

3D rotation invariants by complex moments

Tomáš Suk*, Jan Flusser, Jiří Boldyš

Institute of Information Theory and Automation of the CAS, Pod vodárenskou věží 4, 182 08 Prague 8, Czech Republic



ARTICLE INFO

Article history:

Received 25 June 2013

Received in revised form

29 January 2015

Accepted 7 May 2015

Available online 19 May 2015

Keywords:

Complex moment

Spherical harmonic

Group representation theory

3D rotation invariant

ABSTRACT

A generalization of the complex moments from 2D to 3D is described. Group representation theory is used to construct 3D rotation invariants from them. The algorithm for automatic generating of the invariants of higher orders is proposed. An algorithm for automatic generation of higher order invariants is proposed. The linearly dependent invariants are eliminated. The invariants are experimentally tested on 3D graphical models and also on real volumetric data.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Pattern recognition of objects in two-dimensional (2D) images has been an important and interesting part of image analysis for many years. Images are often geometrically distorted; distortion of a flat scene can be modeled as a combination of translation, rotation and scaling (TRS) in the case of a scanning device parallel to the scene and as a projective transformation in the opposite case. An efficient approach to recognition of deformed objects is based on using certain features that do not vary under the transformation; generally called *invariants*. Thus, TRS invariants can be applied to recognition of objects independent of translation, attitude and scaling.

Recently, the imaging devices of three-dimensional (3D) objects (computer tomography (CT), magnetic resonance imaging (MRI), rangefinders, etc.) become more and more affordable 3D invariants as a tool for 3D object recognition thus increasingly become an option.

One of the most popular family of 3D invariants is based on image moments. While moment invariants in 2D have been studied extensively for decades (see [7] for a survey), the theory of 3D moment invariants has not been fully explored. The first attempts to derive 3D rotation moment invariants are quite dated, e.g. [8] derived only three TRS invariants from geometric moments of the second order, Guo [9] used a different approach for their derivation. Galvez and Canton [10] employed the normalization approach for 3D recognition. Cyganski and Orr [11] proposed a tensor method for derivation of rotation invariants from geometric moments. A method of geometric

primitives [12,13] yields the same results, obtained however by a different approach.

3D rotation invariants can be based on various types of measurements. Compared to 2D, however, it is less straightforward to obtain 3D rotation invariance on them. Each square-integrable function on a sphere surface can be expanded to an analogy of the Fourier series, whose basis functions are the so-called *spherical harmonics*. The coefficients of the series can be normalized in a much simpler fashion than other features and thus form a basis for many types of invariants. There are still many ways, how to arrange the computation. Kakarala and Mao [1] used the bispectrum well-known from statistics for feature computation. His approach is analogous to moments. Kazhdan [2] used an analogy of phase correlation based on spherical harmonics for comparison of two objects. In that paper it was used for registration, but it can also be utilized for recognition. In Kazhdan et al. [3], the authors used amplitude coefficients as the features. Fehr [4] used the power spectrum and bispectrum computed from a tensor function describing an object compound from patches. In another paper, Fehr and Burkhardt [5] employed a technique known as local binary patterns (LBP). Skibbe et al. [6] use local spherical histograms of oriented gradients.

We can also use moments based on the spherical harmonics. We call them *3D complex moments*. Lo and Don [14] derived 3D TRS invariants up to the 3rd order from the complex moments. Their approach was also used in [15], where rotation is coupled with image blurring. Generalization of this method to higher moment orders has not been reported as it is rather complicated. Several application papers were published, e.g. [16], where the authors use 3D TRS invariants to test handedness and gender from MRI snaps of brains. Two other papers [2] and [17] discuss registration applications as well.

The aim of this paper is to extend the Lo and Don method and to derive invariants of arbitrary order. These invariants are based

* Corresponding author. Tel.: +420 26605 2231; fax +420 28468 0730.

E-mail addresses: suk@utia.cas.cz (T. Suk), flusser@utia.cas.cz (J. Flusser), boldys@utia.cas.cz (J. Boldyš).

on the composite complex moment forms. We present an algorithm, which can automatically generate explicit forms of these invariants. It can also discard the reducible invariants. (Here reducible means linearly dependent. Unfortunately the expression reducible is used also in the group representation theory, see below. We believe that meanings of this term are clear in the context.)

The invariants are constructed directly from the 3D complex moments without any conversion from geometric moments, which is a significant difference from the Lo and Don method. We also discuss problems induced by symmetry of the test objects.

The paper is organized as follows. The second section defines moments and invariants. The third section derives rotation invariants by means of group representation theory. The fourth section is devoted to rotation invariants from complex moments. The next two sections deal with numerical experiments.

2. Definitions

Any scalar obtained as a projection of the image f onto a polynomial P is called *moment*. In 2D, the projection is defined as

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(x, y) f(x, y) dx dy, \tag{1}$$

The polynomial P is called basis function (also called kernel function or analysis function) of the moment.¹

We define here two sets of features in 2D that motivate us later to introduce 3D features suitable to calculate 3D rotation invariants.

Geometric moments m_{pq} in 2D are defined:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy. \tag{2}$$

3D definition is analogous.

Complex moments c_{pq} are used in 2D due to their advantageous behavior under rotation:

$$c_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x + iy)^p (x - iy)^q f(x, y) dx dy. \tag{3}$$

Invariants are image features, which stay unchanged (or changed in a very simple way) under certain group of transformations (in our context under rotation). If f is the original image and g is the transformed image, then

$$I(g) = \Lambda I(f). \tag{4}$$

where Λ is a scalar. Invariant is called *absolute* if $\Lambda = 1$ or *relative* otherwise.

3. Invariants to rotations by means of group representation theory

In this section we recall the main results of the group representation theory related to both 2D and 3D rotations that we need to construct the rotation invariants. A reader is not assumed to have any knowledge of this field. More details can be found e.g. in [18].

3.1. 2D and 3D rotations

Rotations in both 2D and 3D form groups. Their operators can be composed of new elements, which are also rotations. The 2D rotations are commutative, while their order is relevant for the 3D rotations. One parameter α – angle of rotation around a given point – is

sufficient to determine a 2D rotation operator $R(\alpha)$. A rotation operator $R(\alpha, \beta, \gamma)$ in 3D can be defined by three *Euler angles* α, β, γ , resp., specifying angles of consecutive rotations around the z, y, z axes, resp.

3.2. Equivalence of moments and monomials under rotation

We now choose a particular moment order $s \equiv p + q$. We denote geometric moments (2) of an original image of order s m_{pq} and moments of its rotation m'_{pq} . For general rotation, every particular moment m'_{pq} of order s depends linearly on all the moments m_{pq} of order s . When we apply a general linear transformation to the definition of moments (see Eq. (2) in case of 2D), absolute value of the Jacobian of the transformation appears in the integrand, see also [19,20]. Since the Jacobian of the rotation transformation equals one, this dependence is determined solely by transformational properties of the set of monomials $x^p y^q$ of the same order s .

This conclusion is independent of the number of dimensions. In the following, we thus employ an apparatus of the theory of group representations, which can describe related properties of the rotation groups. Our motivation is then to apply the theory in the context of image moments.

3.3. Group representations

Group elements G_a (in our context individual rotations) can be represented by corresponding operators $T(G_a)$ such that

$$T(G_a) \circ T(G_b) = T(G_a \circ G_b). \tag{5}$$

The set of operators is called group representation. Choosing a vector space L (we are interested in monomials of the same order) and its basis $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_s$ we can also define corresponding matrix representation $T_{ji}(G_a)$ satisfying

$$T(G_a) \mathbf{e}_i = \sum_j T_{ji}(G_a) \mathbf{e}_j. \tag{6}$$

Often we can find such a basis that the corresponding matrix representation will be block-diagonal:

$$\begin{pmatrix} T^{(1)} & 0 & 0 & \dots \\ 0 & T^{(2)} & 0 & \dots \\ 0 & 0 & T^{(3)} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}. \tag{7}$$

The $T^{(i)}(G_a)$'s are representations themselves. The space L can be divided into invariant subspaces L_i :

$$L = L_1 + L_2 + L_3 + \dots$$

$T^{(i)}(G_a)$ acting on an element from L_i gives again a linear combination of elements from L_i . If $T^{(i)}(G_a)$'s cannot be split further, they are called irreducible components of the reducible $T(G_a)$ and we write

$$T(G_a) = T^{(1)}(G_a) \oplus T^{(2)}(G_a) \oplus T^{(3)}(G_a) \oplus \dots \tag{8}$$

In the context of matrix representations the symbol \oplus denotes block sum of two matrices:

$$\mathbf{A} \oplus \mathbf{B} = \begin{pmatrix} \mathbf{A} & 0 \\ 0 & \mathbf{B} \end{pmatrix}.$$

3.4. 2D rotation representations

The aforementioned theory is well elaborated for both 2D and 3D rotations in [18]. In the case of the 2D rotations, the following set of one-dimensional irreducible representations could appear in (8):

$$T^{(m)}(a) = \exp(-ima), \quad m = 0, \pm 1, \pm 2, \dots \tag{9}$$

¹ Sometimes we use the term “moment” also if the basis polynomial is weighted by a non-polynomial weight.

They all represent 2D rotation by an angle a . This can be easily verified substituting into (5).

If the space L consists of functions $\psi(r, \varphi)$ (including also monomials) defined on R^2 , where r, φ are polar coordinates, then (7) can be diagonalized using (9) and basis functions

$$\psi^m = \exp(im\varphi), \quad m = 0, \pm 1, \pm 2, \dots \tag{10}$$

If we rotate ψ^m by rotation $R(a)$ we get

$$\begin{aligned} T(R(a)) \exp(im\varphi) &= \exp(im(\varphi - a)) \\ &= \exp(-ima) \exp(im\varphi). \end{aligned} \tag{11}$$

In (11) we can verify that the function ψ^m is transformed according to the one-dimensional representation $T^{(m)}(a)$ from (9).

Comparing (11) and (4) we can see that the basis functions (10) are relative invariants to 2D rotation with $\Lambda = \exp(-ima)$, $f = \exp(im\varphi)$ and g its rotation by a .

3.5. 2D complex moments

There is an important conclusion motivating transition to 3D. Using again cartesian coordinates we get

$$r^p \exp(ip\varphi) = (x + iy)^p,$$

$$r^q \exp(-iq\varphi) = (x - iy)^q.$$

Complex polynomials on the right side appear in the definition of complex moments (3) [21]. The basis functions (see (1)) thus originate from the basis functions of one-dimensional irreducible representations of the 2D rotation group. Simple transformational properties of (10) are inducing the same simple transformational properties of the complex moments [22].

3.6. 3D rotation representations

We want to proceed in the same way in the case of 3D rotations. Their irreducible representations are usually denoted $D^{(j)}$, $j = 0, 1, 2, \dots$. Dimension of $D^{(j)}$ is $2j+1$. If the space L consists of functions $\psi(r, \vartheta, \varphi)$ defined on R^3 and expressed in spherical coordinates r, ϑ, φ , it can be decomposed into series with basis functions:

$$R(r)Y_m^\ell(\vartheta, \varphi). \tag{12}$$

The radial part $R(r)$ can be e.g. monomials or spherical Bessel functions. The angular part consists of the so-called spherical harmonics:

$$Y_m^\ell(\vartheta, \varphi), \quad m = -\ell, -\ell+1, \dots, \ell-1, \ell. \tag{13}$$

For one particular ℓ they form a basis of the irreducible representation $D^{(\ell)}$. The corresponding matrix representations are known as Wigner D-matrices.

Below we introduce 3D complex moments, whose transformational properties under rotations are determined by those of the spherical harmonics. This is the reason why it is easy to construct rotation invariants from them.

We are mainly interested in the invariant one-dimensional irreducible subspace transforming itself according to the irreducible representation $D^{(0)}$. Single element of the corresponding matrix representation equals one. Basis function Y_0^0 is thus absolute invariant to 3D rotation.

Basis functions of the irreducible representations with $\ell \neq 0$ are not invariants. However, we can also construct products of the basis functions. If we combine basis functions corresponding to the representations $D^{(j_1)}$ and $D^{(j_2)}$, the result is transformed according to their so-called tensor product (in the context of matrix representations also Kronecker product, denoted \otimes). It can be shown [18] that it is

reducible into

$$D^{(j_1)} \otimes D^{(j_2)} = \sum_{j = |j_1 - j_2|}^{j = j_1 + j_2} D^{(j)}. \tag{14}$$

Thus, we can construct e.g. tensor product of $D^{(2)}$ with itself. According to (14) it is reducible and one of its irreducible components is $D^{(0)}$:

$$D^{(2)} \otimes D^{(2)} = D^{(0)} \oplus D^{(1)} \oplus D^{(2)} \oplus D^{(3)} \oplus D^{(4)}.$$

As it was explained above, we systematically create such one-dimensional representations to find invariants.

In the context of matrix representations, the tensor product of two representations $T_{ik}^{(\alpha)}$ and $T_{j\ell}^{(\beta)}$ is

$$T_{ij,k\ell}^{(\alpha \times \beta)} \equiv T_{ik}^{(\alpha)} T_{j\ell}^{(\beta)}.$$

Denoting now generally the basis functions of the irreducible representations from (14) $\varphi_{j_1}^i$ and $\varphi_{j_2}^i$, and the basis functions of one particular new irreducible representation Ψ_j^i (for $D^{(j)}$), we can write

$$\Psi_j^k = \sum_{m = \max(-j_1, k - j_2)}^{\min(j_1, k + j_2)} \langle j_1, j_2, m, k - m | j, k \rangle \varphi_{j_1}^m \varphi_{j_2}^{k-m}. \tag{15}$$

The coefficients $\langle j_1, j_2, m, k - m | j, k \rangle$ are called Clebsch–Gordan coefficients [18]. They are property of the rotation symmetry and we can use (15) in an arbitrary space L . Ψ_0^0 is an absolute invariant.

3.7. 3D complex moments

In Section 3.5 we provide a reasoning for simple rotation properties of the 2D complex moments. Correspondingly, we can define 3D complex moments of order s , latitudinal repetition ℓ and longitudinal repetition m as projections on the corresponding spherical harmonics times q^s (see also [2] among others)

$$c_{s\ell}^m = \int_0^{2\pi} \int_0^\pi \int_0^\infty q^{s+2} Y_\ell^m(\vartheta, \varphi) f(q, \vartheta, \varphi) \sin \vartheta \, dq \, d\vartheta \, d\varphi. \tag{16}$$

$q^2 \sin \vartheta$ is the Jacobian of the transformation of Cartesian to spherical coordinates q, ϑ and φ .

The spherical harmonics can be expressed in the Cartesian coordinates as $Y_\ell^m(x, y, z)$ after the substitution $\rho = \sqrt{x^2 + y^2 + z^2}$, $\sin \vartheta e^{i\varphi} = (x + iy)/\rho$ and $\cos \vartheta = z/\rho$. When the basis functions of the 3D complex moments are polynomials, the moments form complete and independent description of the image function. For it, we need to eliminate both ρ in the denominator and under an odd exponent. It implies a constraint for the order s : $s = \ell, \ell + 2, \dots$. We can reverse the constraint: for $s = 0, 1, 2, \dots$ we obtain $\ell = 0, 2, 4, \dots, s - 2, s$ for even s and $\ell = 1, 3, 5, \dots, s - 2, s$ for odd s . The constraint for the longitudinal repetition m is the same as for the order of the spherical harmonics: $m = -\ell, -\ell + 1, \dots, \ell$. Then the 3D complex moments can be bijectively converted to geometric moments. They create complete and independent description of an object.

There is a relation between the spherical harmonic of negative and positive order $Y_\ell^{-m} = (-1)^m (Y_\ell^m)^*$; from which $c_{s\ell}^{-m} = (-1)^m (c_{s\ell}^m)^*$.

The complex moments defined in this section are not the same as in [14]. They transform geometric 3D moments into their complex moments that are complex moment forms transforming according to irreducible representations of 3D rotations. Our complex moments are already constructed to have these properties.

4. Moment invariants

Derivation of the rotation invariants computed from complex moments is relatively easy in 2D.

4.1. 2D case

From (3) in polar coordinates we see that the complex moment after the rotation by angle α is

$$c'_{pq} = e^{-i(p-q)\alpha} \cdot c_{pq}. \tag{17}$$

The rotation invariants can be constructed as products:

$$\prod_{i=1}^r c_{p_i q_i}^{k_i}, \tag{18}$$

where k_i , p_i , and q_i , ($i = 1, \dots, r$) are non-negative integers such that

$$\sum_{i=1}^r k_i(p_i - q_i) = 0. \tag{19}$$

Then the phase changes (17) of the moments during rotation are eliminated (so-called phase cancelation) and the product is invariant to the rotation.

4.2. 3D case

In accordance with the previous sections, we construct absolute invariants to 3D rotations here. We prove their invariance by the way how they are derived – we are looking for bases of representations $D^{(0)}$ that are absolute invariants, as it is explained in the previous sections.

c_{00}^0 is immediately an invariant. For other invariants we have to construct other forms transforming according to $D^{(0)}$, see also (14) and (15).

To be able to calculate more invariants we have to make clear what is the relation between 3D complex moments of an original image $c_{s\ell}^m$ and of a rotated image $c_{s\ell}^m(\mathbf{R})$. To find it we can apply a rotation to an image in the definition of the 3D complex moments written in the Cartesian coordinates. After a substitution we can see that the way how the 3D complex moments are transformed is determined by the way how the spherical harmonics are transformed. Spherical harmonics under rotations and the 3D complex moments calculated on rotated images are transformed in the same way.

Specifically, we get a 3D analogy of (17)

$$c_{s\ell}^m(\mathbf{R}) = \sum_{m'=-\ell}^{\ell} D_{m'm}^{\ell}(\mathbf{R}) c_{s\ell}^{m'}. \tag{20}$$

$D_{m'm}^{\ell}(\mathbf{R})$ is the Wigner D-matrix already mentioned above as a matrix irreducible representation corresponding to a particular $D^{(\ell)}$. \mathbf{R} is the rotation operator.

We can now proceed in accordance with Section 3.6. Based on (15) as well as in Lo and Don [14] we construct composite complex moment forms, see [14]

$$c_s(\ell, \ell')_j^k = \sum_{m=\max(-\ell, k-\ell')}^{\min(\ell, k+\ell')} \langle \ell, \ell', m, k-m | j, k \rangle c_{s\ell}^m c_{s\ell'}^{k-m}. \tag{21}$$

combining bases of representations $D^{(\ell)}$ and $D^{(\ell')}$ to get basis of representation $D^{(j)}$. k denotes single basis elements. $c_s(\ell, \ell')_0^0$ is then 3D rotation invariant. Substituting the Clebsch–Gordan coefficients $\langle \ell, \ell', m, -m | 0, 0 \rangle = (-1)^{\ell-m} / \sqrt{2\ell+1}$ it is

$$c_s(\ell, \ell')_0^0 = \frac{1}{\sqrt{2\ell+1}} \sum_{m=-\ell}^{\ell} (-1)^{\ell-m} c_{s\ell}^m c_{s\ell'}^{-m}. \tag{22}$$

Further we can combine the composite complex moment forms and the complex moments. Basis of the corresponding tensor

product is then

$$c_s(\ell, \ell')_j c_{s'} = \frac{1}{\sqrt{2j+1}} \sum_{k=-j}^j (-1)^{j-k} c_s(\ell, \ell')^k c_{s'}^{-k}. \tag{23}$$

Analogously, basis of a tensor product corresponding to combining two composite complex moment forms is

$$c_s(\ell, \ell')_j c_{s'}(\ell'', \ell''')_j = \frac{1}{\sqrt{2j+1}} \sum_{k=-j}^j (-1)^{j-k} c_s(\ell, \ell')^k c_{s'}(\ell'', \ell''')_j^{-k}. \tag{24}$$

If both forms in (24) are identical, we obtain

$$c_s^2(\ell, \ell')_j = \frac{1}{\sqrt{2j+1}} \sum_{k=-j}^j (-1)^{j-k} c_s(\ell, \ell')^k c_s(\ell, \ell')_j^{-k}. \tag{25}$$

In all the formulas, the parameter j of the Clebsch–Gordan coefficients must be even, see also indexing in (16). Conceivably, we can multiply more forms, but as we will see later, the product of two factors in (23)–(25) is usually sufficient.

We can denote one particular $D^{(j)}$ on the right-hand side of (14) as

$$(D^{(j_1)} \otimes D^{(j_2)})^{(j)}.$$

Then c_{00}^0 corresponds to $D^{(0)}$, $c_s(\ell, \ell')_0^0$ to $(D^{(\ell)} \otimes D^{(\ell')})^{(0)}$, $c_s(\ell, \ell')_j c_{s'}$ to $((D^{(\ell)} \otimes D^{(\ell')})^{(j)} \otimes D^{(j)})^{(0)}$ and $c_s(\ell, \ell')_j c_{s'}(\ell'', \ell''')_j$ to $((D^{(\ell)} \otimes D^{(\ell')})^{(j)} \otimes (D^{(\ell'')} \otimes D^{(\ell''')})^{(j)})^{(0)}$. We can use the same expressions for various moment orders. E.g. $D^{(0)}$ leads to a sequence of invariants of all even orders: $c_{20}^0, c_{40}^0, \dots$. The moment c_{00}^0 is also a rotation invariant, but it is usually used for scaling normalization.

4.3. Additional invariance

If we express the spherical harmonics in Cartesian coordinates,² we can use central complex moments to get the translation invariance:

$$c_{s\ell}^m = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} Q^s Y_{\ell}^m(x-x_c, y-y_c, z-z_c) f(x, y, z) dx dy dz, \tag{26}$$

where (x_c, y_c, z_c) is the centroid of the object.

The complex moments can be normalized to scaling:

$$\tilde{c}_{s\ell}^m = \frac{c_{s\ell}^m}{(c_{00}^0)^{s/n+1}}, \tag{27}$$

where n is the dimension of the space, i.e. $n=2$ in 2D and $n=3$ in 3D. Of course, the third index m is not used in 2D.

Thus, if the general 3D complex moments (16) are used for computation of the rotation invariants, the rotation invariants only are obtained. If the central moments (26) are used, the combined invariants to translation and rotation are obtained. If the moments (27) normalized to scaling are finally used, the scaling invariance can be further added. This way an arbitrary combination of invariance to translation, rotation and scaling can be achieved.

² It can be done by substitution $\sin \vartheta e^{i\varphi} = (x+iy) / \sqrt{x^2+y^2+z^2}$ and $\cos \vartheta = z / \sqrt{x^2+y^2+z^2}$.

4.4. Automated generation of the invariants

We have programmed the formulas from the last subsection; the code generates TRS invariants using four methods:

1. Single complex moment c_{s0}^0 :

$$s = 2, 4, 6, \dots$$

2. Single composite complex moment form $c_s(\ell, \ell')_0^0$:

$$s = 2, 3, 4, \dots,$$

$$\ell = s, s-2, s-4, \dots, 1.$$

The last value of ℓ in the loop is not always one, but $1 + \text{rem}(s+1, 2)$, rem is remainder after division. Regardless, the loop breaking condition $\ell \geq 1$ is sufficient. It is true for the following loops, too.

3. Product of a form and a complex moment $c_s(\ell, \ell')_j c_{s'}$:

$$s = 2, 3, 4, \dots,$$

$$s' = 2, 4, 6, \dots, s,$$

$$j = 2, 4, 6, \dots, s',$$

$$\ell = s, s-2, s-4, \dots, 1,$$

$$\ell' = \ell, \ell-2, \ell-4, \dots, \max(\ell-j, 1).$$

where ℓ' must satisfy both $\ell' \geq \ell-j$ and $\ell' \geq 1$.

4. Product of two forms $c_s(\ell, \ell')_j c_{s'}(\ell'', \ell''')_j$:

$$s = 2, 3, 4, \dots,$$

$$s' = 2, 3, 4, \dots, s,$$

$$j = 2, 4, 6, \dots, s',$$

$$\ell = s, s-2, s-4, \dots, 1,$$

$$\ell' = \ell, \ell-2, \ell-4, \dots, \max(\ell-j, 1),$$

$$\ell'' = s', s'-2, s'-4, \dots, 1,$$

$$\ell''' = \ell'', \ell''-2, \ell''-4, \dots, \max(\ell''-j, 1).$$

4.5. Elimination of the reducible invariants

The number of independent invariants n_i can be determined as a difference between the number of independent moments n_m and the number of independent degrees of freedom of the transformation n_{dof} :

$$n_i = n_m - n_{dof}. \tag{28}$$

Since we have $n_m = \binom{s+3}{3}$ moments up to the order s and $n_{dof} = 7$ parameters of TRS in 3D (3 translations, scaling and 3 rotations), we can obtain at most $n_i = \binom{s+3}{3} - 7$ independent invariants.

An important part of the process is the elimination of dependent invariants. If a certain set of invariants is dependent, it means that there exists an invariant which is a function of the others. This function may have various forms: only a product, linear combination, or a polynomial (which is in fact linear combination of various products). The simplest dependent invariants are just zero or they are identical with the others. Elimination of the linear dependencies is feasible. The invariants remaining after the elimination of the linearly dependent ones are called irreducible; those eliminated are called reducible invariants.³ Elimination of the polynomial dependencies among irreducible invariants is much more difficult and this problem has not been fully resolved yet. A complete removal of all polynomial dependencies has been

³ We must distinguish between reducible/irreducible invariant and reducible/irreducible group representation.

solved for very small sets of low orders only. The method of finding all irreducible invariants we propose here is a generalization of our previously published method for the 2D affine case [7].

The process of generation guarantees that the invariants cannot be identically zero, but surprisingly some of them can be identical. E.g. if $s=4$, the following invariants are identical:

$$c_4(4, 2)_4 c_4 = c_4(4, 4)_2 c_4,$$

$$c_4(2, 2)_4 c_4 = c_4(4, 2)_2 c_4,$$

$$c_4(4, 4)_4 c_4(4, 2)_4 = \frac{9}{\sqrt{143}} c_4(4, 4)_2 c_4(4, 2)_2,$$

$$c_4(4, 2)_4 c_4(2, 2)_4 = c_4(4, 2)_2 c_4(2, 2)_2,$$

$$c_4^2(2, 2)_4 = \frac{3}{5} \sqrt{5} c_4^2(2, 2)_2.$$

Then we generate all products of the generated invariants, whose number of factors in one term does not exceed the generated maximum, and search for identities. This way the products of the generated invariants are eliminated. The generated products also serve as a basis for elimination of linear combinations. The numbers of the generated invariants are listed in Table 1. The first row of the table shows the orders of the invariants, the second row contains the cumulative number of all generated invariants up to the given order, the third row contains the number of the irreducible invariants and the fourth row contains the theoretical maximum number of the independent invariants calculated using (28).

4.6. The resulting invariants

The main limitation of the increase of the order of the invariants is numerical precision. The coefficients of the invariants can theoretically be expressed as a square root of a rational number. If a coefficient is stored as a floating-point number, then it sometimes cannot be stored precisely and we are not able to convert it and further work with it. So, in the standard arithmetics, the invariants of higher orders than 6 can be used with coefficients in floating point only. If we continue in the generation to higher orders, we encounter other difficulties. One of them being accuracy of the moments themselves, see e.g. [7], the other is the computing complexity and memory demands associated with elimination of the reducible invariants.

The complex moment rotation invariants of the second order are also expressed in 3D complex moments with their normalization to scaling:

$$\Phi_1 = c_2 = (c_{20}^0)/(c_{00}^0)^{5/3}$$

$$\begin{aligned} \Phi_2 &= c_2(2, 2)_0^0 \\ &= \sqrt{\frac{1}{5}}((c_{22}^0)^2 - 2c_{22}^{-1}c_{22}^1 + 2c_{22}^{-2}c_{22}^2)/(c_{00}^0)^{10/3} \end{aligned}$$

$$\begin{aligned} \Phi_3 &= c_2(2, 2)_2 c_2 = \sqrt{\frac{1}{35}}(-\sqrt{2}(c_{22}^0)^3 + 3\sqrt{2}c_{22}^{-1}c_{22}^1 c_{22}^2 \\ &\quad - 3\sqrt{3}(c_{22}^{-1})^2 c_{22}^2 - 3\sqrt{3}c_{22}^{-2}(c_{22}^1)^2 \\ &\quad + 6\sqrt{2}c_{22}^{-2}c_{22}^0 c_{22}^2)/(c_{00}^0)^5 \end{aligned}$$

Table 1
The numbers of the 3D irreducible and independent complex moment rotation invariants.

Order	2	3	4	5	6	7
All	4	16	49	123	280	573
Irreducible	3	13	37	100	228	486
Independent	3	13	28	49	77	113

The complex moment rotation invariants of the third order are presented as the products of the corresponding composite complex moment forms only:

$$\begin{aligned}\Phi_4 &= c_3(3, 3)_0^0, \\ \Phi_5 &= c_3(1, 1)_0^0, \\ \Phi_6 &= c_3(3, 3)_2 c_2, \\ \Phi_7 &= c_3(3, 1)_2 c_2, \\ \Phi_8 &= c_3(1, 1)_2 c_2, \\ \Phi_9 &= c_3(3, 3)_2 c_2(2, 2)_2, \\ \Phi_{10} &= c_3^2(3, 3)_2, \\ \Phi_{11} &= c_3(3, 3)_2 c_3(3, 1)_2, \\ \Phi_{12} &= c_3(3, 3)_2 c_3(1, 1)_2, \\ \Phi_{13} &= c_3(3, 1)_2 c_3(1, 1)_2.\end{aligned}$$

In [14], the authors published 12 invariants (in this order): Φ_1 , Φ_2 and Φ_3 of the second order and Φ_4 , Φ_5 , Φ_{10} , $c_3^2(3, 1)_2$, Φ_{11} , Φ_{13} , Φ_6 , Φ_7 and Φ_8 of the third order. The invariants Φ_{12} , $c_3^2(3, 1)_2$ and $\Phi_4\Phi_5$ are linearly dependent; we decided to use Φ_{12} , because it has fewer terms (18 instead of 19 in $c_3^2(3, 1)_2$). According to (28), there should be another independent invariant of the third order, which is Φ_9 .

We have chosen the following irreducible invariants of the fourth order for recognition: c_4 , $c_4(4, 4)_0^0$, $c_4(2, 2)_0^0$, $c_4(4, 4)_2 c_2$, $c_4(2, 2)_2 c_2$, $c_4(4, 4)_2 c_4$, $c_4(4, 2)_2 c_4$, $c_4(2, 2)_2 c_4$, $c_4(4, 4)_4 c_4$, $c_4(2, 2)_2 c_3(1, 1)_2$, $c_4^2(4, 4)_2$, $c_4(4, 4)_2 c_4(4, 2)_2$, $c_4(4, 4)_2 c_4(2, 2)_2$, $c_4^2(4, 2)_2$, and $c_4(4, 2)_2 c_4(2, 2)_2$.

4.7. Invariants for symmetric objects

In moment-based recognition, the objects having some kind of symmetry (circular, rotational, axial, dihedral, etc.) often cause difficulties because some of their moments are zero. If such a moment is used as a factor in a product, the product always vanishes. Since the invariants are often products or linear combinations of products of the moments, the object symmetry can make certain invariants trivial. This is of course not a problem if there are no two or more classes with the same symmetry; in such a case vanishing invariants may serve as a discriminatory features. However, if more classes have the same symmetry, the vanishing invariants should not be used for recognition as they decrease the recognition power of the system while increasing computing complexity. For 2D rotation invariants this problem was discussed thoroughly in [23].

In 2D, there are only two infinite sequences of symmetry groups (rotational and dihedral) and one infinite generalization (circular). In 3D, there are 7 infinite sequences of symmetry groups, 7 separate symmetry groups of regular bodies (e.g. tetrahedron or cube) and 3 infinite generalizations (conic, cylinder and sphere). Each symmetry brings additional linear dependency into the complex moments and makes certain invariants vanishing. A detailed study can be found in [24].

5. Experiments on benchmark data

We carried out two experiments to show the behavior of our invariants. The aim of the first one was to demonstrate rotation invariance, robustness and recognition power in case of classes that are similar to one another and that are even difficult to distinguish visually. The second experiment was performed on generic objects that are easy to recognize by humans but their big intra-class variability makes the task very difficult for algorithms.

We used the Princeton Shape Benchmark [25] in both experiments. The database contains 1814 3D binary objects given by triangular patches of their surfaces.

5.1. Airplane recognition

In this experiment, we worked with six classes of jet planes. Each class was represented by only one template plane, see Fig. 1. To create “undefined” airplanes, we rotated each template 10 times randomly and added a zero-mean Gaussian noise to the coordinates of the triangle vertices of the airplane surface. This resulted in rotated airplanes with noisy surfaces, see Fig. 2 for two examples.

To measure the impact of the noise we used signal-to-noise ratio (SNR), defined in this case as

$$\text{SNR} = 10 \log(\sigma_{sa}^2 / \sigma_{na}^2), \quad (29)$$

where σ_{sa} is the standard deviation of the object coordinates in the corresponding axis, i.e. σ_{sx} in the x -axis, σ_{sy} in the y -axis and σ_{sz} in the z -axis, analogously σ_{na} is the standard deviation of the noise in the a -axis, $a \in \{x, y, z\}$. σ_{na} 's are set to equalize SNR for every axis in the experiment. We used 10 SNRs from 3.44 dB to 55 dB. On each level, the noise was generated 2 times and the average success rate was calculated.

We used 100 irreducible invariants up to the 5th order from volume moments and the nearest-neighbor rule in their space for airplane classification. The obtained success rates are visualized in Fig. 3. The recognition was error-free for low noise (SNR > 35 dB), which illustrates the rotation invariance and discrimination power. As the noise becomes heavier, the success rate decreases from 100% for SNR > 35 dB to 20% for SNR=6.67 dB. In the latter case, the deviation of the invariants differ so much that their values for different airplanes may overlap. However, in this noisy case even the visual recognition is tricky for a non-trained person.

To compare the results achieved by means of moments to a different method, we repeated this experiment using a spherical harmonic representation (SHR) from [3]. The method first decomposes an object into concentric spherical layers. Every layer is then approximated as a function on a layer and projected on different irreducible subspaces $D^{(l)}$. L_2 norms of these projections then form rotation invariants. Since this method requires volumetric data, we converted the objects to the volumetric representation with resolution 200 voxels in the directions of their maximum sizes. We used 47 layers approximately 2 voxels thick. The angular spherical coordinates ϑ , φ (latitude and longitude or elevation and azimuth) were quantized into 64 bins. Spherical harmonics from the zeroth to the fifth orders were used, i.e. we had six features in each layer and 282 features altogether. The amplitudes of frequency components of all orders form the rotation invariants.

The success rate is shown again in Fig. 3. Compared to the success rate of the complex moments, we can observe several differences. Low noise (SNR > 30 dB) does not harm the recognition by moments at all while SHR is below 90%. The reason is that SHR is not a complete description of an object because it uses only the amplitudes while the phases are omitted. The results would probably be better if finer sampling was used. For bigger noise to 23 dB the success rates of both methods are comparable. For heavy noise below 7 dB the results are practically random.

A significant difference between the methods is in their complexity. The computation time 2.5 min of SHR against 0.5 s of the complex moments for recognition of one airplane is approximately 300 times longer. To calculate SHR, the object must be actually represented as a volume by its voxels while the moments can be calculated just from the triangulated surface. In this experiment the objects typically consisted of several thousands of surface triangles while the number of voxels was several millions. This leads to the huge difference in complexity. On the other hand, the exponential computing complexity with respect to the moment order against the linear computing complexity with respect to SHR order is negligible when both orders equal five. This extreme complexity of SHR also prevented us from

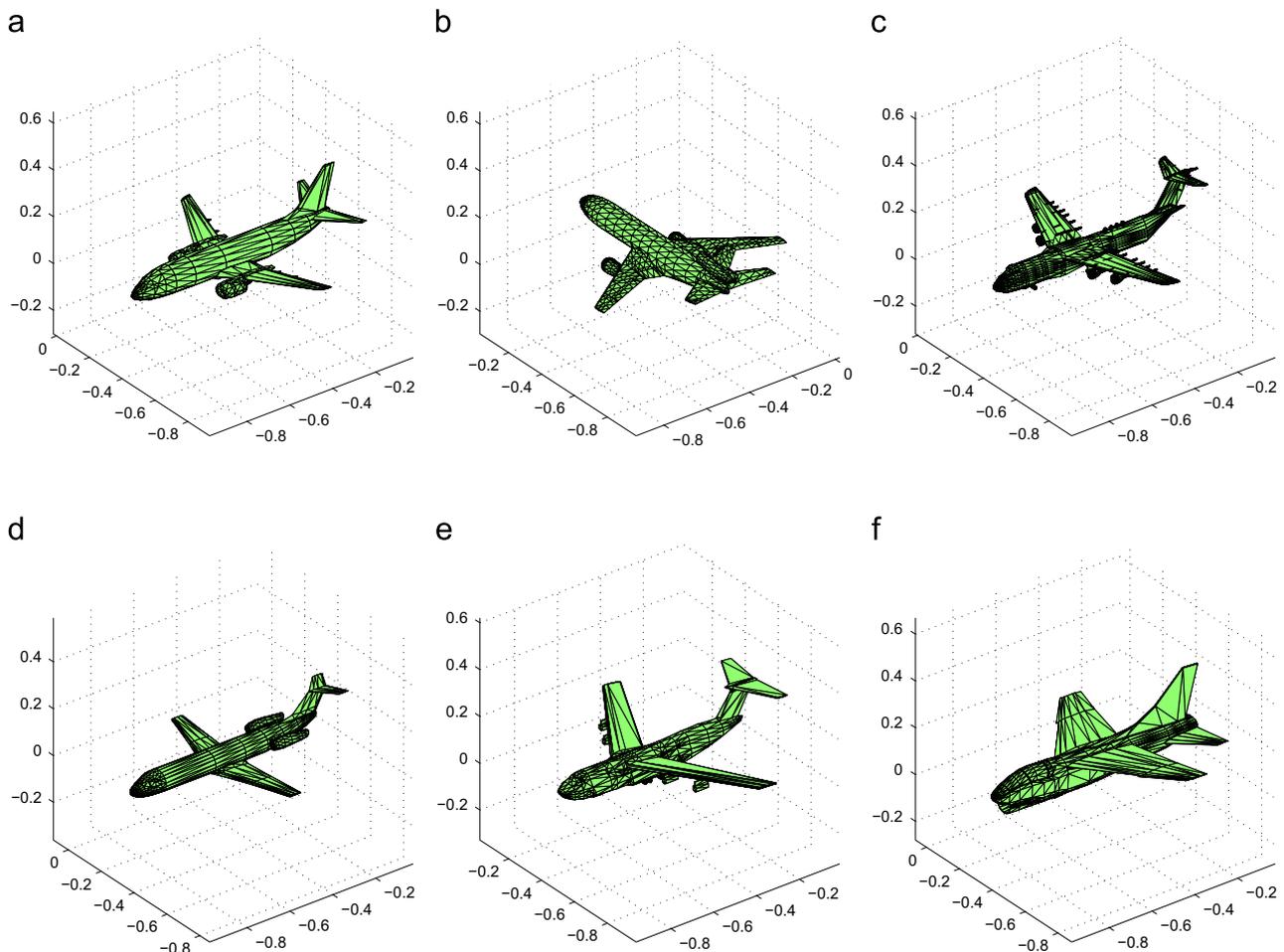


Fig. 1. The airplanes used in the experiment.

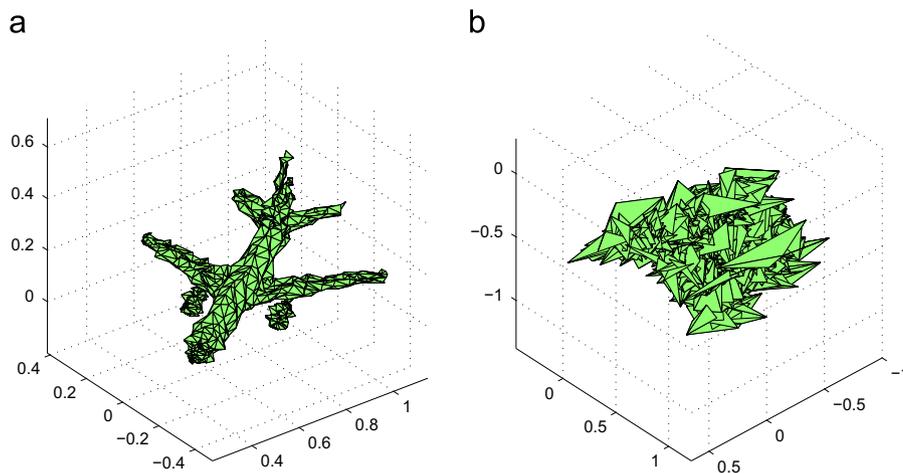


Fig. 2. Example of the rotated and noisy airplane from Fig. 1b with SNR (a) 26 dB and (b) 3.4 dB.

using finer voxel grid. If we did it, we would probably approach 100% success rate for low noise but the experiment would take several weeks of the CPU time.

5.2. Recognition of generic classes

In the second experiment we tested performance of the invariants in a task the invariants are not designed for – recognition of generic objects. Classification to generic classes (such as “car”, “bike”, and “dog”) is easy for humans because they incorporate their experience,

prior knowledge and contextual information. For automatic systems this task is very difficult due to high intra-class variability and usually requires high-level of abstraction, structural description and advanced classification techniques. Here we tried to resolve this task for a limited number of classes by the same invariants as in the previous experiment.

We have chosen six generic classes of the Princeton Shape Benchmark database: sword, two-story home, dining chair, handgun, ship, and sedan. The classes were represented by their training sets consisting of 15 swords, 11 houses, 11 chairs, 10 handguns, 10 ships

and 10 cars (see Fig. 4 for one template per class). The independent test sets contained 16 swords, 10 houses, 11 chairs, 10 handguns, 11 ships and 10 cars. In this experiment we use *surface moments* defined in the next section. For this purpose we use an analogy of the 3D complex moments calculated only on a surface. We do not describe them here in detail, because their properties related to 3D rotation invariance are the same. In the following section we show how they can be calculated.

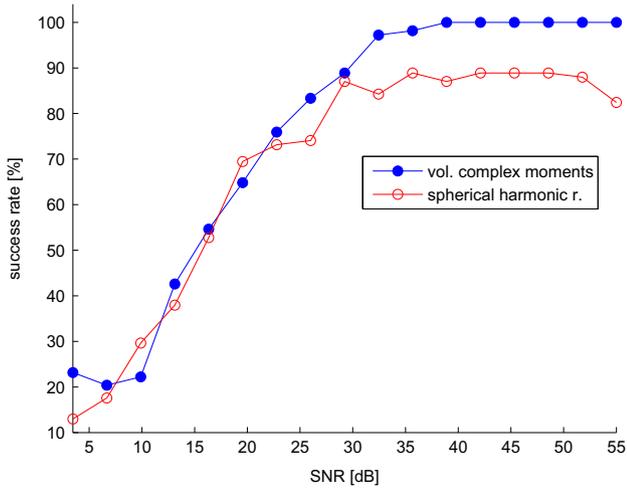


Fig. 3. The success rate of the invariants from the volume complex moments compared to the spherical harmonic representation.

The overall recognition rate was 82.4%, which is higher than we originally expected. It shows that the invariants capture certain global characteristics that are common for all representatives of a generic class. This property does not have any theoretical justification – generic classes are defined by means of other properties than those measurable by moments. For other classes and/or other training sets the success rates could be different. For comparison we repeated the same experiment using the Lo and Don invariants [14]. The success rate dropped to 79.4% because the Lo and Don set is incomplete and does not contain certain invariants which we derived.

5.3. Computation of the moments of triangulated objects

3D images are usually given in a volumetric representation as a “data cube” of voxels. Such representation is typically produced by CT, MRI and many other 3D imaging devices. Then the moments are calculated directly from the definition by volume integration, for instance as

$$m_{pqr} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q z^r f(x, y, z) dx dy dz \tag{30}$$

in case of geometric moments. We call them *volumetric moments*.

However, for binary 3D shapes the volume representation is highly redundant. They can be unambiguously represented by their surface only. In a discrete case, the most common surface representation is via triangulation. Then the whole object is fully determined by the triangle vertices, the number of which is relatively low compared to the number of voxels. This efficient representation is also employed in the Princeton dataset.

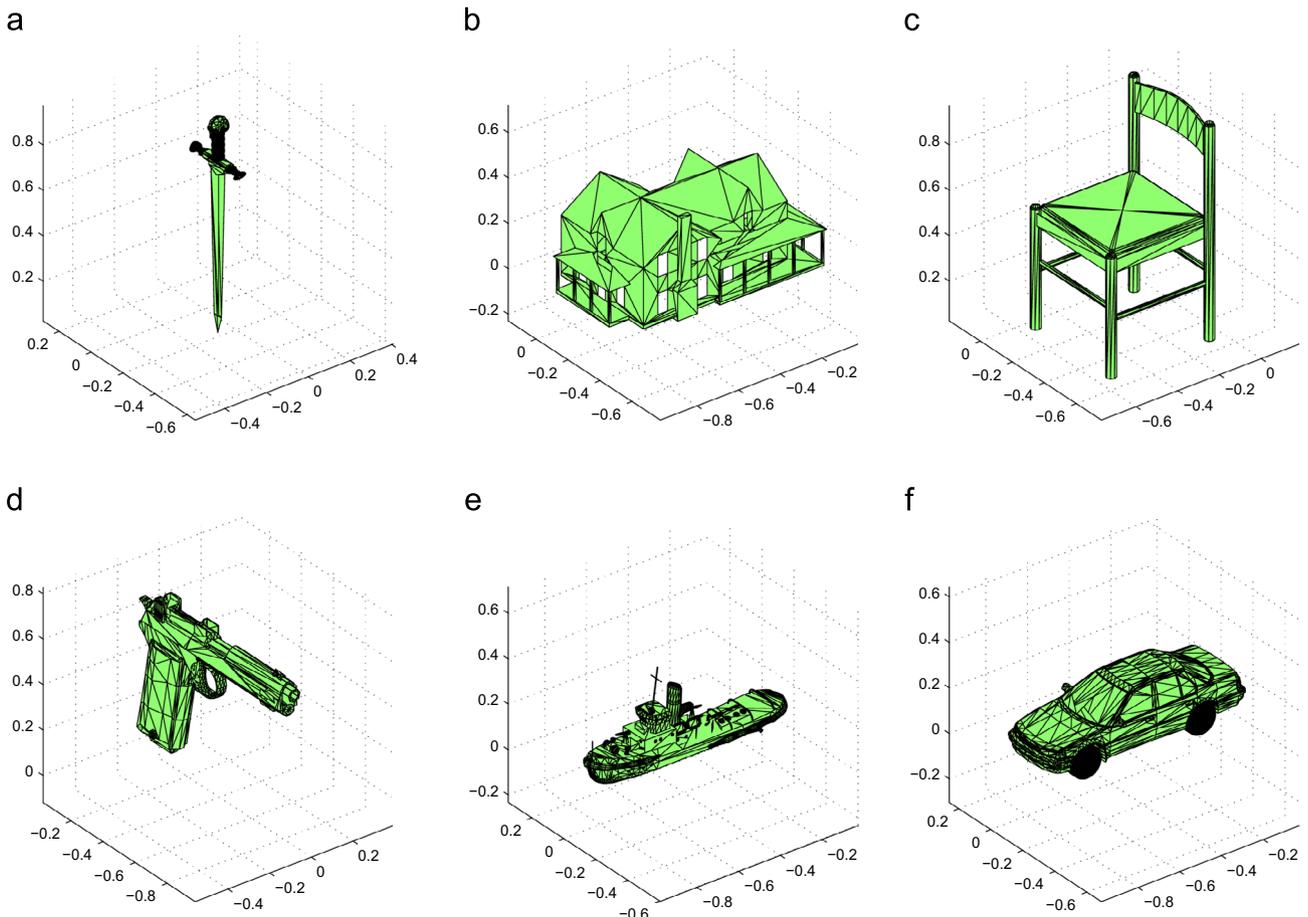


Fig. 4. Examples of the class representatives: (a) sword, (b) two story home, (c) dining chair, (d) handgun, (e) ship and (f) sedan from the Princeton Shape Benchmark.

To calculate moments from the object surface, two basic options are available. The first one is to apply the Gauss–Ostrogradsky Theorem and calculate volume moments (30) by integration over the surface, while the second option is to work directly with *surface moments* [26] defined for geometric moments as

$$s_{pqr} = \iint_S x^p y^q z^r dS, \tag{31}$$

where S is the surface of the object. For triangulated surfaces, there exists an efficient algorithm how to calculate both surface and volume moments [27]. The core formula is

$$t_{pqr} = \frac{p!q!r!}{(p+q+r+n)!} \sum_{(k_{ij}) \in \mathcal{K}} \frac{\prod_{j=1}^3 \left(\left(\sum_{i=1}^3 k_{ij} \right)! \right)}{\prod_{i,j=1}^3 (k_{ij}!) } \sum_{\ell=1}^N A_{\ell} \prod_{i,j=1}^3 \left(a_{ij}^{(\ell)} \right)^{k_{ij}}, \tag{32}$$

where t_{pqr} can be either s_{pqr} or m_{pqr} , N is the number of the triangles, \mathcal{K} is a set of such 3×3 matrices k_{ij} with non-negative integer values that $\sum_{j=1}^3 k_{1j} = p$, $\sum_{j=1}^3 k_{2j} = q$ and $\sum_{j=1}^3 k_{3j} = r$, $a_{ij}^{(\ell)}$ is a matrix of the vertex coordinates of the ℓ -th triangle, i is the index of the coordinate, and j is the index of the vertex, $A_{\ell} = \| (a_{i2}^{(\ell)} - a_{i1}^{(\ell)}) \times (a_{i3}^{(\ell)} - a_{i1}^{(\ell)}) \|$ is double the oriented area of the triangle and the number of dimensions $n=2$. In MATLAB, the formula (32) can be implemented very efficiently because its inner part (the sum over all triangles) needs not to be implemented as a loop, but can be accomplished as a single-instruction matrix operation.

We can use the formula (32) with a slight modification to calculate the *volume moments*. Each triangle is extended to the tetrahedron with the fourth vertex at the origin. The result equals that of (30), where the function $f(x, y, z) = 1$ inside the tetrahedrons and $f(x, y, z) = 0$ outside them. The symbol A_{ℓ} is now 6-tuple the oriented volume of the tetrahedron with the fourth vertex at the origin $A_{\ell} = \det(a_{ij}^{(\ell)})$ and $n=3$. If the triangulated object is closed, the result does not depend on the position of the origin, therefore we fill the holes by additional triangles just in case in our experiments.

Finally, the geometric moments calculated using (32) are converted to the complex moments as follows: we express the corresponding spherical harmonics in Cartesian coordinates as a polynomial:

$$(x^2 + y^2 + z^2)^{s/2} Y_{\ell}^m(x, y, z) = \sum_{\substack{k_x, k_y, k_z = 0 \\ k_x + k_y + k_z = s}} a_{k_x, k_y, k_z} x^{k_x} y^{k_y} z^{k_z} \tag{33}$$

and then the complex moment is computed as

$$c_{s\ell}^m = \sum_{\substack{k_x, k_y, k_z = 0 \\ k_x + k_y + k_z = s}} a_{k_x, k_y, k_z} t_{k_x, k_y, k_z}. \tag{34}$$

6. Experiment on real data

The last experiment was done with real 3D objects and their real rotations. We took a teddy bear and scanned it by means of Kinect device. Then we repeated this process five times with different orientation of the teddy bear in the space. Hence, we obtained six scans differing from each other by rotation and also slightly by scale, quality of details and perhaps by some random errors. For a comparison we also scanned another teddy bear of different shapes (see Fig. 5). When using Kinect, one has to scan the object from several views and Kinect software then produces a triangulated surface of the object, which is basically the same representation as we employed in the previous section. In order to demonstrate that the invariants can be calculated not only from the surface representation but also from the volumetric representation, we converted each teddy bear figure into 3D volumetric representation of the size approximately $100 \times 100 \times 100$ voxels. Then we calculated the volumetric geometric moments of each scan by (30), converted them to the complex moments by (34) and computed the invariants $\Phi_1 - \Phi_{100}$ from them. In Fig. 6 you can see the values of the invariants $\Phi_2, \Phi_{12}, \Phi_{15}$. The values of the first teddy bear almost do not depend on the particular rotation (STD over the six instances is 0.019 (Φ_2), 0.019 (Φ_{12}) and 0.017 (Φ_{15})). On the other hand they are significantly different from the values of the second teddy bear, which demonstrates both desirable properties – invariance and discrimination power.

7. Conclusion

We have proposed and implemented an algorithm to generate 3D rotation moment invariants of arbitrary orders based on the results of group representation theory. The rotation invariants from the complex moments were computed up to the 7th order, with the result of 486 irreducible invariants. The coefficients were converted from the floating-point numbers to the square roots of rational numbers in 100 irreducible invariants up to the 5th order, a vast majority of them being published for the first time. The irreducible invariants are available on our website

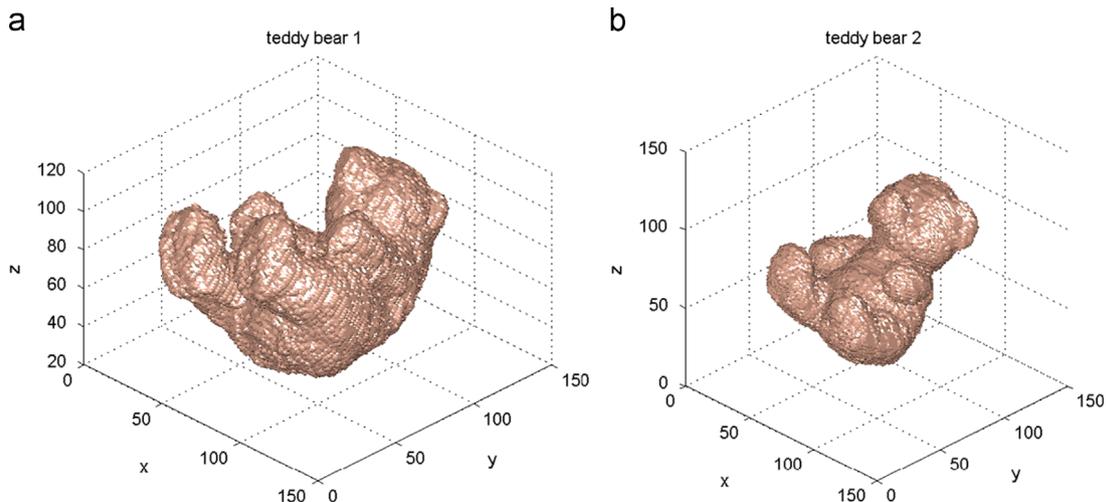


Fig. 5. Volumetric representation of two teddy bears.

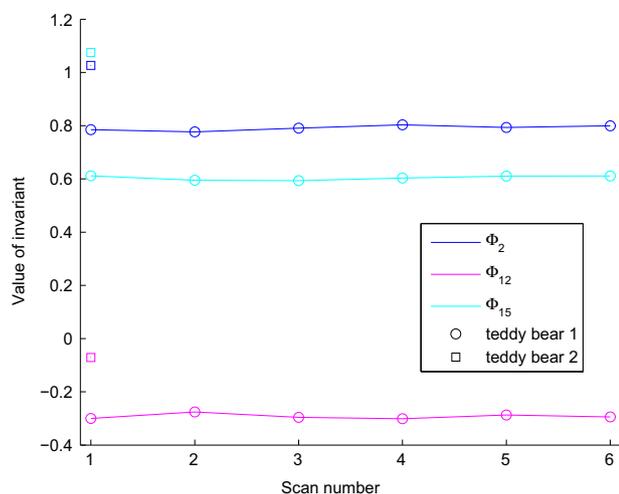


Fig. 6. Values of the invariants calculated in the experiment with the teddy bears.

[28]. The method includes elimination of linearly dependent invariants, but for now it does not contain identification of polynomial dependencies among the invariants.

Conflict of interest

None declared.

Acknowledgements

Thanks to the Grants no. GA13-29225S and GA15-16928S of the Czech Science Foundation for the financial support.

References

- [1] R. Kakarala, D. Mao, A theory of phase-sensitive rotation invariance with spherical harmonic and moment-based representations, in: IEEE Conference on Computer Vision and Pattern Recognition CVPR'10, pp. 105–112.
- [2] M. Kazhdan, An approximate and efficient method for optimal rotation alignment of 3D models, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2007) 1221–1229.
- [3] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz, Rotation invariant spherical harmonic representation of 3D shape descriptors, in: Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, SGP '03, Eurographics Association, Aire-la-Ville, Switzerland, 2003, pp. 156–164.
- [4] J. Fehr, Local rotation invariant patch descriptors for 3D vector fields, in: 20th International Conference on Pattern Recognition ICPR'10, IEEE Computer Society, Piscataway, New Jersey, USA, 2010, pp. 1381–1384.
- [5] J. Fehr, H. Burkhardt, 3D rotation invariant local binary patterns, in: 19th International Conference on Pattern Recognition ICPR'08, IEEE Computer Society, Omnipress, Madison, Wisconsin, USA, 2008, pp. 1–4.
- [6] H. Skibbe, M. Reiser, H. Burkhardt, SHOG—spherical HOG descriptors for rotation invariant 3D object detection, in: R. Mester, M. Felsberg (Eds.), Deutsche Arbeitsgemeinschaft für Mustererkennung DAGM'11, Lecture Notes in Computer Science, 6835, Springer, Berlin, Heidelberg, Germany, 2011, pp. 142–151.
- [7] J. Flusser, T. Suk, B. Zitová, Moments and Moment Invariants in Pattern Recognition, Wiley, Chichester, U.K., 2009.
- [8] F.A. Sadjadi, E.L. Hall, Three dimensional moment invariants, IEEE Trans. Pattern Anal. Mach. Intell. 2 (1980) 127–136.
- [9] X. Guo, Three-dimensional moment invariants under rigid transformation, in: D. Chetverikov, W. Kropatsch (Eds.), Proceedings of the Fifth International Conference on Computer Analysis of Images and Patterns CAIP'93, Lecture Notes in Computer Science, vol. 719, Springer, Berlin, Heidelberg, Germany, 1993, pp. 518–522.
- [10] J.M. Galvez, M. Canton, Normalization and shape recognition of three-dimensional objects by 3D moments, Pattern Recognit. 26 (1993) 667–681.
- [11] D. Cyganski, J.A. Orr, Object recognition and orientation determination by tensor methods, in: T.S. Huang (Ed.), Advances in Computer Vision and Image Processing, JAI Press, Greenwich, Connecticut, USA, 1988, pp. 101–144.
- [12] D. Xu, H. Li, 3-D affine moment invariants generated by geometric primitives, in: Proceedings of the 18th International Conference on Pattern Recognition ICPR'06, IEEE Computer Society, Los Alamitos, California, USA, 2006, pp. 544–547.
- [13] D. Xu, H. Li, Geometric moment invariants, Pattern Recognit. 41 (2008) 240–249.
- [14] C.-H. Lo, H.-S. Don, 3-D moment forms: their construction and application to object identification and positioning, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1989) 1053–1064.
- [15] J. Flusser, J. Boldýš, B. Zitová, Moment forms invariant to rotation and blur in arbitrary number of dimensions, IEEE Trans. Pattern Anal. Mach. Intell. 25 (2003) 234–246.
- [16] J.-F. Mangin, F. Poupon, E. Duchesnay, D. Riviere, A. Cachia, D.L. Collins, A. Evans, J. Regis, Brain morphometry using 3D moment invariants, Med. Image Anal. 8 (2004) 187–196.
- [17] M. Trummer, H. Suesse, J. Denzler, Coarse registration of 3D surface triangulations based on moment invariants with applications to object alignment and identification, in: Proceedings of the 12th International Conference on Computer Vision, ICCV'09, IEEE, 2009, pp. 1273–1279.
- [18] J.P. Elliot, P.G. Dawber, Symmetry in Physics, MacMillan, London, U.K., 1979.
- [19] A.G. Mamistvalov, n -dimensional moment invariants and conceptual mathematical theory of recognition n -dimensional solids, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 819–831.
- [20] J. Boldýš, J. Flusser, Extension of moment features' invariance to blur, J. Math. Imaging Vis. 32 (2008) 227–238.
- [21] Y.S. Abu-Mostafa, D. Psaltis, Recognitive aspects of moment invariants, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1984) 698–706.
- [22] J. Flusser, On the independence of rotation moment invariants, Pattern Recognit. 33 (2000) 1405–1410.
- [23] J. Flusser, T. Suk, Rotation moment invariants for recognition of symmetric objects, IEEE Trans. Image Process. 15 (2006) 3784–3790.
- [24] T. Suk, J. Flusser, Recognition of symmetric 3d bodies, Symmetry 6 (2014) 722–757.
- [25] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The Princeton Shape Benchmark (<http://shape.cs.princeton.edu/benchmark/>), 2004.
- [26] D. Xu, H. Li, 3-D surface moment invariants, in: Proceedings of the 18th International Conference on Pattern Recognition ICPR'06, IEEE Computer Society, 2006, pp. 173–176.
- [27] S.A. Sheynin, A.V. Tuzikov, Explicit formulae for polyhedra moments, Pattern Recognit. Lett. 22 (2001) 1103–1109.
- [28] DIP, 3D Rotation Moment Invariants (<http://zoi.utia.cas.cz/3DRotationInvariants,2011.>).

Tomáš Suk received the M.Sc. degree in electrical engineering from the Czech Technical University, Faculty of Electrical Engineering, Prague, 1987. The Csc. degree (corresponds to Ph.D.) in Computer science from the Czechoslovak Academy of Sciences, Prague, 1992. From 1991 he is a researcher with the Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Prague, member of Department of Image Processing. He has authored 14 journal papers and 31 conference papers. He is a co-author of the book “Moments and Moment Invariants in Pattern Recognition” (Wiley, 2009). Tomáš Suk's research interests include digital image processing, pattern recognition, image filtering, invariant features, moment and point invariants, geometric transformations of images, and applications in remote sensing, astronomy, medicine and computer vision. In 2002 Tomáš Suk received the Otto Wichterle premium of the Academy of Sciences of the Czech Republic for young scientists.

Jan Flusser received the M.Sc. degree in mathematical engineering from the Czech Technical University, Prague, Czech Republic in 1985, the Ph.D. degree in computer science from the Czechoslovak Academy of Sciences in 1990, and the D.Sc. degree in technical cybernetics in 2001. Since 1985 he has been with the Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Prague. In 1995–2007, he was holding the position of a head of Department of Image Processing. Currently (since 2007) he is a Director of the Institute. He is a full professor of computer science at the Czech Technical University and at the Charles University, Prague, Czech Republic, where he gives undergraduate and graduate courses on Digital Image Processing, Pattern Recognition, and Moment Invariants and Wavelets. Jan Flusser's research interest covers moments and moment invariants, image registration, image fusion, multichannel blind deconvolution, and super-resolution imaging. He has authored and

coauthored more than 150 research publications in these areas, including the monograph “Moments and Moment Invariants in Pattern Recognition” (Wiley, 2009), tutorials and invited/keynote talks at major international conferences. In 2007 Jan Flusser received the Award of the Chairman of the Czech Science Foundation for the best research project and won the Prize of the Academy of Sciences of the Czech Republic for the contribution to image fusion theory. In 2010, he was awarded by the prestigious SCOPUS 1000 Award presented by Elsevier.

Jiri Boldyš received the M.Sc. degree in physics from the Charles University, Prague, Czech Republic in 1996, and the Ph.D. degree in mathematical and computer modeling from the same university in 2004. Since 1998, he has been with the Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Prague. He has also been working in private companies in the fields of image processing and mathematical modelling. His research interests include mainly object and pattern recognition, moment invariants, physics and image processing in nuclear medicine, medical imaging, wavelet transform and thin film image processing.