstructed shapes using different Zernike moment orders. As mentioned before, the low-order Zernike moments describe the global contour of the shape; the higher-order moments describe the local detail.

Table 6. The reconstructed shapes of Michelangelo's David by using 3D Zernike moments up to the maximal order $n_{max} = 20, 50, 200, 450$. $NMSE \ \varepsilon^2$ and recovery error rate $\bar{\varepsilon}$ shown at the bottom of each row.

Order	Three different angle views		
20
	$\varepsilon^2 = 0.162109112, \bar{\varepsilon} = 0.151959215$

50
	$\varepsilon^2 = 0.073918465, \bar{\varepsilon} = 0.053645899$

200	$\varepsilon^2 = 0.019460496, \bar{\varepsilon} = 0.00729458$		

450

$$\varepsilon^2 = 0.0041664, \bar{\varepsilon} = 0$$

7.3. Computation Speed

The methods A_0 and A_{16} are performed for every voxel $v(\alpha, \beta, \gamma)$ with center $(x_\alpha, y_\beta, z_\gamma)$ inside $C(H)$. Therefore, the elapsed times for all three test 3D images differ only slightly. Table 7 only shows the elapsed times of applying the methods A_0 and A_{16} to the voxelized 3D Angel image. The value of speed-up entitled A_0/A_{16} and displayed in the fourth column in Table 7 is between 14.125 and 16.544, which is very close to the theoretical value 16.

Table 7. The elapsed time of applying the methods A_0 and A_{16} to the image Angel. The speed-up (A_0/A_{16}) is the ratio of the elapsed time of method A_0 over method A_{16} .

Order	A_0	A_{16}	A_0/A_{16}
10	6.452	0.390	16.544
50	86.276	6.108	14.125
100	464.625	31.899	14.566
150	2015.637	134.796	14.953
200	5690.229	374.069	15.212
450	55396.032	3543.462	15.633

Unit of elapsed times: second.

The elapsed time of method A_{nz} and method $A_{nz,16}$ is proportional to the numbers of non-zero value voxels and the value of J , respectively. The experimental results for the methods A_{nz} and $A_{nz,16}$ applied to the three images are listed in Table 8 and Table 9 respectively. For different Zernike moment orders, the ratios of the elapsed time over the non-zero value voxels (elapsed time/ $|J|$) shown in the same row in Table 8 (Table 9) are only slightly different. Since the values of NZV and $|J|$ are equivalent to the total number of iterations processed in the two algorithms respectively, the elapsed times of these two methods are highly related to the total number of iterations in the algorithm.

Table 8. The elapsed times of method A_{nz} applying to the image Angel, Buddha, and Michelangelo's David with computing moments up to maximal order $n_{max} =$

0, 10, 50, 100, 150, 200, 450. The ratio of the elapsed time over NZV value is also shown. The NZV values of the three iamges are 606492, 630579, and 431142, respectively (Table 3).

Order	Elapsed time			Elapsed time / NZV		
	Angel	Buddha	David	Angel	Buddha	David
10	1.297	1.421	0.968	2.1385E-06	2.2535E-06	2.2452E-06
50	15.777	15.918	10.951	2.6014E-05	2.5244E-05	2.5400E-05
100	83.152	86.698	59.877	1.3710E-04	1.3749E-04	1.3888E-04
150	356.448	369.976	253.613	5.8772E-04	5.8672E-04	5.8824E-04
200	1015.120	1050.080	721.144	1.6738E-03	1.6653E-03	1.6726E-03
450	9773.960	10191.002	6981.083	1.6116E-02	1.6161E-02	1.6192E-02

Unit of time: second.

Table 9. The elapsed times of method $A_{nz,16}$ applying to the image Angel, Buddha, and Michelangelo's David computing moments up to maximal order $n_{max}=10, 50, 100, 150, 200, 450$ with the ratio of the elapsed time over the value of $||J||$. The values of $||J||$ of the three images are 97591, 73580, and 53357, respectively.

Order	Elapsed time			Elapsed time / $ J $		
	Angel	Buddha	David	Angel	Buddha	David
10	0.187	0.172	0.110	1.9162E-06	2.3376E-06	2.0616E-06
50	2.749	2.077	1.718	2.8169E-05	2.8228E-05	3.2198E-05
100	14.903	11.170	8.810	1.5271E-04	1.5181E-04	1.6511E-04
150	63.141	47.114	34.742	6.4700E-04	6.4031E-04	6.5112E-04
200	176.568	130.985	96.040	1.8093E-03	1.7802E-03	1.8000E-03
450	1638.303	1235.695	896.056	1.6787E-02	1.6794E-02	1.6794E-02

Unit of time: second.

From the experimental results in Table 9, as long as the $||J||$ value of both the reference image and the query image is known in advance, we can estimate the elapsed time of the Zernike moments calculation for the query 3D image based on the elapsed time of calculating Zernike moments taken by the reference image for a given maximal order. The elapsed time of the method $A_{nz,16}$ applied to the image Angel at the maximal order 450 takes 1638.303 seconds. The calculation of the proposed method outperforms other methods in terms of speed and accuracy in the literature.

Comparing the speed-up of method $A_{nz,16}$ relative to method A_{nz} with the acceleration factor calculated in Table 3, we list the ratio of the elapsed time of method A_{nz} over the one of $A_{nz,16}$ in Table 10. The experiment shows that the speed-up results are not far from the expected values.

Table 10. The value of speed-up of method A_{nz} and $A_{nz,16}$ for the three test images. The values of the acceleration factor under $G_{16}(\text{ACC})$ of the three images are 6.2146, 8.5700, and 8.0803, respectively (Table 3).

Order	Angel	Buddha	David
10	6.9358	8.2616	8.8000
50	5.7392	7.6639	6.3743
100	5.5795	7.7617	6.7965
150	5.6453	7.8528	7.3000
200	5.7492	8.0168	7.5090
450	5.9659	8.2472	7.7909

Table 11 lists the elapsed times at different orders for using the method $A_{nz,16}$ and the method A_0 (Algorithm A) applied to the three images. The speed-up value of method $A_{nz,16}$ compared to A_0 (Algorithm A) lies between 31.177 and 61.822. The simulation result shows excellent improvement in the calculation. In particular, 3D images have a small number of non-zero voxels and have a large ACC value. This type of image will make the $||J||$ value smaller, which will speed up the 3D Zernike moments calculation.

Table 11. The elapsed times of method A_0 and method $A_{nz,16}$ applying to Angel, Buddha, and Michelangelo's David three images are listed. The speed-up value of $A_{nz,16}$ relative to A_0 is also shown.

Order	Elapsed time of method $A_{nz,16}$			Elapsed time of method A_0	Speed-up ($A_0/A_{nz,16}$)		
	Angel	Buddha	David		Angel	Buddha	David
10	0.187	0.172	0.110	6.452	34.503	37.512	58.655
50	2.749	2.077	1.718	86.276	31.385	41.539	50.219
100	14.903	11.170	8.810	464.625	31.177	41.596	52.738
150	63.141	47.114	34.742	2015.637	31.923	42.782	58.017
200	176.568	130.985	96.040	5690.229	32.227	43.442	59.249
450	1638.303	1235.695	896.056	55396.032	33.813	44.830	61.822

Unit of time: second.

8. Conclusions

The proposed method can achieve accurate numerical results, especially for high-order moments. Applying the group actions of the finite orthogonal group G_{16} can significantly reduce the computational cost and accelerate the calculation. The experimental results showed that the proposed method $A_{nz,16}$ (Algorithm A' and using G_{16} -conjugation) took 896.056 seconds to compute the top 450-order 3D Zernike

moments of Michelangelo's David with a size of $150 \times 150 \times 150$ voxels. If the shape is reconstructed using the Zernike moments up to order $n_{max} = 450$, the normalized mean square error (NMSE) ε^2 is 0.0041664, and the recovery error rate $\bar{\varepsilon}$ is 0. When compared to the proposed method A_0 (Algorithm A without any acceleration), the acceleration factor of applying method ANZ_{16} to Michelangelo's David is between 50.219 and 61.822 when calculating moments at the maximal order between 10 and 450.

This proposed method is superior to other compared voxel-based methods in terms of speed and accuracy when calculating high-order 3D Zernike moments. The algorithm is also suitable for applying 3D Zernike transform to 3D objects containing voxel intensities. For the future, the application of this analysis would be explored, for example, potentially assessing the morphological changes of biological structures between healthy and pathology, including not only the shape but also texture.

ACKNOWLEDGMENTS

This work was supported by Ministry of Science and Technology, Taiwan, under research project number MOST 108-2115-M-231-001 and MOST 108-2221-E-231-004.

REFERENCES

1. Novotni, M. and R. Klein. *3D Zernike descriptors for content based shape retrieval*. in *Proceedings of the eighth ACM symposium on Solid modeling and applications*. 2003. ACM.
2. Novotni, M. and R. Klein, *Shape retrieval using 3D Zernike descriptors*. Computer-Aided Design, 2004. **36**: p. 1047-1062.
3. Wang, K., T. Zhu, and J. Wang, *Real-time Terrain Matching Based on 3D Zernike Moments*. The Journal of Navigation, 2018. **71**(6): p. 1441-1459.
4. Wang, K., et al., *Efficient terrain matching with 3-D Zernike moments*. IEEE Transactions on Aerospace and Electronic Systems, 2018. **55**(1): p. 226-235.
5. Tangelder, J.W. and R.C. Veltkamp. *A survey of content based 3D shape retrieval methods*. in *Proceedings Shape Modeling Applications, 2004*. 2004. IEEE.
6. Ji, X. and H. Liu, *Advances in view-invariant human motion analysis: a review*. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2009. **40**(1): p. 13-24.
7. Yan, H., Z. Zhenwen, and L. Jianfeng. *Human action classification based on the 3D Zernike moment*. in *2014 Sixth International Conference on Measuring Technology and Mechatronics Automation*. 2014. IEEE.

8. Daberdaku, S. and C. Ferrari, *Exploring the potential of 3D Zernike descriptors and SVM for protein-protein interface prediction*. BMC bioinformatics, 2018. **19**(1): p. 35.
9. Canterakis, N. *3D Zernike moments and Zernike affine invariants for 3D image analysis and recognition*. in *In 11th Scandinavian Conf. on Image Analysis*. 1999. Citeseer.
10. Canterakis, N., *Fast 3d zernike moments and-invariants*. 1997.
11. Callahan, P. and M. De Graef, *Precipitate shape fitting and reconstruction by means of 3D Zernike functions*. Modelling and Simulation in Materials Science and Engineering, 2011. **20**(1): p. 015003.
12. Berjón, D., S. Arnaldo, and F. Morán. *A parallel implementation of 3D Zernike moment analysis*. in *Parallel Processing for Imaging Applications*. 2011. International Society for Optics and Photonics.
13. Zhang, L., et al., *Survey on 3D shape descriptors*. Fundagao para a Cincia ea Tecnologia, Lisboa, Portugal, Tech. Rep. Technical Report, DecorAR (FCT POSC/EIA/59938/2004), 2007. **3**.
14. Pozo, J.M., M.-C. Villa-Uriol, and A.F. Frangi, *Efficient 3D geometric and Zernike moments computation from unstructured surface meshes*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010. **33**(3): p. 471-484.
15. Koehl, P., *Fast recursive computation of 3D geometric moments from surface meshes*. IEEE transactions on pattern analysis and machine intelligence, 2012. **34**(11): p. 2158-2163.
16. Singh, C. and E. Walia, *Fast and numerically stable methods for the computation of Zernike moments*. Pattern Recognition, 2011. **44**(4): p. 996-997.
17. Deng, A.-W., C.-H. Wei, and C.-Y. Gwo, *Stable, fast computation of high-order Zernike moments using a recursive method*. Pattern Recognition, 2016. **56**: p. 16-25.
18. Venkatraman, V., L. Sael, and D. Kihara, *Potential for protein surface shape analysis using spherical harmonics and 3D Zernike descriptors*. Cell biochemistry and biophysics, 2009. **54**(1-3): p. 23-32.
19. Mller, C., *Spherical harmonics*. Vol. 17. 2006: Springer.
20. Munkres, J.R., *Elements of algebraic topology*. 2018: CRC Press.
21. Deng, A.-W. and C.-Y. Gwo. *A Stable Algorithm computing high-order 3D Zernike Moments and Shape Reconstructions*. in *Proceedings of the 2020 4th International Conference on Digital Signal Processing*. 2020.
22. Kintner, E.C., *On the mathematical properties of the Zernike polynomials*. Optica Acta: International Journal of Optics, 1976. **8**: p. 679-680.

23. tom Dieck, T., *Transformation groups*. de Gruyter Studies in Mathematics, 1987. **8**.
24. Hosny, K.M. and M.A. Hafez, *An algorithm for fast computation of 3D Zernike moments for volumetric images*. Mathematical Problems in Engineering, 2012. **2012**.
25. Zernike, F., *Beugungstheorie des Schneidenvorgangs und seiner verbesserten Form, der Phasenkontrastmethode*. Physica, 1934. **1**(8): p. 689–704.
26. Teague, M., *Image analysis via the general theory of moments*. J. Opt. Soc. Am, 1980. **70**: p. 920-930.
27. An-Wen Deng and C.-Y. Gwo, *Fast and stable algorithms for high-order Pseudo Zernike moments and image reconstruction*. Applied Mathematics and Computation, 2018. **334**: p. 239-253.
28. An-Wen Deng and C.-Y. Gwo. *Parallel Algorithms for Computing Zernike Moments*. in *International Congress on Engineering and Information (ICEAI 2016)*. 2016. Osaka, Japan.
29. An-Wen Deng and C.-Y. Gwo, *Parallel Computing Zernike Moments via Combined Algorithms*. The SIJ Transactions on Computer Science Engineering & its Applications (CSEA), The Standard International Journals, 2016. **4**(2): p. 11-19.
30. Deng, A.-W. and C.-Y. Gwo, *Efficient computations for generalized Zernike moments and image recovery*. Applied Mathematics and Computation, 2018. **339**: p. 308-322.
31. Szegő, G., *Orthogonal polynomials*. Vol. 23. 1939: American Mathematical Soc.
32. Wigner, E.P., et al., *Gruppentheorie und ihre Anwendung auf die Quantenmechanik der Atomspektren*. 1931: Springer.
33. DiDonato, A., *Recurrence relations for the indefinite integrals of the associated Legendre functions*. mathematics of computation, 1982: p. 547-551.
34. Mathar, R., *ZERNIKE BASIS TO CARTESIAN TRANSFORMATIONS*. Serbian Astronomical Journal, 2009(179).
35. Graham, R.L., et al., *Concrete mathematics: a foundation for computer science*. Computers in Physics, 1989. **3**(5): p. 106-107.