Unitary rotations in two-, three-, and *D*-dimensional Cartesian data arrays

Guillermo Krötzsch,[1] Kenan Uriostegui,[2] and Kurt Bernardo Wolf[1,*]

[1]*Instituto de Ciencias Físicas Universidad Nacional Autónoma de México, Av. Universidad s/n, Cuernavaca, Morelos 62251, Mexico*

[2]*Facultad de Ciencias Universidad Autónoma del Estado de Morelos, Cuernavaca, Morelos 62251, Mexico*

**Corresponding author: bwolf@fis.unam.mx*

Received March 11, 2014; revised May 13, 2014; accepted May 14, 2014;
posted May 15, 2014 (Doc. ID 207892); published June 19, 2014

Using a previous technique to rotate two-dimensional images on an $N \times N$ square pixellated screen unitarily, we can rotate three-dimensional pixellated cubes of side N, and also generally D-dimensional Cartesian data arrays, also unitarily. Although the number of operations inevitably grows as N^{2D} (because each rotated pixel depends on all others), and Gibbs-like oscillations are inevitable, the result is a strictly unitary and real transformation (thus orthogonal) that is invertible (thus no loss of information) and could be used as a standard.

OCIS codes: (070.2025) Discrete optical signal processing; (100.6890) Three-dimensional image processing; (350.6980) Transforms.

http://dx.doi.org/10.1364/JOSAA.31.001531

1. INTRODUCTION

The unitary rotation of data arrays of pixellated images in two or three dimensions has basic implications in image processing and presents an interesting theoretical problem on the conservation of information. While efficient algorithms exist to perform rotations of such data arrays through interpolation techniques [1], our aim is to represent these transformations unitarily, so that they can be inverted with no loss of information. This requirement cannot be met by interpolation or approximation algorithms.

Among the papers that provide a group-theoretical treatment to the problem, Pei and Liu wrote a very interesting paper [2] that used the transformation between 3D Cartesian and angular oscillator modes given by the Moshinsky coefficients $\langle n_x, n_y, n_z | n, \ell, m\rangle$ [3,4], implementing various cutoffs in n (total energy), ℓ (angular momentum) and/or m (z projection) to approximate 3D functions by finite data arrays.

Our approach was originally based on the geometry and dynamics of the discrete and finite harmonic oscillator model on N points [5]. Without invoking explicitly this physical model, we will refer the construction of Cartesian arrays of $N = 2j + 1$ values (pixels) per side to the well-known representations of the Lie algebra $\mathsf{su}(2)$ of spin j. We regard one-dimensional (1D) data arrays as *states* $f(m) \equiv {}_1\langle j, m|\mathbf{f}\rangle \in \mathcal{C}^N$ on the usual complex vector space of functions on $N = 2j + 1$ equidistant points $m|_{-j}^{j}$; these points belong to the spectrum of the "position" operator J_1, chosen among the three $\mathsf{su}(2)$ generators. Another generator of $\mathsf{su}(2)$, J_3, numbers a convenient orthonormal basis for $\mathcal{C}^N$, that of *Kravchuk* functions. The necessary expressions are briefly reviewed in Section 2.

In two dimensions (2D), we see and call the states pixellated *images* $f(m_x, m_y) \equiv {}_{1x}\langle j, m_x|_{1y}\langle j, m_y|\mathbf{f}\rangle$ on an $N \times N$ grid of points m_x, $m_y|_{-j}^{j}$, whose coordinates are the eigenvalues of two commuting operators $J_{1,x}$, $J_{1,y} \in \mathsf{su}(2)_x \oplus \mathsf{su}(2)_y = \mathsf{so}(4)$—the 4D orthogonal algebra, in the "square" representation $j_x = j_y =: j$, where again $N = 2j + 1$. As we recall in Section 3, we can *import* [6] bases of states with "angular momentum" $\ell|_0^j$, with components classified by $\mu|_{-\ell}^{\ell}$ [7], on which rotations act through multiplication by phases. This rotation is unitary in the complex vector space $|\mathbf{f}\rangle \in \mathcal{C}^{N^2}$ of all 2D images [8]. In Section 4, we use these 2D results for three-dimensional (3D) rotations through Euler angles (α, β, γ) around the z, y, and z axes, respectively. Each factor is a unitary 2D rotation in a stack of pixel planes.

In Section 5, we use the generalized Euler angle parametrization for D-dimensional rotations of Cartesian data arrays, factoring the $\mathsf{SO}(D)$ manifold into $\mathsf{SO}(D-1)$ and a D-sphere S^D. In the concluding Section 6, we briefly indicate approximations which are invariant under rotations and some evident remarks on the extension to the Fourier group of linear Hamiltonian systems [9,10].

2. ONE-DIMENSIONAL FINITE OSCILLATOR BASIS

The eigenvalues and eigenvectors of the generators $\{J_i\}_{i=1}^3$ of $\mathsf{su}(2)$ in the representation j will provide the labels and bases for $\mathcal{C}^N$ $(N = 2j + 1)$ as follows:

$$\begin{aligned} &\text{position } m\text{: } J_1|j, m\rangle_1 = m|j, m\rangle_1, \qquad m|_{-j}^{j}, \\ &\text{mode } n\text{: } J_3|j, n\rangle_3 = (n - j)|j, n\rangle_3, \qquad n|_0^{2j}, \end{aligned} \tag{1}$$

while "momentum" is identified with $-J_2$, and the "energy" eigenvalue of $|j, n\rangle_3$ is $n + (1/2)$ in the finite oscillator model [5]. They have the usual commutation relations [11]

$$[J_3, J_1] = \mathrm{i}J_2, \qquad [J_3, -J_2] = \mathrm{i}J_1, \qquad [J_1, J_2] = \mathrm{i}J_3, \tag{2}$$

where the first two correspond to the geometric and dynamical Hamilton equations of an oscillator, and the third is the

"deformed" commutator between position and mode bases yields the finite oscillator eigenstates,

$$\Psi_n(m) := {}_1\langle j,m|j,n\rangle_3 = {}_3\langle j,j+m|e^{+i\frac{1}{2}\pi J_2}|j,n\rangle_3 = d^j_{n-j,m}\left(\frac{1}{2}\pi\right) \quad (3)$$

$$= (-1)^n\Psi_n(-m) = (-1)^m\Psi_{2j-n}(m) \quad (4)$$

$$= \frac{(-1)^n}{2^j}\sqrt{\binom{2j}{n}\binom{2j}{j+m}}K_n\left(j+m;\frac{1}{2};2j\right), \quad (5)$$

where $d^j_{m,m'}(\beta)$ in Eq. (3) are the well-known Wigner "little d" functions [11], which provide the linear combination coefficients of the mode eigenstates under rotations generated by J_2, namely,

$$\exp(-i\beta J_2)|j,n\rangle_3 = \sum_{n'=0}^{2j} d^j_{n-j,n'-j}(\beta)|j,n'\rangle_3, \quad (6)$$

$$d^j_{\mu,\mu'}(\beta) = {}_3\langle j,j+\mu|e^{-i\beta J_2}|j,j+\mu'\rangle_3 = d^j_{\mu',\mu}(-\beta)$$
$$= \sqrt{(j+\mu)!(j-\mu)!(j+\mu')!(j-\mu')!}$$
$$\times\sum_k \frac{(\cos\tfrac{1}{2}\beta)^{2j-2k+\mu-\mu'}(\sin\tfrac{1}{2}\beta)^{2k-\mu+\mu'}}{k!(j+\mu-k)!(j-\mu'-k)!(\mu'-\mu+k)!}, \quad (7)$$

which are real (by convention), and their generic form will be needed below. In Eq. (5), $K_n(x;(1/2),2j) = K_x(n;(1/2),2j) = {}_2F_1(-n,-x;-2j;2)$ are the *Kravchuk polynomials* of degree n in x, so the $\Psi_n(m)$ have been called the Kravchuk *functions*—they are real, orthogonal, and complete under the natural $\mathcal{C}^N$ inner product. A state $\mathbf{f}\equiv|\mathbf{f}\rangle$ is given by the set of values $f_m = {}_1\langle j,m|\mathbf{f}\rangle \in \mathcal{C}^N$.

3. TWO-DIMENSIONAL ROTATIONS

In two dimensions, we have pixellated images on $N\times N$ square "screens" given as states $\mathbf{f}\equiv|\mathbf{f}\rangle$ of the elements $f_{m_x,m_y} = {}_1\langle j;m_x,m_y|\mathbf{f}\rangle \in \mathcal{C}^{N^2}$, with the direct products $|j;m_x,m_y\rangle_1 \equiv |j,m_x\rangle_{1x}|j,m_y\rangle_{1y}$ of two 1D eigenbases of the two commuting generators $J_{1,x},J_{1,y}\in \mathsf{su}(2)_x\oplus\mathsf{su}(2)_y$ and "position" eigenvalues m_x, $m_y|^j_{-j}$. The 2D Cartesian basis of Kravchuk functions is thus the product of two 1D Kravchuk functions from Eq. (3),

$$\Psi_{n_x,n_y}(m_x,m_y) := {}_1\langle j;m_x,m_y|j;n_x,n_y\rangle_3 = \Psi_{n_x}(m_x)\Psi_{n_y}(m_y), \quad (8)$$

that have the total mode $n = n_x + n_y$ ("energy" n_x+n_y+1), and a mode *difference* $\mu := n_x - n_y$.

On each $n =$ constant $\le 2j$ mode level lie the $n+1$ states

$$\{|j;0,n\rangle_3, |j;1,n-1\rangle_3,\ldots,|j;n,0\rangle_3\}, \qquad n|^{2j}_0, \quad (9)$$

while for $n\ge 2j$ lie the $4j-n+1$ states

$$\{|j;n-2j,2j\rangle_3, |j;n-2j+1,2j-1\rangle_3,\ldots,|j;2j,n-2j\rangle_3\}, \qquad n|^{4j}_{2j}, \quad (10)$$

and the two sets overlap in the $n=2j$ level. We underline that the subsets of $n=$ constant states *do not* form submultiplets under $\mathsf{su}(2)_x\oplus\mathsf{su}(2)_y$, because this algebra does not contain linear raising and lowering operators to connect (n_x,n_y) with $(n_x\pm1, n_y\mp1)$; these will be *imported*.

The importation process [6] subjects the states in each $n=$ constant level to the same linear combination with which the *continuous* oscillator model relates Cartesian and radial-angular mode separations—between the 2D Hermite–Gauss and Laguerre–Gauss modes. In this way the row in Eq. (9), in the lower half of the $\mathsf{so}(4)$ multiplet, having $n+1$ members, is endowed with an "angular momentum" $\ell = (1/2)n = (1/2)(n_x+n_y)$ and projection $\mu = (1/2)(n_x-n_y)$, $|\mu|\le\ell$. We write these states as $|j;n,\mu)$, where the round ket indicates that these *are not* the eigenstates of any $\mathsf{so}(4)$ generator. In the upper half-rhombus [Eq. (10)], we replace n with $4j-n$. The linear combination coefficients are again the well-known Wigner little-d matrices $d^{n/2}_{\mu,\mu'}(\beta)$ of $\mathsf{su}(2)$ [11]. This imported $\mathsf{su}(2)$ will have integer and half-integer representations $\ell = (1/2)n$.

In analogy to the continuous Hermite–Gauss and Laguerre–Gauss functions, out of the Kravchuk functions we form the *Laguerre–Kravchuk* functions given by [7,8,10]

$$\Lambda^j_{n,\mu}(m_x,m_y) := {}_1\langle j;m_x,m_y|j;n,\mu)$$
$$= \sum_{n_x+n_y=n} e^{i(n_x-n_y)/4}d^{n/2}_{\mu/2,(n_x-n_y)/2}\left(\frac{1}{2}\pi\right)$$
$$\times\Psi_{n_x,n_y}(m_x,m_y), \quad (11)$$

which take complex values, and where the ranges of μ are

$$\mu|^{\nu}_{-\nu} \qquad \text{for } \nu = \begin{cases} n, & \text{when } 0\le n\le 2j,\\ 4j-n, & \text{when } 2j\le n\le 4j.\end{cases} \quad (12)$$

On the left side of Fig. 1, we show the images of the Cartesian Kravchuk states [Eq. (8)] on the (m_x,m_y) screens, while on the right we show the images of the Laguerre–Kravchuk states [Eq. (11)]. From Eqs. (4) and (11) we obtain their reality and parity properties

$$\Lambda^j_{n,\mu}(m_x,m_y) = \Lambda^j_{n,-\mu}(m_x,m_y)^*$$
$$= (-1)^n\Lambda^j_{n,\mu}(-m_x,-m_y)$$
$$= (-1)^{m_x+m_y}\Lambda^j_{4j-n,\mu}(m_x,m_y), \quad (13)$$

so that the upper triangle of states [Eq. (10)] mirrors the lower triangle [Eq. (9)] multiplied by a checkerboard of signs: their high-frequency counterparts.

Both sets of discrete functions in Fig. 1 are orthonormal and complete bases for $\mathcal{C}^{N^2}$, and ${}_1\langle j;m_x,m_y|j;n,\mu)$ is the unitary transform kernel between the pixel basis (m_x,m_y) and the mode and angular momentum basis (n,μ). Finally, in the $j\to\infty$ limit, as the upper triangle "vanishes upward," the Kravchuk and Laguerre–Kravchuk functions become the well-known Hermite–Gauss and Laguerre–Gauss beam functions.

Now we can import the action of a rotation $\mathcal{R}(\alpha)$ on the states $|j;n,\mu)$ through their natural multiplication by phases $e^{-i\mu\alpha}$:

$$\mathcal{R}(\alpha)\colon \Lambda^j_{n,\mu}(m_x,m_y) := {}_1\langle j;m_x,m_y|\mathcal{R}(\alpha)|j;n,\mu)$$
$$= e^{-i\mu\alpha}\Lambda^j_{n,\mu}(m_x,m_y). \quad (14)$$

Since $\Lambda^j_{n,\mu}(m_x,m_y)$ are the elements of a unitary matrix of rows and columns (m_x,m_y) and (n,μ), and $\mathcal{R}(\alpha)$ in Eq. (14)

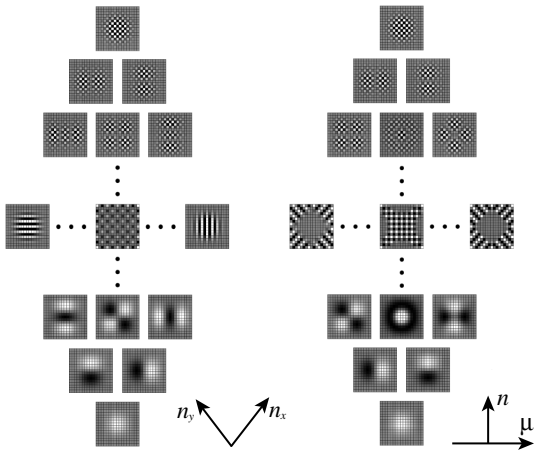

Fig. 1. *Left*: Images of the 2D Cartesian Kravchuk functions $\Psi_{n_x,n_y}(m_x,m_y)$ in Eq. (8), of points m_x, $m_y|_{-j}^{j}$ for $j=8$ on 17×17 pixellated screens, arranged by modes (n_x,n_y). *Right*: The Laguerre–Kravchuk functions $\Lambda^j_{n,\mu}(m_x,m_y)$ in Eq. (11) on the same pixellated screens. At its center ($\mu=0$) the functions are real; to the right ($\mu>0$) are the real parts of $\Lambda^j_{n,\mu}$, and to the left ($\mu<0$) are the imaginary parts of $\Lambda^j_{n,-\mu}$, which can be seen to be off by a phase of $(1/4)\pi$.

is diagonal and unitary in the latter, the images in the pixel basis of the screen, $\{f_{m_x,m_y}\}$, m_x, $m_y|_{-j}^{j}$, will transform unitarily within the 2D square screen with an $N^2\times N^2$ matrix kernel $\mathbf{R}^{(j)}(\alpha)$ given by its elements as

$$\mathcal{R}(\alpha)\colon f_{m_x,m_y}=\sum_{m'_x,m'_y=-j}^{j} f_{m'_x,m'_y}R^{(j)}_{m'_x,m'_y;m_x,m_y}(\alpha), \tag{15}$$

$$R^{(j)}_{m'_x,m'_y;m_x,m_y}(\alpha):=\sum_{n,\mu}\Lambda^j_{n,\mu}(m'_x,m'_y)e^{-\mathrm{i}\mu\alpha}\Lambda^j_{n,\mu}(m_x,m_y)^*. \tag{16}$$

This kernel is unitary and also real, because complex conjugation of Eq. (11) exchanges $\mu\leftrightarrow-\mu$, so under rotations real images will remain real. In Fig. 2, we show rotations obtained with Eqs. (15) and (16) for an image with 1-or-0 values; as in ordinary Fourier analysis, the "discontinuities" inevitably lead to Gibbs-like oscillations, but smoothing quickly lets them die away [8].

The composition of rotations, $\mathcal{R}(\alpha_1)\mathcal{R}(\alpha_2)=\mathcal{R}(\alpha_1+\alpha_2)$, proceeds through the corresponding composition of their representing $N^2\times N^2$ matrices,

$$\sum_{m'_x,m'_y}R^{(j)}_{m_x,m_y;m'_x,m'_y}(\alpha_1)R^{(j)}_{m'_x,m'_y;m''_x,m''_y}(\alpha_2)=R^{(j)}_{m_x,m_y;m''_x,m''_y}(\alpha_1+\alpha_2). \tag{17}$$

The null rotation is $R^{(j)}_{m_x,m_y;m'_x,m'_y}(0)=\delta_{m_x,m'_x}\times\delta_{m_y,m'_y}$, the inverse of a rotation is given by the matrix transpose $R^{(j)}_{m_x,m_y;m'_x,m'_y}(-\alpha)=R^{(j)}_{m'_x,m'_y;m_x,m_y}(\alpha)$, and associativity holds. The matrices in Eq. (16) are thus a (reducible) unitary representation of the group SO(2) of 2D rotations.

4. THREE-DIMENSIONAL ROTATIONS

Three-dimensional rotations $\mathcal{R}_3(\alpha,\beta,\gamma)$ form a group SO(3) that will act on $N\times N\times N$ pixellated cubes parametrized by Euler angles as

$$\mathcal{R}_3(\alpha,\beta,\gamma)=\mathcal{R}_{x,y}(\alpha)\mathcal{R}_{z,x}(\beta)\mathcal{R}_{x,y}(\gamma), \tag{18}$$

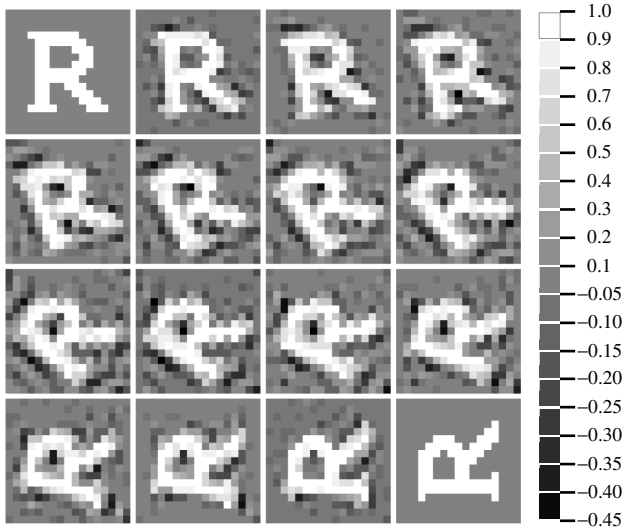

Fig. 2. Rotation of a pixellated letter "*R*" in steps of 6°, from 0° to 90°, for $j=8$; i.e., $N=17$. The $N^2\times N^2$ matrix kernel $R^{(j)}_{m_x,m_y;m'_x,m'_y}(\alpha)$ was computed once for $\alpha=6°$ and applied successively to each image to obtain the next one. At right is the gray-tone range for the pixels in the figure.

where $\mathcal{R}_{i,i'}(\alpha)$ are 2D rotations in the (i,i') plane (i.e., around the z, y, and z axes, respectively). With an evident notation, it is natural to use this decomposition to rotate three-dimensional images $\mathbf{f}\equiv|\mathbf{f}\rangle$ of N^3 components $f_{m_x,m_y,m_z}={}_1\langle j;m_x,m_y,m_z|\mathbf{f}\rangle\in\mathcal{C}^{N^3}$ by applying Eq. (18) successively to stacks of pixel planes as

$$\mathcal{R}_3(\alpha,\beta,\gamma)\colon f_{m_x,m_y,m_y}=\sum_{m'_x,m'_y,m'_z=-j}^{j} f_{m'_x,m'_y,m'_z}\times R^{(j)}_{m'_x,m'_y,m'_z;m_x,m_y,m_z}(\alpha,\beta,\gamma), \tag{19}$$

$$R^{(j)}_{m'_x,m'_y,m'_z;m_x,m_y,m_z}(\alpha,\beta,\gamma):=\sum_{m''_x,m'''_x,m''_y=-j}^{j} R^{(j)}_{m'_x,m'_y;m''_x,m''_y}(\alpha)\times R^{(j)}_{m'_z,m''_x;m_z,m'''_x}(\beta)R^{(j)}_{m'''_x,m''_y;m_x,m_y}(\gamma), \tag{20}$$

where the multi-indices (m_x,m_y,m_z) can be replaced with a single index by $m=m_x+Nm_y+N^2m_z$ if need be. The matrices $R^{(j)}_{m';m}(\alpha,\beta,\gamma)$ form the kernels of real and unitary (and hence orthogonal) transformations of the N^3-dimensional vector space of $\mathcal{C}^{N^3}$ pixellated images in the cube.

Another useful parametrization of SO(3) is the *polar*, which specifies the rotation angle ψ around a unit axis $\hat{n}=(n_x,n_y,n_z)$. The relation to Euler angles is obtained from their 2×2 representation as

$$\cos^2\frac{1}{2}\beta=\cos^2\frac{1}{2}\psi+n_z^2\sin^2\frac{1}{2}\psi, \tag{21}$$

$$\cos\frac{1}{2}(\gamma+\alpha)=\cos\frac{1}{2}\psi/\cos\frac{1}{2}\beta, \tag{22}$$

$$\tan\frac{1}{2}(\gamma-\alpha)=n_y/n_x. \tag{23}$$

In Fig. 3, we show successive rotations of a 3D image by angles ψ about the axis that joins two opposing vertices of the cube; at $2\pi/3=120^\circ$, the original 3D image is

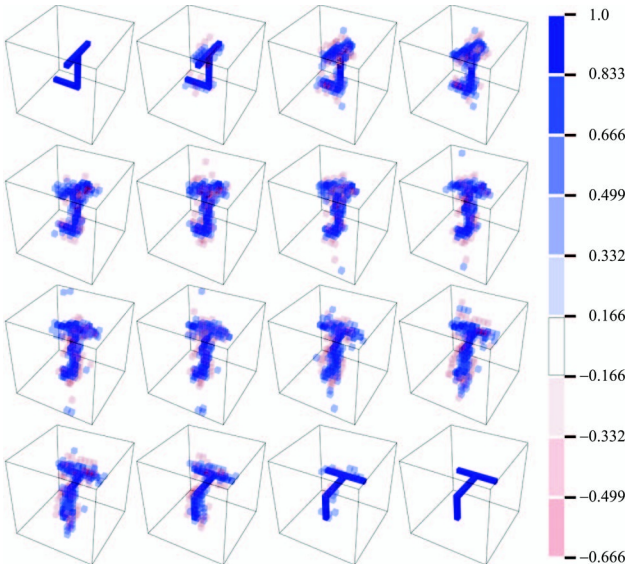

Fig. 3. Rotation of a pixellated 1-and-0 three-dimensional image in steps of 8° from 0° to 120°, around the axis $\hat{n} = (1,1,1)/\sqrt{3}$, for $j = 8$; i.e., $N = 17$. The color bar at right indicates the density range of the partially transparent pixels in the figure. The "TL" figure is *not* centered in the cube.

reconstituted without Gibbs-like oscillations. From Eqs. (21)–(23), this corresponds to the Euler angles $(0,(1/2)\pi,(1/2)\pi)$, where the rotation kernel [Eq. (20)] reduces to the cyclic permutation $m_x \mapsto m_y \mapsto m_z$ of the components of $f_{m_x,m_y,m_z} \in \mathcal{C}^{N^3}$. One common problem with 3D displays is the visual representation of images; we render them using transparent small cubes without smoothing.

Since Eq. (20) is the product of three real unitary transformations in separate planes, these 3D transformations are real and unitary (i.e., orthogonal) maps of $\mathcal{C}^{N^3}$, and are hence reversible with no loss of information. Perhaps surprisingly, they do *not* form a *representation* of SO(3), as Eq. (16) does for plane SO(2) rotations—that is, the product of two 3D transformations [Eq. (19)] will generally *not* follow the product of the corresponding 2×2 spin realization of this group, because the imported 2D bases $\Lambda^j_{n,\mu}(m_i,m_{i'})$ in Eq. (11) do not commute with rotations outside their planes. Those that *do* form SO(2) subgroups are rotations around the z axis $\mathcal{R}_3(\alpha,0,0)$ parametrized by $\alpha \in \mathsf{S}^1$ (the circle), and rotations with any axis $\hat{n}$ that lies in the (x,y) plane, $\mathcal{R}_3(\gamma,\beta,-\gamma)$, where γ is fixed, so $\hat{n} = (-\sin\gamma, \cos\gamma, 0)$ and $\beta \in \mathsf{S}^1$. In these cases, the matrix elements of $R^{(j)}_{m'_x,m'_y,m'_z;m_x,m_y,m_z}(\alpha,\beta,\gamma)$ may be computed only once and applied successively to build all its multiples. Otherwise, as in Fig. 3, each step needs the computation of Eqs. (21)–(23) to produce the corresponding transform kernel.

5. *D*-DIMENSIONAL ROTATIONS

The Euler angle parametrization can be extended to D dimensions given by $(1/2)D(D-1)$ rotations in successive planes. One minor point concerns the z–y–z sequence of 3D rotations, which is bettered by considering z–x–z rotations in the planes (1,2)–(2,3)–(1,2), because this can be more easily generalized [12]. As mentioned above, the Wigner little-d functions $d^j_{m,m'}(\beta)$ that represent rotations around the y axis are real by convention, while those around the x axis will be multiplied by phases $\exp(\mathrm{i}(1/2)\pi(m-m'))$; equivalently, we can use the *little-d*'s for rotations in the (2,3) plane and place a phase factor $\exp(-\mathrm{i}(1/2)\pi(m-m'))$ for rotations in the (3,1) plane (which will not be needed).

Following this z–x–z convention, and generally indicating by $\theta^{(d)}_{i,i'}$ a rotation angle in the (i,i') plane as before, one can write the factorization of the SO(D) group manifold as

$$\mathsf{SO}(D) = \mathsf{SO}(D-1) \otimes \mathsf{S}^{D-1}, \tag{24}$$

where S^d is the manifold of the d-sphere, whose d-angle coordinates and ranges are

$$\begin{aligned} \{\theta^{(d)}_{\cdot,\cdot}\} &:= \{\theta^{(d)}_{1,2}, \theta^{(d)}_{2,3}, \ldots, \theta^{(d)}_{d,d+1}\}, \\ 0 &\le \theta^{(d)}_{1,2} < 2\pi, \\ 0 &\le \theta^{(d)}_{i,i+1} \le \pi, 2 \le i \le d. \end{aligned} \tag{25}$$

Reserving the upper index $1 \le d \le D-1$ to distinguish between distinct rotations in the same plane, one defines SO(D) Euler angles enclosing collective variables in braces, as

$$\{\theta_{\cdot,\cdot}\}^{(d)} := \{\theta^{(1)}_{1,2}, \{\theta^{(2)}_{\cdot,\cdot}\}, \ldots, \{\theta^{(d)}_{\cdot,\cdot}\}\}. \tag{26}$$

Generic rotations $\mathcal{R}_D(\{\theta_{\cdot,\cdot}\}^{(D-1)}) \in \mathsf{SO}(D)$; are thus decomposed recursively into 2D rotations $\mathcal{R}(\theta_{\cdot,\cdot})$ as

$$\mathcal{R}_{d+1}(\{\theta_{\cdot,\cdot}\}^{(d)}) = \mathcal{R}_d(\{\theta_{\cdot,\cdot}\}^{(d-1)})\mathcal{S}_d(\{\theta^{(d)}_{\cdot,\cdot}\}), \tag{27}$$

$$\mathcal{S}_d(\{\theta^{(d)}_{\cdot,\cdot}\}) := \mathcal{R}(\theta^{(d)}_{d,d+1})\mathcal{R}(\theta^{(d)}_{d-1,d})\ldots\mathcal{R}(\theta^{(d)}_{2,3})\mathcal{R}(\theta^{(d)}_{1,2}). \tag{28}$$

Cartesian-pixellated D-dimensional images, given by the set of values $f_{m_1,m_2,\ldots,m_D} \equiv f_{\{m\}} \in \mathcal{C}^{N^D}$, with all $m_i|^j_{-j}$ in a hypercube of edge $N = 2j+1$, will transform under SO(D) rotations with each factor $\mathcal{R}(\theta^{(d)}_{i,i+1})$ rotating in the $(i,i+1)$ plane, and affecting only the indices m_i and m_{i+1} by $\theta^{(d)}_{i,i+1}$ as in Eqs. (15) and (16). As before, these transformations are real and unitary, and hence orthogonal.

6. APPROXIMATION, EXTENSIONS, AND CONCLUSION

We have extended previous work on plane rotations of Cartesian-pixellated $N\times N$ images to generic dimension D—in particular, to three dimensions. The main advantage of these rotations is that they are unitary and real—i.e., orthogonal—in the complex vector space $\mathcal{C}^{N^D}$, so information is conserved. The main drawback of unitarity is the heavy computational cost, because every pixel of the transformed image generally depends on the values of all the original ones, leading to $\propto N^{2D}$ growth in the number of product and sum operations. No fast algorithm seems to exist. Also, beyond $D = 2$ these transformations do not *represent* the group faithfully, since they do not follow its product law exactly.

As we saw in Section 3, the rows of states $n|^{4j}_0$ in the $D = 2$ case do not mix under the imported rotations. Thus, if we approximate (smoothen or filter) an image $\{f_{m_x,m_y}\}$ by its total mode components $n = n_x + n_y$ (such as eliminating high-frequency ones), the resulting (*approximate*) images will transform equally under $\mathcal{R}_2(\theta)$; i.e., these approximants will be *covariant* under 2D rotations [8].

The strategy to use the plane rotations for the D-dimensional case can be extended rather straightforwardly

to the *Fourier* group $\mathsf{U}_{\mathrm{F}}(D) \supset \mathsf{SO}(D)$ [9], which includes fractional Fourier–Kravchuk transforms and gyrations. The former involve multiplication of the states $\Psi_{n_i}(m_i)$ by phases $e^{in_i\phi}$, and the latter are products with rotations [10]. The D^2 parameters of $\mathsf{U}_{\mathrm{F}}(D)$ can be built as the analogues of the Euler-angle decomposition out of phases and plane rotations. Yet only for $D = 2$ plane-pixellated images will these transformations follow the $\mathsf{U}(2)$ product law.

Finally, the Fourier group is but a subgroup of the most general group $\mathsf{U}(N^D)$ of unitary transformations among the N^D pixel elements of images $f_{m_1,\ldots,m_D} \in \mathcal{C}^{N^D}$. This is the "aberration group" described in Ref. [13] for 1D finite signals. At present, we see no compelling application for this group beyond the two-dimensional case. Yet it would further the understanding of the structure of all transformations that conserve information in finite discrete systems, from the same viewpoint where linear and nonlinear canonical transformations conserve the structure of Hamiltonian geometric optics.

ACKNOWLEDGMENTS

Support for this research has been provided by the *Óptica Matemática* projects by UNAM (papiit in101011) and by the National Council for Science and Technology (sep-conacyt 79899).

REFERENCES AND NOTE

1. See, for example, the Image Processing tutorial in WOLFRAM MATHEMATICA, www.wolfram.com/mathematica.
2. S.-C. Pei and C.-L. Liu, "Discrete spherical harmonic oscillator transforms on the Cartesian grids using transformation coefficients," IEEE Trans. Signal Process. **61**, 1149–1164 (2013).
3. M. Moshinsky, *The Harmonic Oscillator in Modern Physics: From Atoms to Quarks* (Gordon & Breach, 1969).
4. V. D. Efros, "Some properties of the Moshinsky coefficients," Nucl. Phys. A **202**, 180–190 (1973).
5. N. M. Atakishiyev and K. B. Wolf, "Fractional Fourier–Kravchuk transform," J. Opt. Soc. Am. A **14**, 1467–1477 (1997).
6. L. Barker, Ç. Çandan, T. Hakioğlu, A. Kutay, and H. M. Ozaktas, "The discrete harmonic oscillator, Harper's equation, and the discrete fractional Fourier transform," J. Phys. A **33**, 2209–2222 (2000).
7. N. M. Atakishiyev, G. S. Pogosyan, L. E. Vicent, and K. B. Wolf, "Finite two-dimensional oscillator. I: the Cartesian model," J. Phys. A **34**, 9381–9398 (2001).
8. L. E. Vicent and K. B. Wolf, "Analysis of digital images into energy-angular momentum modes," J. Opt. Soc. Am. A **28**, 808–814 (2011).
9. R. Simon and K. B. Wolf, "Fractional Fourier transforms in two dimensions," J. Opt. Soc. Am. A **17**, 2368–2381 (2000).
10. K. B. Wolf and T. Alieva, "Rotation and gyration of finite two-dimensional modes," J. Opt. Soc. Am. A **25**, 365–370 (2008).
11. L. C. Biedenharn and J. D. Louck, *Angular Momentum in Quantum Physics, Theory and Application*, G.-C. Rota, ed., Vol. **8** of Encyclopedia of Mathematics and Its Applications (Addison-Wesley, 1981).
12. K. B. Wolf, "A recursive method for the calculation of the SO_n, $\mathrm{SO}_{n,1}$, and ISO_n representation matrices," J. Math. Phys. **12**, 197–206 (1971).
13. K. B. Wolf, "Linear transformations and aberrations in continuous and in finite systems," J. Phys. A **41**, 304026 (2008).