
Volume 44, Issue 9
September 2011
ISSN 0031-3203
ELSEVIER
PATTERN
RECOGNITION
SPECIAL ISSUE SECTION: COMPUTER ANALYSIS OF IMAGES AND PATTERNS
GUEST EDITORS: XIAOYI JIANG AND NICOLAI PETKOV
Available online at www.sciencedirect.com
ScienceDirect

Pattern Recognition 44 (2011) 2047–2056

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

# Affine moment invariants generated by graph method

Tomáš Suk *, Jan Flusser

*Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 4, 18208 Praha 8, Czech Republic*

ARTICLE INFO

Available online 31 May 2010

*Keywords:*
Image moments
Object recognition
Affine transformation
Affine moment invariants
Pseudoinvariants
Graph representation
Irreducibility
Independence

ABSTRACT

The paper presents a general method of an automatic deriving affine moment invariants of any weights and orders. The method is based on representation of the invariants by graphs. We propose an algorithm for eliminating reducible and dependent invariants. This method represents a systematic approach to the generation of all relevant moment features for recognition of affinely distorted objects. We also show the difference between pseudoinvariants and true invariants.



## 1. Introduction

Recognition of objects and patterns that are deformed in various ways has been a goal of much recent research. There are basically three major approaches to this problem—brute force (full search), image normalization, and invariant features. In the brute force approach we search the parametric space of all possible image degradations which leads to extreme time complexity. In the normalization approach, the objects are transformed into a certain standard position before they enter the classifier. This is very efficient in the classification stage but the object normalization itself usually requires solving difficult inverse problems that are often ill-conditioned or even ill-posed.

The approach using invariant features has appeared to be the most promising and has been used extensively. Its basic idea is to describe the objects by a set of measurable quantities called *invariants* that are insensitive to particular deformations and that provide enough discrimination power to distinguish among objects belonging to different classes. From mathematical point of view, invariant $I$ is a functional defined on the space of all admissible image functions which does not change its value under degradation operator $\mathcal{D}$, i.e. which satisfies the condition $I(f)=I(\mathcal{D}(f))$ for any image function $f$. This property is called *invariance*. Another desirable property of $I$, as important as invariance, is *discriminability*. For objects belonging to different classes, $I$ must have significantly different values. Clearly, these two requirements are antagonistic—the broader the invariance, the less discrimination power and vice versa. Choosing a proper trade-off between invariance and discrimination power is a very important task in feature-based object recognition.

The existing invariant features used for describing 2D objects can be categorized from various points of view. Most straightforward is the categorization according to the type of invariance. We recognize translation, rotation, scaling, affine, projective, and elastic geometric invariants. Another possible categorization is according to the mathematical tools used and yet another viewpoint reflects what part of the object is needed to calculate the invariant. One can find hundreds of papers belonging to each category.

In this paper we focus on invariants with respect to affine transform of spatial coordinates, which are based on image moments. Such features are called *Affine moment invariants* (*AMIs*).

Invariants to affine transform play a very important role in object recognition. In most cases, we represent 3D objects and structures by their projections onto a 2D plane because photography is a 2D medium. An exact model of photographing a planar scene by a pin-hole camera whose optical axis is not perpendicular to the scene is a projective transform

$$x' = \frac{a_0+a_1x+a_2y}{1+c_1x+c_2y},$$

$$y' = \frac{b_0+b_1x+b_2y}{1+c_1x+c_2y}. \tag{1}$$

Since the projective transform is non-linear, construction of projective invariants is generally difficult; construction of projective moment invariants is even impossible (see [1,2] for details). Fortunately, affine transformation (which is linear) can—under certain circumstances—approximate the projective transform.

* Corresponding author.
*E-mail addresses:* suk@utia.cas.cz (T. Suk), flusser@utia.cas.cz (J. Flusser).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.05.015

Affine transformation is defined as

$$x' = a_0 + a_1 x + a_2 y,$$

$$y' = b_0 + b_1 x + b_2 y. \qquad (2)$$

It maps a square onto a general parallelogram and preserves collinearity (see Fig. 1). The Jacobian of the affine transform is $J = a_1 b_2 - a_2 b_1$. In this paper, we consider only regular affine transforms whose Jacobian is non-zero. Contrary to the projective transform, $J$ does not depend on the spatial coordinates $x$ and $y$, which makes searching for invariants easier. Affine transform is a particular case of the projective transform with $c_1 = c_2 = 0$. If the object is small comparing to the camera-to-scene distance, then both $c_1$ and $c_2$ approach zero, the perspective effect becomes negligible, and the affine model is a reasonable approximation of the projective model. This is why the affine transform and affine invariants are so important in computer vision.

Among various affine invariants published in the literature, AMIs have an important position. Their rich history began almost 50 years ago. The first attempt to derive moment invariants to affine transform was presented already in the first Hu's paper [3] but the *Fundamental theorem of affine invariants* was stated incorrectly there. Thirty years later, Reiss [4] and Flusser and Suk [5,6] independently discovered and corrected this mistake, published new sets of the AMI's and proved their applicability in simple recognition tasks. In their papers, the derivation of the AMI's originated from the traditional theory of algebraic invariants from the 19th century, e.g. [7–11]. The same results achieved by a slightly different approach were later published by Mamistvalov [12].

The AMI's can be derived in several ways that differ from each other by mathematical tools used [13]. Apart from the above mentioned algebraic invariants, one may use graph theory, tensor algebra, direct solution of proper partial differential equations, and derivation via image normalization. All these methods end up with equivalent sets of invariants.

In this paper, we present a method using graph theory. It is probably the simplest and the most transparent way allowing to generate systematically affine moment invariants of any orders and weights. In a preliminary version, it was proposed in [14] and is similar to an earlier *tensor method* [15] and geometric moment invariants [16]. One of the main advantages of the graph method is that it provides an insight into the structure of the invariants and allows to eliminate the dependencies among them. The algorithm which identifies dependent invariants is a major contribution of this paper.

The automatic generation of the affine moment invariants based on the graph theory is described in Section 3. The algorithm for elimination of the dependent invariants is presented in Section 4. In Section 5 we show two illustrative numerical experiments.

## 2. Basic terms

Let us introduce a few basic terms first.

By *image function* (or *image*) we understand any real function $f(x,y)$ having a bounded support and a finite non-zero integral.

*Geometric moment* $m_{pq}^{(f)}$ of the image $f(x,y)$, where $p$, $q$ are non-negative integers and $(p+q)$ is called *order* of the moment, is defined as

$$m_{pq}^{(f)} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y)\,\mathrm{d}x\,\mathrm{d}y. \qquad (3)$$

Corresponding *central moment* $\mu_{pq}^{(f)}$ is defined as

$$\mu_{pq}^{(f)} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x-x_c)^p (y-y_c)^q f(x,y)\,dx\,dy, \qquad (4)$$

where the coordinates $x_c = m_{10}/m_{00}$, $y_c = m_{01}/m_{00}$ denote the centroid of $f(x,y)$. Central moments are invariant to translation of the image, hence any function of central moments is shift invariant.

The theory of affine moment invariants has been traditionally connected to the theory of algebraic invariants. Hilbert [11] defined an *algebraic invariant* as a polynomial function of the coefficients $a,b,\ldots$ of binary forms that satisfies the equation

$$I(a_0',a_1',\ldots,a_{p_a}';b_0',b_1',\ldots,b_{p_b}';\ldots) = J^w I(a_0,a_1,\ldots,a_{p_a};b_0,b_1,\ldots,b_{p_b};\ldots), \qquad (5)$$

where $w$ is called the *weight* of the invariant. The Fundamental theorem describes the link between algebraic and moment invariants.

**Theorem 1.** *Fundamental theorem of affine moment invariants. If the binary forms of orders $p_a$, $p_b,\ldots$ have an algebraic invariant of weight $w$ and degree $r$*

$$I(a_0',a_1',\ldots,a_{p_a}';b_0',b_1',\ldots,b_{p_b}';\ldots) = J^w I(a_0,a_1,\ldots,a_{p_a};b_0,b_1,\ldots,b_{p_b};\ldots),$$

*then the moments of the same orders have the same invariant but with the additional factor $|J|^r$:*

$$I(\mu_{p_a0}',\mu_{p_a-1,1}',\ldots,\mu_{0p_a}';\mu_{p_b0}',\mu_{p_b-1,1}',\ldots,\mu_{0p_b}';\ldots)$$
$$= J^w |J|^r I(\mu_{p_a0},\mu_{p_a-1,1},\ldots,\mu_{0p_a};\mu_{p_b0},\mu_{p_b-1,1},\ldots,\mu_{0p_b};\ldots).$$

If we knew an algebraic invariant, we can construct a corresponding affine moment invariant easily just by interchanging the coefficients and the central moments and

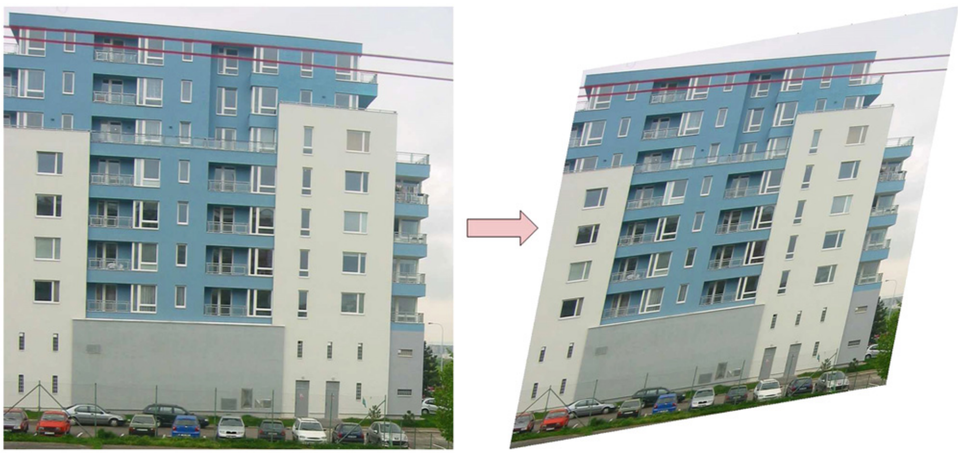


**Fig. 1.** The affine transform maps a square to a parallelogram.

normalizing the invariant by $\mu_{00}^{w+r}$ to eliminate the factor $J^w \cdot |J|^r$. Unfortunately, the Theorem does not provide any instructions how to find algebraic invariants and this is why many other techniques of the AMI's construction have appeared. In the rest of the paper, the Theorem is not employed at all.

## 3. AMIs generated by graphs

The construction of the AMIs which we call *graph method* is based on creating all possible functionals of the form

$$I(f) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \prod_{k,j=1}^{r} C_{kj}^{n_{kj}} \cdot \prod_{i=1}^{r} f(x_i,y_i)\ \mathrm{d}x_i\ \mathrm{d}y_i, \tag{6}$$

where the cross-product $C_{kj} = x_k y_j - x_j y_k$ is the oriented double area of the triangle, whose vertices are $(x_k, y_k)$, $(x_j, y_j)$, and $(0,0)$, and $n_{kj}$ are non-negative integers. Note that it is meaningful to consider only $j > k$, because $C_{kj} = -C_{jk}$ and $C_{kk} = 0$. After an affine transform (for simplicity, we assume $a_0 = b_0 = 0$) it holds $C'_{kj} = J \cdot C_{kj}$. Hence, functional $I(f)$ becomes after the transform

$$I(f)' = J^w |J|^r \cdot I(f), \tag{7}$$

where $w = \sum_{k,j} n_{kj}$ is the *weight* of the invariant and $r$ is the *degree* of the invariant (these terms are equivalent to those used in the Fundamental theorem). If $I(f)$ is normalized by $\mu_{00}^{w+r}$, we obtain a desirable absolute affine invariant

$$\left(\frac{I(f)}{\mu_{00}^{w+r}}\right)' = (\mathrm{sign}\ J)^w \left(\frac{I(f)}{\mu_{00}^{w+r}}\right). \tag{8}$$

The maximum order of moments contained in the invariant is called the *order* of the invariant. The order is always less than or equal to the weight. Another important characteristic of the invariant is its *structure*. The structure of the invariant is defined by an integer vector $\mathbf{s} = (k_2, k_3, \ldots, k_s)$, where $s$ is the invariant order and $k_j$ is the total number of moments of the $j$th order contained in each term of the invariant (all terms have the same structure). Since always $k_0 = k_1 = 0$, these two quantities are not included in the structure vector.

If $w$ is odd then the invariant changes its sign if the respective affine transform contains mirror reflection (i.e. if $J < 0$). Such invariants are not "true" AMIs and are called *pseudoinvariants* [17]. As we will see in the experiment, the use of pseudoinvariants may be desirable if mirror reflection of the objects cannot occur. On the other hand, if admissible transforms may contain mirroring, the use of the pseudoinvariants would be misleading and they should be avoided. The pseudoinvariants are also useless for recognition of axially symmetric objects. All pseudoinvariants of any object with at least one axis of symmetry are identically zero. One of the advantages of the graph method is that from Eq. (6) we can immediately see whether or not the given functional $I$ is true invariant or pseudoinvariant.

We illustrate the use of the general formula (6) on two simple invariants. First, let $r = 2$ and $w = n_{12} = 2$. Then

$$I(f) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (x_1 y_2 - x_2 y_1)^2 f(x_1,y_1) f(x_2,y_2)\, \mathrm{d}x_1\, \mathrm{d}y_1\, \mathrm{d}x_2\, \mathrm{d}y_2 = 2(m_{20}m_{02} - m_{11}^2). \tag{9}$$

When replacing the geometric moments by corresponding central moments in order to handle also image translation and when normalizing the invariant by $\mu_{00}^4$ we obtain a true invariant to general affine transform

$$I_1 = (\mu_{20}\mu_{02} - \mu_{11}^2)/\mu_{00}^4.$$

This is the simplest affine invariant, uniquely containing the second-order moments only. Its structure is $\mathbf{s} = (2)$. In this form it is commonly referred to in the literature regardless of the method used for derivation.

Similarly, for $r = 3$ and $n_{12} = 2$, $n_{13} = 2$, $n_{23} = 0$ we obtain

$$I(f) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (x_1 y_2 - x_2 y_1)^2 (x_1 y_3 - x_3 y_1)^2 f(x_1,y_1) f(x_2,y_2) f(x_3,y_3) \times \mathrm{d}x_1\, \mathrm{d}y_1\, \mathrm{d}x_2\, \mathrm{d}y_2\, \mathrm{d}x_3\, \mathrm{d}y_3 = m_{20}^2 m_{04} - 4m_{20}m_{11}m_{13} + 2m_{20}m_{02}m_{22} + 4m_{11}^2 m_{22} - 4m_{11}m_{02}m_{31} + m_{02}^2 m_{40} \tag{10}$$

and the normalizing factor is in this case $\mu_{00}^7$. Both the weight and the order of this invariant equal 4 and its structure is $\mathbf{s} = (2, 0, 1)$.

### 3.1. Representing the invariants by graphs

In this section we explain the link between Eq. (6) and graph theory. Each invariant generated by the formula (6) can be represented by a connected graph, where each point $(x_k, y_k)$ corresponds to a node $k$ and each cross-product $C_{kj}$ corresponds to an edge $kj$ between the nodes $k$ and $j$. We allow multiple edges between two nodes. The number $n_{kj}$ says, how many edges connect the nodes $k$ and $j$ (such graphs are called *multigraphs*). Thus, the number of nodes equals the degree of the invariant and the total number of the graph edges equals the weight $w$ of the invariant. From the graph one can also learn about the orders of the moments the invariant is composed of and about its structure. The number of edges incident to each node equals the order of the moments involved. Each invariant of the form (6) is in fact a sum where each term is a product of a certain number of moments. This number, the degree of the invariant, is constant for all terms of one particular invariant and is equal to the total number of graph nodes. Two examples of such graphs are in Fig. 2.

Now one can see that the problem of derivation of the AMIs up to the given weight $w$ is equivalent to generating all connected graphs with at least two nodes and at most $w$ edges. Let us denote this set of graphs as $G_w$. Generating all graphs of $G_w$ is a combinatorial task with exponential complexity but formally easy to implement. It should be noted that some elements of $G_w$ are meaningless and their generation can be skipped; for instance any graph with a node having only one incident edge leads always to zero invariant because $\mu_{10} = \mu_{10} = 0$.

Each graph is represented in the algorithm by the list of its edges. The list of edges for graph (9) is

$$\begin{matrix} 1 & 1 \\ 2 & 2 \end{matrix} \tag{11}$$

and for graph (10) it is

$$\begin{matrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 3 & 3. \end{matrix} \tag{12}$$

If we need to generate all invariants of a certain weight $w$, it is sufficient to generate all graphs from

$$\begin{matrix} 1 & 1 & 1 & \ldots & 1 & 1 & 1 \\ 2 & 2 & 2 & \ldots & 2 & 2 & 2 \end{matrix} \tag{13}$$

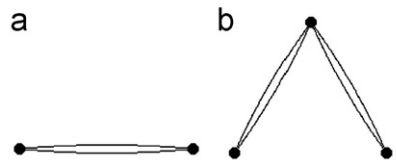



**Fig. 2.** The graphs corresponding to the invariants: (a) from (9) and (b) from (10).

to

$$\begin{matrix} w-1 & w-1 & w-1 & \ldots & w-1 & w-1 & w-1 \\ w & w & w & \ldots & w & w & w. \end{matrix} \quad (14)$$

The algorithm which generates all graphs of $G_w$ starts with the graph (13). Then iteratively repeats the following loop which in each run generates the next graph.

*Step* 1: Find the first element from the end of the second row that can be increased, i.e. that is less than the corresponding value in (14).

*Step* 2: If it exists then increase it by one to $\nu_1$. Fill the rest of the row by the greater of two values: $\nu_1$ and $a_1+1$, where $a_1$ is the value in the first row above the filled one.

*Step* 3: If it does not exist then find the first element from the end of the first row, which can be increased.

*Step* 4: If it exists then increase it by one to $\nu_2$. Fill the rest of the row by $\nu_2$.

*Step* 5: If it does not exist then stop else go to Step 1.

This algorithm guarantees that no relevant graph/invariant is omitted. However, it generates some invariants which are dependent and should be removed afterwards. The next section presents a method of identifying such invariants.

## 4. Independence of the AMI's

The above construction algorithm does not guarantee that there are no dependent invariants in the generated set. Actually, the number of dependent invariants may be much higher than that of the independent ones. The dependent invariants do not increase the discrimination power of the recognition system at all while increasing the dimensionality of the problem, which leads to growth of the complexity and even to misclassifications. Using dependent features in recognition tasks is a serious mistake. Thus, identifying and discarding dependent invariants is highly desirable.

### 4.1. The number of independent invariants

Before we proceed to the selection of independent invariants, it is worth analyzing how many invariants may exist. An intuitive rule suggests that the number $n$ of independent invariants created from $m$ independent measurements (i.e. moments in our case) is

$$n = m - p, \quad (15)$$

where $p$ is the number of independent constraints that must be satisfied (see, e.g., Ref. [1]). This is true but the number of independent constraints is often hard to determine. The dependencies both in constraints and in measurements may be hidden. It may be hard to identify those measurements and constraints that are actually independent and those that are not.

In most cases we estimate $p$ as the number of transformation parameters. This works perfectly in the case of rotation but for affine transform this estimate is not true in general. An affine transform has six parameters, so we would not expect any second-order affine invariant (6 moments − 6 parameters = 0). However, in the previous section we proved the existence of the second-order invariant $I_1$. On the other hand, for invariants up to the third order we have 10 moments − 6 parameters = 4, which is actually the correct number of independent invariants. The same estimation works well for the fourth order (15 − 6 = 9 invariants) and for all higher orders where the actual number of independent invariants is known. It is a common belief (although not exactly proven) that the order $r=2$ is the only exception to this rule.

A detailed analysis of the number of independent AMIs regarding their weights, degrees and structures can be found in [18]. It is based on classic results of the theory of algebraic invariants, particularly on the Cayley–Sylvester theorem [9,11].

### 4.2. Possible dependencies among the AMIs

There might be various kinds of dependency in the set of all AMIs (i.e. in the set $G_w$ of all graphs). Let us categorize them into five groups and explain how they can be eliminated.

1. *Zero invariants*: Some AMIs may be identically zero regardless of the image on which they are calculated. If there are one or more nodes with one incident edge only, then all terms of the invariants contain first-order moment(s). When using central moments, they are zero by definition and, consequently, such an invariant is zero, too. However, some other graphs may also generate zero invariants due to the terms cancellation, for instance the graph in Fig. 3 leads to

$$I(f) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (x_1y_2 - x_2y_1)^3 f(x_1,y_1) f(x_2,y_2)\, dx_1\, dy_1\, dx_2\, dy_2 = m_{30}m_{03} - 3m_{21}m_{21} + 3m_{21}m_{21} - m_{30}m_{03} = 0. \quad (16)$$

2. *Identical invariants*: All isomorphic graphs (and also some non-isomorphic ones) generate identical invariants. Elimination can be done by comparing the invariants termwise.
3. *Products*: Some invariants may be products of other invariants.
4. *Linear combinations*: Some invariants may be linear combinations of other invariants.
5. *Polynomial dependencies*: If there exists a finite sum of products of invariants (including their integer powers) that equals zero, the invariants involved are polynomially dependent.

The invariants suffering from the dependencies 1–4 are called *reducible* invariants. After eliminating all of them, we obtain a set of so called *irreducible* invariants. However, irreducibility does not mean independence, as we will see later. In the following two sections we show how to detect reducible and polynomially dependent invariants.

### 4.3. Removing reducible invariants

The dependencies 1 and 2 are trivial and easy to find. To remove the products (type 3), we perform an incremental exhaustive search. All possible pairs of the admissible invariants (the sum of their individual weights must not exceed $w$) are multiplied and the independence of the result is checked.

To remove the dependencies of the type 4 (linear combinations), all possible linear combinations of the admissible invariants (the invariants must have the same structure) are considered and their independence is checked. The algorithm should not miss any possible combination and can be implemented as follows.

Since only invariants of the same structure can be linearly dependent, we sort the invariants (including all products) according to the structure first. For each group of the same structure we construct a matrix of coefficients of all invariants. The "coefficient" here means the multiplicative constant of each term; if the invariant does not contain all possible terms, the corresponding coefficients are zero. Thanks to this, all coefficient vectors are of the same length and we can actually arrange them into a matrix. If this matrix has a full rank, all invariants are

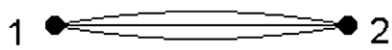



**Fig. 3.** The graph leading to the zero invariant (16).

linearly independent. To calculate the rank of the matrix, we use singular value decomposition (SVD). The rank of the matrix is given by the number of non-zero singular values and equals the number of linearly independent invariants of the given structure. Each zero singular value corresponds to a linearly dependent invariant that should be removed.

At the end of the above procedure we obtain a set of all irreducible invariants up to the given weight or order. However, it is of exponential complexity and thus is a very expensive procedure even for low weights. On the other hand, this procedure is run only once. As soon as the invariants are known in explicit forms they can be used in future experiments without repeating the derivation.

Table 1 presents the number of invariants found by the above method for $w \le 12$. From left to right, the table shows the weight, the total number of all generated graphs, the number of zero invariants eliminated afterwards, the number of invariants which are identical to some others, the number of invariants which are products of some others, the number of invariants which are linear combinations of some others or linear combinations of products, the number of irreducible invariants, and the number of independent invariants, respectively. Each row shows cumulative values for all weights less or equal $w$. Note that the number of independent invariants is much lower than the number of invariants originally generated. This illustrates a crucial role of the elimination algorithm.

A few examples of irreducible invariants which will be further used in experiments are shown in Appendix. A full set of 1589 irreducible invariants of $w \le 12$ can be found in [19], the accompanying web pages of [13].

### 4.4. Removing polynomial dependencies

Polynomial dependencies among irreducible invariants pose a serious problem because the dependent invariants cannot be easily identified and removed. Unfortunately we cannot ignore them. Out of 1589 irreducible invariants of the weight of 12 and less, only 85 invariants at most can be independent, which means that at least 1504 of them must be polynomially dependent. An algorithm for a complete removal of dependent invariants is known in principle, we can use the algorithms for elimination of products and linearly dependent invariants with higher weight limit.

Let us suppose we have generated some set of invariants with a weight limit $w_g$. If we want to eliminate linearly dependent invariants only, then it is sufficient to compute all products of all pairs of invariants up to the weight $w_g$. If we want also to eliminate polynomially dependent invariants, we must continue the multiplication up to some weight limit $w_p > w_g$ (ideally $w_p \gg w_g$). As the result, all invariants with weight $w > w_g$ will be products of some others. Now we can use the algorithm for elimination of the linearly dependent invariants with the weight limit $w_p$ again. If we find some product that is linearly dependent at the same time, we have found a polynomial dependency among the invariants.

**Table 1**
The numbers of invariants with various types of dependency up to the given weight.

| Weight | Graphs | Zero | Ident. | Prod. | Linear | Irr. | Ind. |
|---|---|---|---|---|---|---|---|
| 2 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 3 | 6 | 5 | 0 | 0 | 0 | 1 | 1 |
| 4 | 40 | 31 | 4 | 1 | 0 | 4 | 4 |
| 5 | 300 | 287 | 7 | 1 | 0 | 5 | 5 |
| 6 | 2475 | 2344 | 105 | 5 | 3 | 18 | 17 |
| 7 | 22 022 | 21 632 | 345 | 6 | 8 | 31 | 25 |
| 8 | 208 208 | 205 544 | 2495 | 32 | 60 | 77 | 33 |
| 9 | 2 068 560 | 2 057 804 | 10 358 | 54 | 184 | 160 | 44 |
| 10 | 21 414 900 | 21 352 373 | 61 008 | 227 | 930 | 362 | 55 |
| 11 | 229 523 800 | 229 236 771 | 282 474 | 526 | 3287 | 742 | 66 |
| 12 | 2 533 942 752 | 2 532 349 394 | 1 575 126 | 2105 | 14 538 | 1589 | 80 |

Many dependencies were discovered by this method. For instance, among the invariants listed in Appendix, there are two polynomial dependencies

$$-4I_1^3I_2^2+12I_1^2I_2I_3^2-12I_1I_3^4-I_2I_4^2+4I_3^3I_4-I_5^2=0, \tag{17}$$

and

$$-16I_1^3I_7^2-8I_1^2I_6I_7I_8-I_1I_6^2I_8^2+4I_1I_6I_9^2+12I_1I_7I_8I_9+I_6I_8^2I_9-I_7I_8^3-4I_9^3-I_{10}^2=0. \tag{18}$$

This proves that the invariant $I_5$ is dependent on $\{I_1,I_2,I_3,I_4\}$ and $I_{10}$ is dependent on $\{I_1,I_6,I_7,I_8,I_9\}$. To obtain a complete and independent set of AMIs up to the fourth order, $I_5$ and $I_{10}$ are omitted and $I_{19}$ is added to $\{I_1, I_2, I_3, I_4, I_6, I_7, I_8, I_9\}$.

This algorithm has two principal drawbacks. It performs in fact an exhaustive search (with certain constraints) which is very expensive even for small $w$. Moreover, we can easily get beyond the capacity of our computer. The other drawback is caused by higher-order dependencies. In the previous case of linear dependencies, an invariant that has been proven to be a linear combination of other invariants was simply omitted. This cannot be done in the case of polynomial dependencies because the identified dependencies among invariants may not be independent. Let us illustrate this in a hypothetical example. Assume $I_a$, $I_b$, $I_c$ and $I_d$ to be irreducible invariants with the three dependencies:

$$S_1 : I_a^2-I_cI_d^2=0,$$

$$S_2 : I_b^2+I_cI_d=0,$$

$$S_3 : I_b^4+2I_b^2I_cI_d+I_a^2I_c=0.$$

If we claimed three invariants to be dependent and we omitted them, it would be a mistake because the third dependency is a combination of the first and the second one

$$I_cS_1+S_2^2-S_3=0$$

and does not bring any new information. Among these four invariants, only two of them are dependent and two are independent. This is an example of a *second-order dependency*, i.e. "dependency among dependencies", while $S_1$, $S_2$, and $S_3$ are first-order dependencies. The second-order dependencies may be of the same kind as the first-order dependencies–identities, products, linear combinations and polynomials. They can be found in the same way; the algorithm from the previous section requires only minor modifications.

This consideration can be further extended and we can define $k$th order dependencies. The number $n$ of independent invariants is then

$$n=n_0-n_1+n_2-n_3+\cdots, \tag{19}$$

where $n_0$ is the number of irreducible invariants, $n_1$ is the number of first-order dependencies, $n_2$ is the number of second-order dependencies, etc. If we consider only the invariants of a certain finite order, this chain is always finite (the proof of finiteness for algebraic invariants originates from Hilbert (a reference is in [11], the proof for moment invariants is essentially the same).

## 5. Numerical experiments

In this paper, we do not repeat "traditional" numerical experiments with the AMIs which show (usually on simulated data) that the AMIs are actually invariant with respect to affine transform and also study their robustness to additive noise and to small non-linear perturbations. Such experiments were carried out in earlier publications (see [5,16] for instance) and their results are of course valid also for the AMIs presented in this paper. Here, we demonstrate two phenomena. In the first experiment we illustrate the importance of using independent set for object recognition. The second experiment clarifies the difference between "true" invariants and pseudoinvariants. Both experiments are carried out on real photographs taken by the camera Canon PowerShot S30. Geometric distortions of the objects were introduced solely by projecting 3D world onto a 2D photograph; no computer-generated distortions were applied.

### 5.1. Recognition of the scrabble tiles

In this experiment we tested the AMIs' ability to recognize letters on the scrabble tiles[1] First, we photographed all letters side by side from an almost perpendicular view (see Fig. 4). They were automatically segmented and separately binarized (mostly by means of tools available in MATLAB Image Processing Toolbox 2.0). The threshold for conversion to binary images was found by Otsu's method (function "graythresh" in MATLAB), then individual objects were found and labeled. The objects larger than 20 000 pixels (parts of the background) and smaller than 1000 pixels (shadows) were removed. Similarly, holes less than 10 pixels (dust) in the objects were removed. The result is a "gallery" of 21 binary letters that served as our template set.

Then we placed the tiles almost randomly on the table, we only ensured that no tiles are upside down and that all letters are completely visible. We took 9 photographs of such scene variously rotated (see Fig. 5 for one of them). On each photograph, the letters were automatically located and segmented by the same method as in case of the template set. We classified all the 168 letters by minimum distance with respect to the templates in the space of invariants $I_1$, $I_2$, $I_3$, $I_4$, $I_5$, $I_6$, $I_7$, $I_8$, $I_9$, $I_{10}$, $I_{19}$, $I_{47}$, $I_{48}$, $I_{49}$, $I_{50}$, $I_{64}$ and $I_{65}$ (see Fig. 6 for a 2D subspace of $I_2$ and $I_6$). There were nine misclassifications. However, as shown in the previous section, the AMIs $I_5$ and $I_{10}$ are dependent. We omitted them and run the classification again. Now the number of misclassifications decreased to 5. Finally, for a comparison, we omitted from the initial set two independent invariants $I_{64}$ and $I_{65}$. We received 16 misclassifications which is much worse than before.

This clearly illustrates that dependent features do not contribute to the inter-class discriminability and may even decrease the performance of the classifier. Thanks to the algorithm introduced in Section 4.4, one can identify the dependent AMIs on theoretical basis before they are applied to particular data and create a powerful independent set of invariants.

### 5.2. Assembling the baby puzzles

This experiment tries to automatically "assemble" a set of baby puzzles. Unlike traditional mosaic puzzles, solving baby puzzles means identifying a piece and putting it into a proper hole in the base board. Each piece can be unambiguously identified by its shape or by the picture on it. The babies usually combine these two cues together but here we discarded the pictorial information at all and worked with the shape only.

The aim of this experiment is to demonstrate how important is to know which AMIs are true invariants and which are pseudoinvariants. When generating the AMIs as described in Section 3, we know exactly the weight $w$ of the invariant. Eq. (8) implies that the AMIs with odd weights are pseudoinvariants.

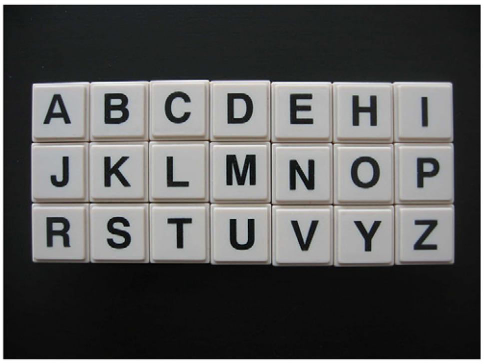



**Fig. 4.** Scrabble tiles—the templates.

**Fig. 5.** Scrabble tiles to be recognized—a sample scene.

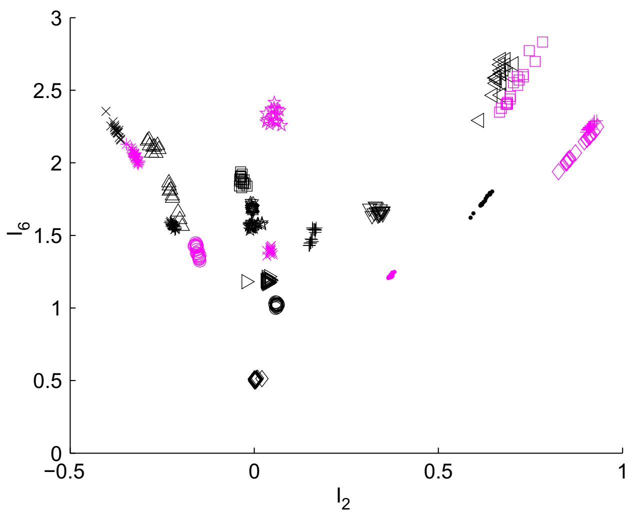



**Fig. 6.** The space of invariants $I_2$ and $I_6$. Even in two dimensions the clustering tendency is evident. Legend: ● A, ◯ B, × C, + D, ∗ E, □ H, ◇ I, ▽ J, △ K, ◁ L, ▷ M, ☆ N, ☆ O, ● P, ◯ R, × S, + T, ∗ U, □ V, ◇ Y, ☆ Z.

[1] We used the tiles from the Scrabble Upwords by Hasbro, see www.hasbro.com for details.

They change the sign if the affine transform contains mirror reflection, i.e. if its Jacobian is negative.

The setting of the experiment is quite simple (see Fig. 7). First, we segmented the holes in the base board. Then we segmented the tiles from the background by region growing, binarized them and tried to find the corresponding hole for each tile. We solved this task twice—by means of true invariants $I_2$, $I_6$, and $I_{47}$ and by pseudoinvariants $I_5$, $I_{10}$, and $I_{64}$. In both cases, we successfully placed all the tiles.

Then we turned all six tiles upside down, photographed them and run the experiment again. The true invariants still recognized the shapes correctly, while the pseudoinvariants were not able to identify any tile. This result is in a good accordance with the theory because turning upside down can be mathematically described by mirroring. Since all shapes in the puzzle are asymmetric, mirroring actually changes the values of all pseudoinvariants, which are no longer invariant and cannot be used for tile recognition. The situation is illustrated by a look at the feature space. While in the space of true invariants $I_2$ and $I_6$, each hole and the corresponding tiles create a compact cluster (see Fig. 8), in the space of pseudoinvariants $I_5$ and $I_{10}$, the tile laid face down is far from the hole and the tile laid face up (see Fig. 9).

However, from this experiment we cannot draw a conclusion that pseudoinvariants should be always excluded from the set of the AMIs. The decision whether or not to use them depends on the particular application. If the assumed shape transforms (i.e. intraclass variability) may contain mirror reflection like in our puzzle experiment, only true invariants and magnitudes of pseudoinvariants should be used. On the other hand, if we consider mirrored shapes as different classes then only pseudoinvariants provide desirable discrimination power.


**Fig. 7.** The base board with six holes and the corresponding puzzle tiles from the baby puzzle experiment.

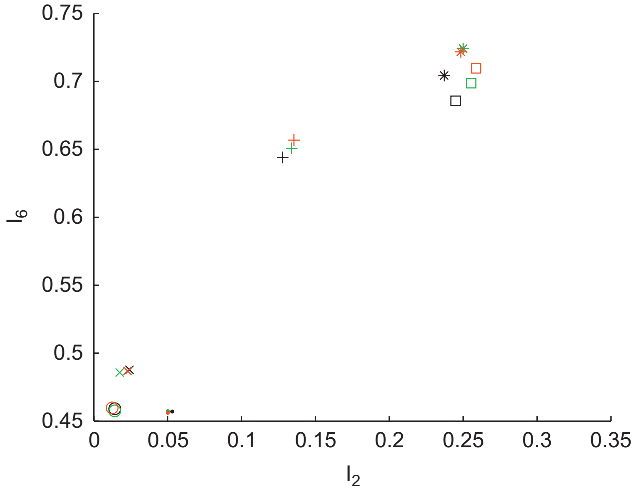



**Fig. 8.** The feature space of true invariants $I_2$ and $I_6$. The black symbols denote the holes in the base board, the green (lighter gray) symbols denote the puzzle tiles and the red (darker gray) symbols denote the tiles turned upside down. Legend: • hedgehog, ○ stork, × hare, + mouse, * mole, □ frog. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

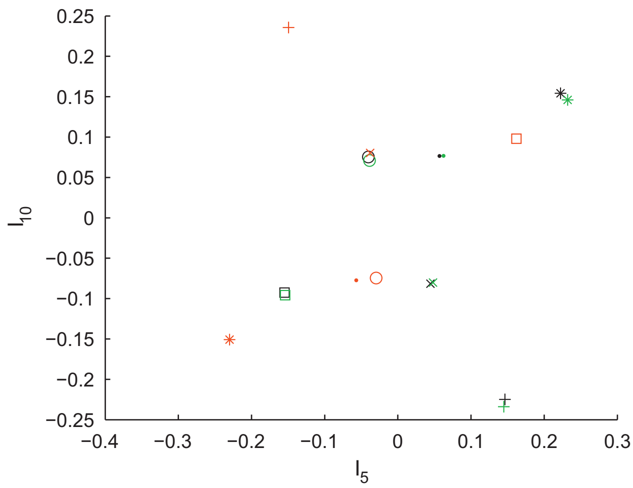



**Fig. 9.** The feature space of pseudoinvariants $I_5$ and $I_{10}$. The black symbols denote the holes in the base board, the green (lighter gray) symbols denote the puzzle tiles and the red (darker gray) symbols denote the tiles turned upside down. Legend: • hedgehog, ○ stork, × hare, + mouse, * mole, □ frog. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

## 6. Conclusion

We presented a general method how to automatically generate the AMI's of any weights and orders. The method is based on representation of the AMI's by graphs. We developed an algorithm for eliminating all reducible invariants and we also discussed how to identify polynomial dependencies among irreducible invariants. The possibility of identifying and eliminating dependent invariants is a major contribution of the paper.

We demonstrated by experiments how important is to choose proper invariants. It should be emphasized that the choice of invariants was discussed here solely from the point of view of their theoretical properties (independence, invariance to mirroring, etc.) which are independent of the data set. In practice, the next step after the theoretical analysis should be adaptation of the invariants to the given training sets. This can be accomplished by standard feature-selection algorithms which either maximize Mahalanobis or another distance measure between the classes or directly optimize the success rate of a particular classifier. However, such selection is always data-dependent and cannot be generalized to other data sets. For instance, if the classes consisted of axi-symmetric objects, the selection algorithm would discard all pseudoinvariants because their recognition ability would be zero, which might be completely incorrect in another case.

## Acknowledgment

This work has been supported by the Grant no. 102/08/1593 of the Czech Science Foundation.

## Appendix

Here we present the irreducible invariants which were used in the experiments in explicit forms along with their weights, structures, and generating graphs

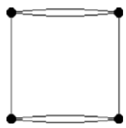
$I_1 = (\mu_{20}\mu_{02} - \mu_{11}^2)/\mu_{00}^4,$

$w = 2, \quad \mathbf{s} = (2),$



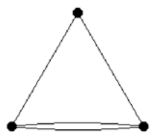
$I_2 = (-\mu_{30}^2\mu_{03}^2 + 6\mu_{30}\mu_{21}\mu_{12}\mu_{03} - 4\mu_{30}\mu_{12}^3 - 4\mu_{21}^3\mu_{03} + 3\mu_{21}^2\mu_{12}^2)/\mu_{00}^{10},$

$w = 6, \quad \mathbf{s} = (0,4),$



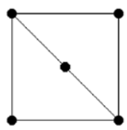
$$I_3 = (\mu_{20}\mu_{21}\mu_{03} - \mu_{20}\mu_{12}^2 - \mu_{11}\mu_{30}\mu_{03} + \mu_{11}\mu_{21}\mu_{12} + \mu_{02}\mu_{30}\mu_{12} - \mu_{02}\mu_{21}^2)/\mu_{00}^7,$$

$w = 4, \quad \mathbf{s} = (1,2),$



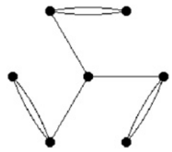
$$I_4 = (-\mu_{20}^3\mu_{03}^2 + 6\mu_{20}^2\mu_{11}\mu_{12}\mu_{03} - 3\mu_{20}^2\mu_{02}\mu_{12}^2 - 6\mu_{20}\mu_{11}^2\mu_{21}\mu_{03} - 6\mu_{20}\mu_{11}^2\mu_{12}^2 + 12\mu_{20}\mu_{11}\mu_{02}\mu_{21}\mu_{12} - 3\mu_{20}\mu_{02}^2\mu_{21}^2 + 2\mu_{11}^3\mu_{30}\mu_{03} + 6\mu_{11}^3\mu_{21}\mu_{12} - 6\mu_{11}^2\mu_{02}\mu_{30}\mu_{12} - 6\mu_{11}^2\mu_{02}\mu_{21}^2 + 6\mu_{11}\mu_{02}^2\mu_{30}\mu_{21} - \mu_{02}^3\mu_{30}^2)/\mu_{00}^{11},$$

$w = 6, \quad \mathbf{s} = (3,2),$



$$I_5 = (\mu_{20}^3\mu_{30}\mu_{03}^3 - 3\mu_{20}^3\mu_{21}\mu_{12}\mu_{03}^2 + 2\mu_{20}^3\mu_{12}^3\mu_{03} - 6\mu_{20}^2\mu_{11}\mu_{30}\mu_{12}\mu_{03}^2 + 6\mu_{20}^2\mu_{11}\mu_{21}^2\mu_{03}^2 + 6\mu_{20}^2\mu_{11}\mu_{21}\mu_{12}^2\mu_{03} - 6\mu_{20}^2\mu_{11}\mu_{12}^4 + 3\mu_{20}^2\mu_{02}\mu_{30}\mu_{12}^2\mu_{03} - 6\mu_{20}^2\mu_{02}\mu_{21}^2\mu_{12}\mu_{03} + 3\mu_{20}^2\mu_{02}\mu_{21}\mu_{12}^3 + 12\mu_{20}\mu_{11}^2\mu_{30}\mu_{12}^2\mu_{03} - 24\mu_{20}\mu_{11}^2\mu_{21}^2\mu_{12}\mu_{03} + 12\mu_{20}\mu_{11}^2\mu_{21}\mu_{12}^3 - 12\mu_{20}\mu_{11}\mu_{02}\mu_{30}\mu_{12}^3 + 12\mu_{20}\mu_{11}\mu_{02}\mu_{21}^3\mu_{03} - 3\mu_{20}\mu_{02}^2\mu_{30}\mu_{21}^2\mu_{03} + 6\mu_{20}\mu_{02}^2\mu_{30}\mu_{21}\mu_{12}^2 - 3\mu_{20}\mu_{02}^2\mu_{21}^3\mu_{12} - 8\mu_{11}^3\mu_{30}\mu_{12}^3 + 8\mu_{11}^3\mu_{21}^3\mu_{03} - 12\mu_{11}^2\mu_{02}\mu_{30}\mu_{21}^2\mu_{03} + 24\mu_{11}^2\mu_{02}\mu_{30}\mu_{21}\mu_{12}^2 - 12\mu_{11}^2\mu_{02}\mu_{21}^3\mu_{12} + 6\mu_{11}\mu_{02}^2\mu_{30}^2\mu_{21}\mu_{03} - 6\mu_{11}\mu_{02}^2\mu_{30}^2\mu_{12}^2 - 6\mu_{11}\mu_{02}^2\mu_{30}\mu_{21}^2\mu_{12} + 6\mu_{11}\mu_{02}^2\mu_{21}^4 - \mu_{02}^3\mu_{30}^3\mu_{03} + 3\mu_{02}^3\mu_{30}^2\mu_{21}\mu_{12} - 2\mu_{02}^3\mu_{30}\mu_{21}^3)/\mu_{00}^{16},$$

$w = 9, \quad \mathbf{s} = (3,4),$

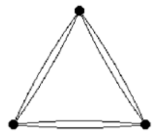


$I_6 = (\mu_{40}\mu_{04} - 4\mu_{31}\mu_{13} + 3\mu_{22}^2)/\mu_{00}^6,$

$w = 4, \quad \mathbf{s} = (0,0,2),$

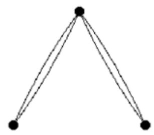
$I_7 = (\mu_{40}\mu_{22}\mu_{04} - \mu_{40}\mu_{13}^2 - \mu_{31}^2\mu_{04} + 2\mu_{31}\mu_{22}\mu_{13} - \mu_{22}^3)/\mu_{00}^9,$

$w = 6, \quad \mathbf{s} = (0,0,3),$



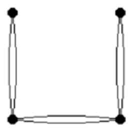
$$I_8 = (\mu_{20}^2\mu_{04} - 4\mu_{20}\mu_{11}\mu_{13} + 2\mu_{20}\mu_{02}\mu_{22} + 4\mu_{11}^2\mu_{22} - 4\mu_{11}\mu_{02}\mu_{31} + \mu_{02}^2\mu_{40})/\mu_{00}^7,$$

$w = 4, \quad \mathbf{s} = (2,0,1),$



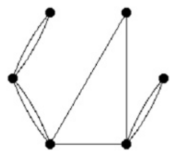
$$I_9 = (\mu_{20}^2\mu_{22}\mu_{04} - \mu_{20}^2\mu_{13}^2 - 2\mu_{20}\mu_{11}\mu_{31}\mu_{04} + 2\mu_{20}\mu_{11}\mu_{22}\mu_{13} + \mu_{20}\mu_{02}\mu_{40}\mu_{04} - 2\mu_{20}\mu_{02}\mu_{31}\mu_{13} + \mu_{20}\mu_{02}\mu_{22}^2 + 4\mu_{11}^2\mu_{31}\mu_{13} - 4\mu_{11}^2\mu_{22}^2 - 2\mu_{11}\mu_{02}\mu_{40}\mu_{13} + 2\mu_{11}\mu_{02}\mu_{31}\mu_{22} + \mu_{02}^2\mu_{40}\mu_{22} - \mu_{02}^2\mu_{31}^2)/\mu_{00}^{10},$$

$w = 6, \quad \mathbf{s} = (2,0,2),$



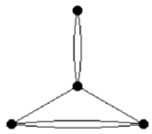
$$I_{10} = (\mu_{20}^3\mu_{31}\mu_{04}^2 - 3\mu_{20}^3\mu_{22}\mu_{13}\mu_{04} + 2\mu_{20}^3\mu_{13}^3 - \mu_{20}^2\mu_{11}\mu_{40}\mu_{04}^2 - 2\mu_{20}^2\mu_{11}\mu_{31}\mu_{13}\mu_{04} + 9\mu_{20}^2\mu_{11}\mu_{22}^2\mu_{04} - 6\mu_{20}^2\mu_{11}\mu_{22}\mu_{13}^2 + \mu_{20}^2\mu_{02}\mu_{40}\mu_{13}\mu_{04} - 3\mu_{20}^2\mu_{02}\mu_{31}\mu_{22}\mu_{04} + 2\mu_{20}^2\mu_{02}\mu_{31}\mu_{13}^2 + 4\mu_{20}\mu_{11}^2\mu_{40}\mu_{13}\mu_{04} - 12\mu_{20}\mu_{11}^2\mu_{31}\mu_{22}\mu_{04} + 8\mu_{20}\mu_{11}^2\mu_{31}\mu_{13}^2 - 6\mu_{20}\mu_{11}\mu_{02}\mu_{40}\mu_{13}^2 + 6\mu_{20}\mu_{11}\mu_{02}\mu_{31}^2\mu_{04} - \mu_{20}\mu_{02}^2\mu_{40}\mu_{31}\mu_{04} + 3\mu_{20}\mu_{02}^2\mu_{40}\mu_{22}\mu_{13} - 2\mu_{20}\mu_{02}^2\mu_{31}^2\mu_{13} - 4\mu_{11}^3\mu_{40}\mu_{13}^2 + 4\mu_{11}^3\mu_{31}^2\mu_{04} - 4\mu_{11}^2\mu_{02}\mu_{40}\mu_{31}\mu_{04} + 12\mu_{11}^2\mu_{02}\mu_{40}\mu_{22}\mu_{13} - 8\mu_{11}^2\mu_{02}\mu_{31}^2\mu_{13} + \mu_{11}\mu_{02}^2\mu_{40}^2\mu_{04} + 2\mu_{11}\mu_{02}^2\mu_{40}\mu_{31}\mu_{13} - 9\mu_{11}\mu_{02}^2\mu_{40}\mu_{22}^2 + 6\mu_{11}\mu_{02}^2\mu_{31}^2\mu_{22} - \mu_{02}^3\mu_{40}^2\mu_{13} + 3\mu_{02}^3\mu_{40}\mu_{31}\mu_{22} - 2\mu_{02}^3\mu_{31}^3)/\mu_{00}^{15},$$

$w = 9, \quad \mathbf{s} = (3,0,3),$



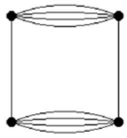
$$I_{19}=(\mu_{20}\mu_{30}\mu_{12}\mu_{04}-\mu_{20}\mu_{30}\mu_{03}\mu_{13}-\mu_{20}\mu_{21}^2\mu_{04}+\mu_{20}\mu_{21}\mu_{12}\mu_{13}+\mu_{20}\mu_{21}\mu_{03}\mu_{22}-\mu_{20}\mu_{12}^2\mu_{22}-2\mu_{11}\mu_{30}\mu_{12}\mu_{13}+2\mu_{11}\mu_{30}\mu_{03}\mu_{22}+2\mu_{11}\mu_{21}^2\mu_{13}-2\mu_{11}\mu_{21}\mu_{12}\mu_{22}-2\mu_{11}\mu_{21}\mu_{03}\mu_{31}+2\mu_{11}\mu_{12}^2\mu_{31}+\mu_{02}\mu_{30}\mu_{12}\mu_{22}-\mu_{02}\mu_{30}\mu_{03}\mu_{31}-\mu_{02}\mu_{21}^2\mu_{22}+\mu_{02}\mu_{21}\mu_{12}\mu_{31}+\mu_{02}\mu_{21}\mu_{03}\mu_{40}-\mu_{02}\mu_{12}^2\mu_{40})/\mu_{00}^{10},$$

$w=6,\quad \mathbf{s}=(1,2,1),$



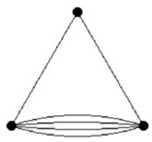
$$I_{47}=(-\mu_{50}^2\mu_{05}^2+10\mu_{50}\mu_{41}\mu_{14}\mu_{05}-4\mu_{50}\mu_{32}\mu_{23}\mu_{05}-16\mu_{50}\mu_{32}\mu_{14}^2+12\mu_{50}\mu_{23}^2\mu_{14}-16\mu_{41}^2\mu_{23}\mu_{05}-9\mu_{41}^2\mu_{14}^2+12\mu_{41}\mu_{32}^2\mu_{05}+76\mu_{41}\mu_{32}\mu_{23}\mu_{14}-48\mu_{41}\mu_{23}^3-48\mu_{32}^3\mu_{14}+32\mu_{32}^2\mu_{23}^2)/\mu_{00}^{14},$$

$w=10,\quad \mathbf{s}=(0,0,0,4),$



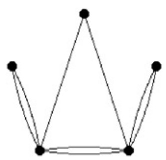
$$I_{48}=(\mu_{20}\mu_{41}\mu_{05}-4\mu_{20}\mu_{32}\mu_{14}+3\mu_{20}\mu_{23}^2-\mu_{11}\mu_{50}\mu_{05}+3\mu_{11}\mu_{41}\mu_{14}-2\mu_{11}\mu_{32}\mu_{23}+\mu_{02}\mu_{50}\mu_{14}-4\mu_{02}\mu_{41}\mu_{23}+3\mu_{02}\mu_{32}^2)/\mu_{00}^9,$$

$w=6,\quad \mathbf{s}=(1,0,0,2),$



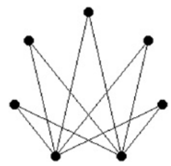
$$I_{49}=(\mu_{20}^3\mu_{23}\mu_{05}-\mu_{20}^3\mu_{14}^2-3\mu_{20}^2\mu_{11}\mu_{32}\mu_{05}+3\mu_{20}^2\mu_{11}\mu_{23}\mu_{14}+\mu_{20}^2\mu_{02}\mu_{41}\mu_{05}-\mu_{20}^2\mu_{02}\mu_{32}\mu_{14}+2\mu_{20}\mu_{11}^2\mu_{41}\mu_{05}+4\mu_{20}\mu_{11}^2\mu_{32}\mu_{14}-6\mu_{20}\mu_{11}^2\mu_{23}^2-\mu_{20}\mu_{11}\mu_{02}\mu_{50}\mu_{05}-3\mu_{20}\mu_{11}\mu_{02}\mu_{41}\mu_{14}+4\mu_{20}\mu_{11}\mu_{02}\mu_{32}\mu_{23}+\mu_{20}\mu_{02}^2\mu_{50}\mu_{14}-\mu_{20}\mu_{02}^2\mu_{41}\mu_{23}-4\mu_{11}^3\mu_{41}\mu_{14}+4\mu_{11}^3\mu_{32}\mu_{23}+2\mu_{11}^2\mu_{02}\mu_{50}\mu_{14}+4\mu_{11}^2\mu_{02}\mu_{41}\mu_{23}-6\mu_{11}^2\mu_{02}\mu_{32}^2-3\mu_{11}\mu_{02}^2\mu_{50}\mu_{23}+3\mu_{11}\mu_{02}^2\mu_{41}\mu_{32}+\mu_{02}^3\mu_{50}\mu_{32}-\mu_{02}^3\mu_{41}^2)/\mu_{00}^{13},$$

$w=8,\quad \mathbf{s}=(3,0,0,2),$



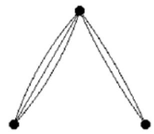
$$I_{50}=(-\mu_{20}^5\mu_{05}^2+10\mu_{20}^4\mu_{11}\mu_{14}\mu_{05}-5\mu_{20}^4\mu_{02}\mu_{14}^2-20\mu_{20}^3\mu_{11}^2\mu_{23}\mu_{05}-20\mu_{20}^3\mu_{11}^2\mu_{14}^2+40\mu_{20}^3\mu_{11}\mu_{02}\mu_{23}\mu_{14}-10\mu_{20}^3\mu_{02}^2\mu_{23}^2+20\mu_{20}^2\mu_{11}^3\mu_{32}\mu_{05}+60\mu_{20}^2\mu_{11}^3\mu_{23}\mu_{14}-60\mu_{20}^2\mu_{11}^2\mu_{02}\mu_{32}\mu_{14}-60\mu_{20}^2\mu_{11}^2\mu_{02}\mu_{23}^2+60\mu_{20}^2\mu_{11}\mu_{02}^2\mu_{32}\mu_{23}-10\mu_{20}^2\mu_{02}^3\mu_{32}^2-10\mu_{20}\mu_{11}^4\mu_{41}\mu_{05}-40\mu_{20}\mu_{11}^4\mu_{32}\mu_{14}-30\mu_{20}\mu_{11}^4\mu_{23}^2+40\mu_{20}\mu_{11}^3\mu_{02}\mu_{41}\mu_{14}+120\mu_{20}\mu_{11}^3\mu_{02}\mu_{32}\mu_{23}-60\mu_{20}\mu_{11}^2\mu_{02}^2\mu_{41}\mu_{23}-60\mu_{20}\mu_{11}^2\mu_{02}^2\mu_{32}^2+40\mu_{20}\mu_{11}\mu_{02}^3\mu_{41}\mu_{32}-5\mu_{20}\mu_{02}^4\mu_{41}^2+2\mu_{11}^5\mu_{50}\mu_{05}+10\mu_{11}^5\mu_{41}\mu_{14}+20\mu_{11}^5\mu_{32}\mu_{23}-10\mu_{11}^4\mu_{02}\mu_{50}\mu_{14}-40\mu_{11}^4\mu_{02}\mu_{41}\mu_{23}-30\mu_{11}^4\mu_{02}\mu_{32}^2+20\mu_{11}^3\mu_{02}^2\mu_{50}\mu_{23}+60\mu_{11}^3\mu_{02}^2\mu_{41}\mu_{32}-20\mu_{11}^2\mu_{02}^3\mu_{50}\mu_{32}-20\mu_{11}^2\mu_{02}^3\mu_{41}^2+10\mu_{11}\mu_{02}^4\mu_{50}\mu_{41}-\mu_{02}^5\mu_{50}^2)/\mu_{00}^{17},$$

$w=10,\quad \mathbf{s}=(5,0,0,2),$



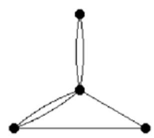
$$I_{64}=(-\mu_{20}\mu_{30}\mu_{05}+3\mu_{20}\mu_{21}\mu_{14}-3\mu_{20}\mu_{12}\mu_{23}+\mu_{20}\mu_{03}\mu_{32}+2\mu_{11}\mu_{30}\mu_{14}-6\mu_{11}\mu_{21}\mu_{23}+6\mu_{11}\mu_{12}\mu_{32}-2\mu_{11}\mu_{03}\mu_{41}-\mu_{02}\mu_{30}\mu_{23}+3\mu_{02}\mu_{21}\mu_{32}-3\mu_{02}\mu_{12}\mu_{41}+\mu_{02}\mu_{03}\mu_{50})/\mu_{00}^8,$$

$w=5,\quad \mathbf{s}=(1,1,0,1),$



$$I_{65}=(\mu_{20}^2\mu_{21}\mu_{05}-2\mu_{20}^2\mu_{12}\mu_{14}+\mu_{20}^2\mu_{03}\mu_{23}-\mu_{20}\mu_{11}\mu_{30}\mu_{05}-\mu_{20}\mu_{11}\mu_{21}\mu_{14}+5\mu_{20}\mu_{11}\mu_{12}\mu_{23}-3\mu_{20}\mu_{11}\mu_{03}\mu_{32}+\mu_{20}\mu_{02}\mu_{30}\mu_{14}-\mu_{20}\mu_{02}\mu_{21}\mu_{23}-\mu_{20}\mu_{02}\mu_{12}\mu_{32}+\mu_{20}\mu_{02}\mu_{03}\mu_{41}+2\mu_{11}^2\mu_{30}\mu_{14}-2\mu_{11}^2\mu_{21}\mu_{23}-2\mu_{11}^2\mu_{12}\mu_{32}+2\mu_{11}^2\mu_{03}\mu_{41}-3\mu_{11}\mu_{02}\mu_{30}\mu_{23}+5\mu_{11}\mu_{02}\mu_{21}\mu_{32}-\mu_{11}\mu_{02}\mu_{12}\mu_{41}-\mu_{11}\mu_{02}\mu_{03}\mu_{50}+\mu_{02}^2\mu_{30}\mu_{32}-2\mu_{02}^2\mu_{21}\mu_{41}+\mu_{02}^2\mu_{12}\mu_{50})/\mu_{00}^{10},$$

$w=6,\quad \mathbf{s}=(2,1,0,1).$



## References


[1] E.P.L. Van Gool, T. Moons, A. Oosterlinck, Vision and Lie's approach to invariance, Image and Vision Computing 13 (4) (1995) 259–277.
[2] T. Suk, J. Flusser, Projective moment invariants, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (10) (2004) 1364–1367.
[3] M.-K. Hu, Visual pattern recognition by moment invariants, IRE Transactions on Information Theory 8 (2) (1962) 179–187.
[4] T.H. Reiss, The revised fundamental theorem of moment invariants, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (8) (1991) 830–834.
[5] J. Flusser, T. Suk, Pattern recognition by affine moment invariants, Pattern Recognition 26 (1) (1993) 167–174.
[6] J. Flusser, T. Suk, Pattern recognition by means of affine moment invariants, Research Report 1726, Institute of Information Theory and Automation, 1991.
[7] J.J. Sylvester, Assisted by F. Franklin, Tables of the generating functions and groundforms for the binary quantics of the first ten orders, American Journal of Mathematics 2 (1879) 223–251.
[8] J.J. Sylvester, Assisted by F. Franklin, Tables of the generating functions and groundforms for simultaneous binary quantics of the first four orders taken two and two together, American Journal of Mathematics 2 (1879) 293–306, 324–329.
[9] I. Schur, Vorlesungen über Invariantentheorie, Springer, Berlin, 1968.
[10] G.B. Gurevich, Foundations of the Theory of Algebraic Invariants, Nordhoff, Groningen, The Netherlands, 1964.
[11] D. Hilbert, Theory of Algebraic Invariants, Cambridge University Press, Cambridge, 1993.
[12] A.G. Mamistvalov, *n*-Dimensional moment invariants and conceptual mathematical theory of recognition *n*-dimensional solids, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 819–831.

[13] J. Flusser, T. Suk, B. Zitová, Moments and Moment Invariants in Pattern Recognition, Wiley, Chichester, 2009.
[14] T. Suk, J. Flusser, Graph method for generating affine moment invariants, in: Proceedings of the 17th International Conference on Pattern Recognition ICPR'04, IEEE Computer Society, Cambridge, UK, 2004, pp. 192–195.
[15] T.H. Reiss, Recognizing Planar Objects Using Invariant Image Features, in: Lecture Notes in Computer Science, vol. 676, Springer, Berlin, 1993.
[16] D. Xu, H. Li, Geometric moment invariants, Pattern Recognition 41 (1) (2008) 240–249.
[17] Å. Wallin, O. Kübler, Complete sets of complex Zernike moment invariants and the role of the pseudoinvariants, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (11) (1995) 1106–1110.
[18] T. Suk, J. Flusser, Tables of affine moment invariants generated by the graph method, Research Report 2156, Institute of Information Theory and Automation, 2005.
[19] J. Flusser, T. Suk, B. Zitová ⟨http://zoi.utia.cas.cz/moment_invariants⟩, the accompanying web pages of the book: Moments and Moment Invariants in Pattern Recognition, Wiley, Chichester, 2009.

**Tomás Suk** received the M.Sc. degree in Electrical Engineering from the Czech Technical University, Faculty of Electrical Engineering, Prague, 1987. The C.Sc. degree (corresponds to Ph.D.) in Computer Science from the Czechoslovak Academy of Sciences, Prague, 1992. From 1991 he is 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 published 14 papers in international journals and 31 conference papers, mostly from international conferences. In 2002 he received the Otto Wichterle's premium of the Academy of Sciences of the Czech Republic for young scientists. His research interests includes digital image processing, pattern recognition, image filtering, invariant features, moment and point invariants, geometric transformations of images. Applications in remote sensing, astronomy, medicine and computer vision.

**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. Since 1995, he has been holding the position of a head of Department of Image Processing and since 2007, he is a director of the Institute. Since 1991, he has also been affiliated with Charles University, Prague, and the Czech Technical University, Prague, where he teaches courses on digital image processing and pattern recognition. He has been a Full Professor since 2004. His current research interests include all aspects of digital image processing and pattern recognition, namely 2-D object recognition, moment invariants, blind deconvolution, image registration, and image fusion, and has authored and coauthored more than 200 research publications in these areas.