# SYMMETRIC POLYNOMIALS FOR 2D SHAPE REPRESENTATION

*Renato M. P. Negrinho,   Pedro M. Q. Aguiar*

Institute for Systems and Robotics / Instituto Superior Técnico, Lisboa, Portugal
renato.negrinho@ist.utl.pt, aguiar@isr.ist.utl.pt

## ABSTRACT

We discuss the usage of symmetric polynomials for representing 2D shapes in their most general form, *i.e.*, arbitrary sets of unlabeled points in the plane. Although particular families of these polynomials have been used in the past, we present general results that pave the way for the development of new representations that exhibit key properties for shape recognition. We show that all monomial symmetric polynomials satisfy homogeneity, which enables leveraging on previous work on spectral invariants to obtain invariance/completeness with respect to shape orientation. Representations based on symmetric polynomials are invariant to shape point relabeling. We single out elementary symmetric polynomials and power sums as particular families of polynomials that further enable obtaining completeness with respect to point labeling. We discuss the efficient computation of these polynomials and study how perturbations in the shape point coordinates affect their values.

***Index Terms***— Symmetric polynomials, Monomial symmetric polynomials, Elementary symmetric polynomials, Power sums, Homogeneity, Invariance, Completeness, Shape representation

## 1. INTRODUCTION

Several image analysis tasks, ranging from classification, recognition, and retrieval, to denoising and coding, require dealing with the shape of image regions. The representation of two-dimensional (2D) shape has thus deserved attention in the past decades (surveys of proposed methods can be found in [1, 2, 3]). The majority of approaches consider shapes that are well-described by closed contours, which are easy to represent, *e.g.*, [4, 5, 6]. However, in many real-life scenarios, the underlying shapes contain multiple contours, lines, and/or small isolated regions that are better modeled as points. For us, a 2D shape is thus an arbitrary set of points in the plane.

The representation of shapes whose points are labeled only requires factoring out rigid geometric transformations and dealing with point coordinate disturbances [7, 8]. However, often shape points come without labels or natural ordering, *e.g.*, when arising from automatic edge/corner/interest-point detection. In these cases, simple normalization takes care of global position and scale, but normalizing a point set with respect to rotation is harder than it could seem at first sight (degenerate cases and sensitivity to the noise motivates ongoing research, *e.g.*, [9, 10]). A distinct approach is to simultaneously estimate the global rotation and point label permutation relating shapes to compare. This is a non-convex problem that imposes the usual limitations on iterative algorithms like ICP [11] or EM [12, 13] (uncertain convergence, sensitivity to the initialization).

We consider using symmetric polynomials to represent 2D shape, which is immediately motivated by their invariance to shape

point relabeling. Focusing on particular families of symmetric polynomials, we address completeness, which guarantees that the representation preserves all shape information. In what respects to dealing with unknown shape orientation, we show how the usage of spectral invariants enables extending to this case the properties of the representation. The end result is a theoretically sound way to represent arbitrary shapes as simple points in a shape space, where standard algorithms (for classification, filtering, etc) can be used. We conclude the paper with the illustration of a task that uses this space: the computation of the mean shape.

## 2. SYMMETRIC POLYNOMIALS OF 2D SHAPE POINTS

A polynomial $s : \mathbb{C}^N \to \mathbb{C}$ is a *symmetric polynomial* if it is invariant to all permutations of its variables. Some examples of symmetric polynomials in $N$ complex variables, $z_1, \ldots, z_N \in \mathbb{C}$, are:

$$p_1(z_1, \ldots, z_N) = z_1 + \cdots + z_N, \quad (1)$$

$$p_3(z_1, \ldots, z_N) = z_1^3 + \cdots + z_N^3, \quad (2)$$

$$s(z_1, \ldots, z_N) = z_1 + z_1^3 + \cdots + z_N + z_N^3, \quad (3)$$

$$e_2(\cdot) = z_1 z_2 + \cdots + z_1 z_N + z_2 z_3 + \cdots + z_2 z_N + \cdots + z_{N-1} z_N. \quad (4)$$

It is immediate that polynomials (1), (2), and (3) are symmetric. Polynomial (4) is also symmetric, since it is given by the sum of all products of pairs of its variables (a permutation of these variables amounts to reordering the terms in the sum, which does not change the polynomial). It can also be easily seen that summing or multiplying two arbitrary symmetric polynomials yields a symmetric polynomial. (In fact, symmetric polynomials form a ring [14].)

The invariance just stated is the most immediate characteristic of symmetric polynomials that motivates their usage for shape representation. In fact, representing a 2D shape, *i.e.*, an arbitrary set of points in the plane, $z_1, \ldots, z_N \in \mathbb{C}$, via a set of symmetric polynomials evaluated at $z_1, \ldots, z_N$, is appealing, since the representation results insensitive to the point labels, which are unavailable in many practical scenarios. In this paper, we study other properties of this kind of representation that are key to shape recognition, in particular, its completeness and the way it behaves with a rigid geometric transformation of the shape (since shape position and scale are easily factored out through normalization, what remains is a rotation).

Each of the terms $z_1^{d_1} \cdots z_N^{d_N}$, with $d_1, \ldots, d_N \in \mathbb{N} \cup \{0\}$, in a polynomial, is a monomial, which we will denote by $(d_1, \ldots, d_N)$. For example, in the variables $z_1, z_2, z_3, z_4$, the monomial $z_1 z_3^2$ is denoted by $(1, 0, 2, 0)$. The order of a monomial is given by the sum of the degrees of its variables. For example, polynomials (1), (2), and (4) have monomials of order 1, 3 and 2, respectively.

## 3. MONOMIAL SYMMETRIC POLYNOMIALS

If a symmetric polynomial has a monomial $(d_1, \ldots, d_N)$, it also has to have all the monomials that can be generated from $(d_1, \ldots, d_N)$ by a permutation, *i.e.*, all the monomials $(d_{\pi(1)}, \ldots, d_{\pi(N)})$, where $\pi$ denotes an arbitrary element of the symmetric group $S_N$ [15]. The so-called *monomial symmetric polynomials* are generated this way.

To construct a monomial symmetric polynomials, consider a (so-called indexing) monomial $(d_1, \ldots, d_N)$, with $d_1 \geq \ldots \geq d_N$. Summing all the monomials that can be obtained by arbitrary permutations of the variables of the indexing monomial $(d_1, \ldots, d_N)$, we obtain the indexed monomial symmetric polynomial:

$$s_{(d_1, \ldots, d_N)}(z_1, \ldots, z_N) = \frac{1}{|G|} \sum_{\pi \in S_N} z_{\pi(1)}^{d_1} \cdots z_{\pi(N)}^{d_N}, \quad (5)$$

where $|G|$ denotes the number of permutations that do not change the indexing monomial, acting as a normalization. The non-increasing order of $d_1, \ldots, d_N$ guarantees the uniqueness of the indexing, *i.e.*, that we do not have two different monomials originating the same symmetric polynomial. Naturally, a monomial that is not given in non-increasing order can be appropriately sorted without changing the resultant monomial symmetric polynomial.

Any permutation of the variables $z_1, \ldots, z_N$ in (5) amounts to reordering the monomials in the sum, leaving the polynomial unchanged. As an example of the construction process, the indexing monomial $(3, 2, 1)$, for which $|G| = 1$, yields the polynomial

$$s_{(3,2,1)}(\cdot) = z_1^3 z_2^2 z_3 + z_1^3 z_2 z_3^2 + z_1^2 z_2^3 z_3 + z_1^2 z_2 z_3^3 + z_1 z_2^3 z_3^2 + z_1 z_2^2 z_3^3.$$

Polynomials (1), (2), and (4), are also examples of monomial symmetric polynomials. Polynomial (3) is not a monomial symmetric polynomial, being the sum of (1) and (2). (In fact, any symmetric polynomial can be written as a linear combination of monomial symmetric polynomials [14].)

## 4. HOMOGENEITY AND SHAPE ROTATION

Any monomial symmetric polynomial satisfies the homogeneity property, which states that when all the variables are multiplied by a constant, the resulting polynomial is the original one multiplied by the same constant raised to the order of the polynomial. This is shown by considering $w_n = \alpha z_n, n = 1, \ldots, N$, with $\alpha \in \mathbb{C}$; then,

$$\begin{aligned} s_{(d_1, \ldots, d_N)}(w_1, \ldots, w_N) &= \frac{1}{|G|} \sum_{\pi \in S_N} w_{\pi(1)}^{d_1} \cdots w_{\pi(N)}^{d_N} \\ &= \frac{1}{|G|} \sum_{\pi \in S_N} (\alpha z_{\pi(1)})^{d_1} \cdots (\alpha z_{\pi(N)})^{d_N} \\ &= \alpha^{\sum_{n=1}^{N} d_n} \frac{1}{|G|} \sum_{\pi \in S_N} z_{\pi(1)}^{d_1} \cdots z_{\pi(N)}^{d_N} \\ &= \alpha^{\sum d_n} s_{(d_1, \ldots, d_N)}(z_1, \ldots, z_N), \quad (6) \end{aligned}$$

where we just used definition (5) and simple manipulations.

Of interest in the context of shape representation are constants of the type $\alpha = e^{j\theta}$, since multiplying by such a factor corresponds to a counter-clockwise rotation by an angle of $\theta$ radians. Considering two differently oriented versions of the same shape, $z_1, \ldots, z_N$ and $w_1, \ldots, w_N$, with $w_n = z_n e^{j\theta}$, the corresponding monomial symmetric polynomials are related by the homogeneity property (6) for this particular case:

$$s_{(d_1, \ldots, d_N)}(w_1, \ldots, w_N) = e^{j\theta \sum d_n} s_{(d_1, \ldots, d_N)}(z_1, \ldots, z_N). \quad (7)$$

The homogeneity property enables using spectral invariants when representing shape via symmetric polynomials. In fact, expression (7) is analogous to the one relating the coefficients of the Fourier series of a signal with the ones of a shifted version of it. Spectral invariants to signal shifts have been extensively studied in the past. In particular, it has been shown that the bispectrum is not only shift-invariant, but also complete, in the sense of determining the underlying signal (up to a shift), under mild conditions [16, 17, 18]. Thus, a bispectrum computed from monomial symmetric polynomials inherits the invariance/completeness, now in what respects to shape rotation. A shape representation using this idea with a particular family of polynomials was proposed in [19], where experimental results illustrate its applicability in practical scenarios. Many other papers report experiments testifying the good behavior of the bispectrum in recognition applications, *e.g.*, [20, 21, 22]. We now see that, according to (7), the same type of spectral invariants can be built from arbitrary sets of monomial symmetric polynomials, such as the ones we focus in the sequel.

## 5. POWER SUMS

*Power sums* are the most widely known symmetric polynomials. The power sum of order $k > 0$ is the polynomial $p_k : \mathbb{C}^N \to \mathbb{C}$ that sums the $k$-th powers of all its variables, $z_1, \ldots, z_N$:

$$p_k(z_1, \ldots, z_N) = \sum_{n=1}^{N} z_n^k. \quad (8)$$

Examples of symmetric polynomials (1) and (2) are power sums of orders 1 and 3, respectively.

It is also easy to see that, in particular, the power sum of order $k$ is a monomial symmetric polynomial, with the indexing monomial given by $(k, 0, \ldots, 0)$, *i.e.*,

$$p_k(z_1, \ldots, z_N) = s_{(k,0,\ldots,0)}(z_1, \cdots, z_N).$$

For $k = 0$, we have $p_0(z_1, \ldots, z_N) = s_{(0,\ldots,0)}(z_1, \ldots, z_N) = 1$.

## 6. ELEMENTARY SYMMETRIC POLYNOMIALS

The *elementary symmetric polynomial* $e_k : \mathbb{C}^N \to \mathbb{C}$ of order $k \geq 0$ is defined as the sum of all different products of $k$ variables, *i.e.*, the sum of all different monomials of order $k$ involving each of the variables $z_1, \ldots, z_N$ zero or one times. Therefore, this sum is composed by $N$-choose-$k$ monomials, which can be formalized as

$$e_k(z_1, \ldots, z_N) = \sum_{(i_1, \ldots, i_N) \in I_k^N} z_1^{i_1} \ldots z_N^{i_N}, \quad (9)$$

where $I_k^N$ is the set of tuples $(i_1, \ldots, i_N)$ verifying $\sum_{n=1}^{N} i_n = k$ and $i_1, \ldots, i_N \in \{0, 1\}$. If $k = 0$, $I_k^N$ is a singleton, yielding $e_0(\cdot) = 1$; if $k > N$, $I_k^N$ is empty, thus $e_k(\cdot) = 0$. Examples of elementary symmetric polynomials of orders 1 and 2 are, respectively, (1), which is also a power sum, and (4).

As it happens for the power sums, the elementary symmetric polynomials are also monomial symmetric polynomials – the indexing monomial for $e_k$ has the first $k$ entries equal to one and the remaining $N - k$ equal to zero:

$$e_k(z_1, \ldots, z_N) = s_{(1,\ldots,1,0,\ldots 0)}(z_1, \ldots, z_N).$$

## 7. COMPLETENESS AND SHAPE REPRESENTATION

Remarkably, the polynomials introduced in Sections 5 and 6 are not only invariant to permutations of the variables (as any symmetric polynomial) but also complete, in the sense that both families unambiguously determine the values of those variables. The completeness is easily proved for the family of elementary symmetric polynomials by noting that these polynomials are straightforwardly related to the coefficients of the monic polynomial with roots $z_1, \ldots, z_N$, see, e.g., [23]:

$$\prod_{n=1}^{N} (t - z_n) = \sum_{k=0}^{N} (-1)^k e_k(z_1, \ldots, z_N) t^{N-k}. \quad (10)$$

Since the coefficients of a polynomial uniquely identify its roots, the values of the elementary symmetric polynomials $e_k(z_1, \ldots, z_N)$, with $k = 1, \ldots, N$, uniquely determine the variables $z_1, \ldots, z_N$, up to a permutation.

The completeness for the family of power sums is obtained by using Newton's identities to relate them to the elementary symmetric polynomials, see, e.g., [14]:

$$k e_k(\cdot) + \sum_{r=1}^{k} (-1)^r e_{k-r}(\cdot) p_r(\cdot) = 0.$$

This expression can be easily rearranged in a way that conveys the recursive computation of $p_k(z_1, \ldots, z_N)$, $k = 1, \ldots, N$, from $e_k(z_1, \ldots, z_N)$, $k = 1, \ldots, N$ (or vice versa), proving that the power sums and the elementary symmetric polynomials are equivalent in what respects to completeness.

The completeness just proved is key for shape representation. In fact, it guarantees that either the family of power sums or the one of elementary symmetric polynomials, evaluated at the 2D shape points, suffice to determine the point coordinates. Thus, besides being invariant to permutations of the point labels, both families preserve all shape information, as obviously desired for successful recognition. Fig. 1 illustrates the invariance and completeness of the elementary symmetric polynomials with a simple shape.
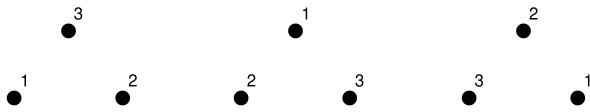


**Fig. 1**. Three shapes with points of coordinates $-1$, $1$, $j$, differing by permutations of the labels. They all yield the same values for the elementary symmetric polynomials: $e_1(-1, 1, j) = e_1(j, -1, 1) = e_1(1, j, -1) = j$, $e_2(-1, 1, j) = e_2(j, -1, 1) = e_2(1, j, -1) = -1$, and $e_3(-1, 1, j) = e_3(j, -1, 1) = e_3(1, j, -1) = -j$. Furthermore, $\{-1, 1, j\}$ is the only set of three points yielding this result.

## 8. EFFICIENT COMPUTATION

An apparent problem concerning the usage of the monomial symmetric polynomials in practice is their computation, since a direct implementation of definition (5) may be computationally intractable, even for moderate values of $N$ and $k$. To gain insight, note that evaluating the elementary symmetric polynomial $e_k(z_1, \ldots, z_N)$, for $N = 50$ and $k = 10$, according to definition (9), requires computing and summing more than 10 billion monomials. Reference [24]

develops a dynamic programming approach by exploiting the fact that the evaluation of any monomial symmetric polynomial can be decomposed in overlapping subproblems. We now summarize this recursive computation for the polynomials of Sections 5 and 6.

For the power sums, the recursive computation, for $k > 0$, is:

$$p_k(z_1, \ldots, z_N) = p_k(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N) + z_n^k, \quad (11)$$

where $n$ is arbitrary, between 1 and $N$. This expression, which is actually immediate from (8), does not lead to computational gain, since each polynomial evaluated along the recursion is used only once, i.e., there is no gain in storing its value for later usage.

The case is distinct for the elementary symmetric polynomials, for which a similar decomposition yields an enormous computational gain. For $k > 0$, it is shown in [24] that

$$e_k(z_1, \ldots, z_N) = e_k(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N)$$
$$+ z_n e_{k-1}(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N). \quad (12)$$

where, again, $n$ is arbitrary. The computational gain comes from storing the values of the polynomials evaluated along recursion (12), which are used multiple times. To clarify this point, we consider $n = N$ and draw the attention of the reader to the array depicted in Fig. 2, where entry $(i, j)$ stores the value of $e_j(z_1, \ldots, z_i)$. The computation of $e_j(z_1, \ldots, z_i)$ according to (12) is performed in constant time after entries $(i - 1, j - 1)$ and $(i - 1, j)$ have been previously computed (see the arrows indicating this dependency in Fig. 2). Thus, decomposition (12) is recursively used until a position whose content is known is reached, which eventually happens, since $e_0(z_1, \ldots, z_N) = 1$, for all $N$, and $e_k(z_1, \ldots, z_N) = 0$, for $k > N$ (we did not even include in the array these known entries, i.e., the column $j = 0$ and entries $(i, j)$ with $j > i$). It is simple to conclude that the evaluation of $e_k(z_1, \ldots, z_N)$ requires computing $k(N - k + 1)$ array entries (grey cells in the example of Fig. 2). Using again the example of the beginning of the section, $N = 50, k = 10$, for which definition (9) requires computing more than 10 billion terms, the recursion just described uses only 410 array entries. When evaluating the entire set of polynomials $e_k(z_1, \ldots, z_N)$, for $k = 1, \ldots, N$, an additional computational gain comes from reusing many times previously computed entries. In this case, the total number of array entries involved is $N(N + 1)/2$.
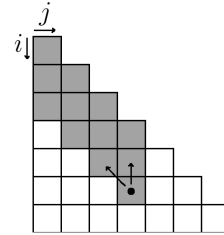


**Fig. 2**. Array storing the values of elementary symmetric polynomials for computational efficiency. Entry $(i, j)$ stores $e_j(z_1, \ldots, z_i)$. The computation of $e_4(z_1, \ldots, z_6)$ via decomposition (12) is represented by arrows relating to $e_3(z_1, \ldots, z_5)$ and $e_4(z_1, \ldots, z_5)$. Starting from an empty array, the evaluation of $e_4(z_1, \ldots, z_6)$ through recursive usage of (12) only requires filling the 12 gray cells.

## 9. PERTURBATION ANALYSIS

Polynomials are sums of products, which makes nontrivial the task of deriving how noise propagates. We study how the values of power

sums and elementary symmetric polynomials change with perturbations of their arguments. To derive the first-order approximation of the perturbed polynomials, let us denote the (assumed small) perturbation affecting $z_n$ as $\Delta z_n \in \mathbb{C}$ and the perturbed version of $z_n$ as $z'_n = z_n + \Delta z_n$, with $n = 1, \ldots, N$.

For the power sums, the partial derivative of $p_k$ with respect to $z_n$ is easily obtained from (11), being the gradient of $p_k$ given by

$$\nabla p_k(z_1, \ldots, z_N) = \begin{bmatrix} kz_1^{k-1} \\ \vdots \\ kz_N^{k-1} \end{bmatrix}.$$

The first-order approximation of the perturbed power sum is

$$\begin{aligned} p_k(z'_1, \ldots, z'_N) &\approx p_k(z_1, \ldots, z_N) + \nabla p_k(z_1, \ldots, z_N)^T \Delta z \\ &\approx p_k(z_1, \ldots, z_N) + kz_1^{k-1}\Delta z_1 \\ &\quad + \cdots \\ &\quad + kz_N^{k-1}\Delta z_N, \end{aligned}$$

where $\Delta z = [\Delta z_1, \ldots, \Delta z_N]^T$. We can see that a perturbation $\Delta z_n$ induces a perturbation on the power sum $p_k(z_1, \ldots, z_N)$ that does not depend on any variable other than the perturbed one, $z_n$.

For the elementary symmetric polynomials, decomposition (12) enables obtaining the partial derivative of $e_k$ with respect to $z_n$:

$$\frac{\partial e_k}{\partial z_n}(z_1, \ldots, z_N) = e_{k-1}(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N).$$

Consequently, the gradient of $e_k$ is given by

$$\nabla e_k(z_1, \ldots, z_N) = \begin{bmatrix} e_{k-1}(z_2, \ldots, z_N) \\ e_{k-1}(z_1, z_3, \ldots, z_N) \\ \vdots \\ e_{k-1}(z_1, \ldots, z_{N-2}, z_N) \\ e_{k-1}(z_1, \ldots, z_{N-1}) \end{bmatrix}.$$

and the first-order approximation of the perturbed polynomial is

$$\begin{aligned} e_k(z'_1, \ldots, z'_N) &\approx e_k(z_1, \ldots, z_N) + \nabla e_k(z_1, \ldots, z_N)^T \Delta z \\ &\approx e_k(z_1, \ldots, z_N) + e_{k-1}(z_2, \ldots, z_N)\Delta z_1 \\ &\quad + \ldots \\ &\quad + e_{k-1}(z_1, \ldots, z_{N-1})\Delta z_N. \end{aligned}$$

It is curious that the perturbation on the elementary symmetric polynomial $e_k(z_1, \ldots, z_N)$ induced by a perturbation on variable $z_n$ does not depend on the perturbed variable (it depends on the polynomial $e_{k-1}(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N)$ and, naturally, on $\Delta z_n$).

## 10. THE MEAN SHAPE

The comparison of shapes that are not aligned (due to distinct pose and point labeling) is sometimes performed in the space of shape point coordinates, after first aligning the shapes. However, not only the alignment step is nontrivial, but also modern machine learning algorithms require more than shape differences; they require each shape to be represented by a simple point in a shape space, where statistics can be easily computed. We now illustrate how the representations discussed in the paper enable doing this.

Consider the problem of computing the mean of a set of shapes, *i.e.*, a set of sets of unlabeled points. Now, represent each shape by the elementary symmetric polynomials evaluated at the shape

points, as discussed above. Since the representation is invariant to point relabeling, it makes sense to compute the mean representation $\bar{e}_k$. Since the representation is also complete, it makes sense to invert $\bar{e}_k$, *i.e.*, to determine the mean shape, the one whose representation is $\bar{e}_k$. This inversion can be performed by computing the roots of a polynomial whose coefficients are easily related to $\bar{e}_k$, see (10).

An illustrative example of the procedure just summarized is conveyed by Figs. 3 and 4. We emphasize that the mean shape in Fig. 3 is computed in a way that avoids any kind of geometric grouping/clustering of shape points; it rather corresponds to a standard average in the shape space, as represented in Fig. 4. This kind of operation can be extended to shapes that also exhibit rigid geometric distortions (as discussed in the paper, orientation is taken care of by the homogeneity of monomial symmetric polynomials and the completeness of spectral invariants and position/scale via normalization).
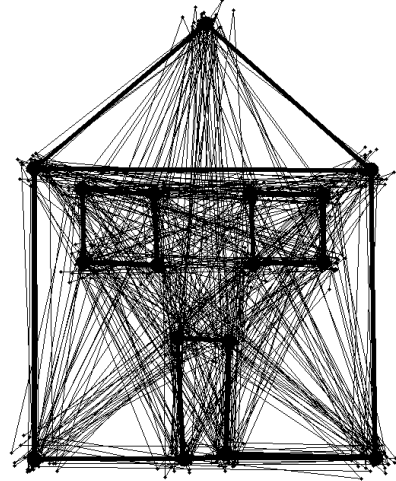


**Fig. 3**. The small dots are 30 noisy samples of a 17-point shape (the thin lines joining them represent the arbitrary order by which they are stored). The obtained mean shape is represented by the larger dots (joined by thick lines, used for better visualization).
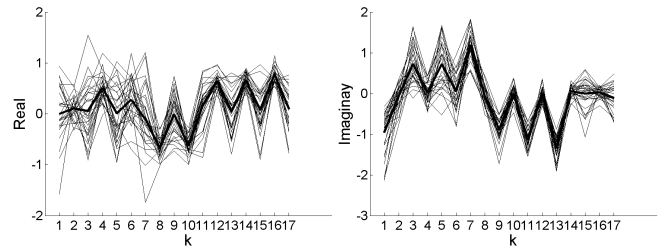


**Fig. 4**. Real and imaginary parts (normalized for better visualization) of the elementary symmetric polynomials representing the shapes in Fig. 3. The thin lines are the representations of the noisy samples; the thick one is their mean, which enables computing the mean shape.

## 11. CONCLUSION

We have discussed the usage of *symmetric polynomials* for the representation of 2D shapes, seen as unordered sets of points. After analytically showing properties of the representation, we illustrate with a task that requires *invariance* and *completeness*: the computation the mean shape, *i.e.*, the mean of a set of unordered point sets.

## 12. REFERENCES

[1] S. Loncaric, "A survey of shape analysis techniques," *Pattern Recognition*, vol. 31, no. 8, 1998.

[2] R. Veltkamp, "Shape matching: Similarity measures and algorithms," in *IEEE Int. Conf. on Shape Modeling and Applications*, Genoa, Italy, 2001.

[3] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, no. 1, 2004.

[4] T. Adamek and N. O'Connor, "A multiscale representation method for nonrigid shapes with a single closed contour," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 5, 2004.

[5] D. Zhang and G. Lu, "Study and evaluation of different Fourier methods for image retrieval," *ELSEVIER Image and Vision Computing*, vol. 23, no. 1, 2005.

[6] I. Bartolini, P. Ciaccia, and M. Patella, "Warp: accurate retrieval of shapes using phase of Fourier descriptors and time warping distance," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, 2005.

[7] C. Small, *The Statistical Theory of Shape*, Springer Series in Statistics. Springer-Verlag, 1997.

[8] D. Kendall, D. Barden, T. Carne, and H. Le, *Shape and Shape Theory*, John Wiley & Sons, 2009.

[9] J. Flusser, B. Zitova, and T. Suk, *Moments and Moment Invariants in Pattern Recognition*, Wiley, 2009.

[10] J. Crespo and P. Aguiar, "Revisiting complex moments for 2-D shape representation and image normalization," *IEEE Trans. on Image Processing*, vol. 20, no. 10, 2011.

[11] P. Besl and N. McKay, "A method for registration of 3-D shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, 1992.

[12] G. McNeill and S. Vijayakumar, "Hierarchical procrustes matching for shape retrieval," in *IEEE Conf. on Computer Vision and Pattern Recognition*, New York, NY, USA, 2006.

[13] G. McNeill and S. Vijayakumar, "A probabilistic approach to robust shape matching," in *IEEE Int. Conf. on Image Processing*, Atlanta, GA, USA, 2006.

[14] I. Macdonald, *Symmetric Functions and Hall Polynomials*, Oxford University Press, 1998.

[15] D. Dummit and R. Foote, *Abstract Algebra*, Prentice Hall, 1999.

[16] J. Yellott and G. Iverson, "Uniqueness properties of higher-order autocorrelation functions," *Journal of the Optical Society of America A*, vol. 9, no. 3, 1992.

[17] I. Jouny and R. Moses, "The bispectrum of complex signals: definitions and properties," *IEEE Trans. on Signal Processing*, vol. 40, no. 11, 1992.

[18] R. Kakarala and G. Iverson, "Uniqueness of results for multiple correlations of periodic functions," *Journal of the Optical Society of America A*, vol. 10, no. 7, 1993.

[19] R. Negrinho and P. Aguiar, "Shape representation via elementary symmetric polynomials: a complete invariant inspired by the bispectrum," in *IEEE Int. Conf. on Image Processing*, Melbourne, Australia, 2013.

[20] B. Sadler and G. Giannakis, "Shift-and rotation-invariant object reconstruction using the bispectrum," *Journal of the Optical Society of America A*, vol. 9, no. 1, 1992.

[21] J. Heikkila, "A new class of shift-invariant operators," *IEEE Signal Processing Letters*, vol. 11, no. 6, 2004.

[22] K. Kubicki and R. Kakarala, "Experimental results of bispectral invariants discriminative power," in *SPIE Conf. on Three-dimensional Image Processing (3DIP) and Applications II*, Burlingame, CA, USA, 2012.

[23] K. Kanatani, *Group Theoretical Methods in Image Understanding*, Springer Verlag, Berlim, Germany, 1990.

[24] R. Negrinho, "Shape Representation via Symmetric polynomials: a Complete Invariant Inspired by the Bispectrum," M.S. thesis, Instituto Superior Técnico, Lisboa, Portugal, 2013, Available at `www.isr.ist.utl.pt/˜aguiar/rnegrinho_thesis.pdf`.