TÉCNICO LISBOA

Shape Representation Via Symmetric Polynomials: a Complete Invariant Inspired by the Bispectrum

Renato Manuel Pereira Negrinho

Thesis to obtain the Master of Science Degree in:

Electrical and Computer Engineering

Examination Committee

Chairperson: Prof. João Fernando Cardoso Silva Sequeira
Supervisor: Prof. Pedro Manuel Quintas Aguiar
Members of the Committee: Prof. Mário Alexandre Teles de Figueiredo

October 2013

aos meus pais

Agradecimentos

Cinco anos atrás, jamais poderia imaginar como iria ser a minha passagem pelo Técnico. Os primeiros tempos foram difíceis, mas após o choque inicial, o sentimento que ficou foi, claramente, o da mais profunda satisfação. Sinto-me privilegiado de ter passado por esta instituição e, na minha opinião, de ter colhido o que ela tem para oferecer: uma imensa oportunidade para crescer e aprender. É com muito orgulho que digo que sou do Técnico.

No que toca a este trabalho, nada teria sido possível sem o professor Pedro Aguiar. Agradeço-lhe a paciência e o apoio; a discussão e a crítica. Com ele aprendi muito sobre a arte de fazer perguntas. Espero que a pertinência dos seus comentários tenha tido a merecida recepção. Mais do que um orientador, posso dizer que se tornou um amigo.

Agradeço à minha família e, particularmente, aos meus pais, que sempre trabalharam arduamente para dar aos filhos educações melhores do que aquelas a que eles tiveram acesso. Devo-lhes os valores que me incutiram e o entendimento de que tudo é possível quando somos determinados e, acima de tudo, trabalhamos muito e com o coração. Sei que eles estarão sempre a meu lado. Esta tese é dedicada a eles.

À Joana, agradeço por me dar tanto sem nunca pedir nada em troca. Devo-lhe muitos sorrisos. Obrigado por me aturares e me fazeres tão feliz.

Resumo

Nesta tese, abordamos o problema de representação de formas bidimensionais na sua forma mais geral, i.e., conjuntos arbitrários de pontos. Exemplos destas formas surgem em várias situações, sob a forma de conjuntos esparsos de pontos representativos, ou conjuntos densos de pontos de contornos de imagens. Os nossos alvos são problemas de reconhecimento, onde é fundamental gerir dois objectivos contraditórios: formas que diferem de transformações rígidas ou permutações dos pontos devem ter a mesma representação (invariância), mas formas geometricamente distintas devem ter representações diferentes (completude).

Introduzimos uma nova representação de forma que junta propriedades dos polinómios simétricos e do biespectro. Tal como o espectro de potência, o biespectro é invariante a translações do sinal; mas, ao contrário do espectro de potência, o biespectro é completo. Conjuntos particulares de polinómios simétricos, os chamados polinómios elementares simétricos e as somas de potências, são completos e invariantes a permutações das variáveis. Mostramos que estes polinómios dos pontos da forma dependem da orientação de uma maneira que nos permite interpretá-los no domínio da frequência e construir um biespectro a partir deles. O resultado é uma representação de forma que é completa e invariante a transformações rígidas e a permutações dos pontos.

Descrevemos o problema de representação de forma de uma maneira muito geral através do uso de conceitos de teoria de grupos. O conceito de forma é determinado pela definição de transformações preservadoras de forma (e.g, permutação dos pontos e/ou transformações geométricas) através de acções de grupos. As formas são então identificadas com as órbitas das acções daqueles grupos e representar forma reduz-se a representar essas órbitas. Desta maneira, tal como pretendido, elementos que pertencem à mesma órbita têm a mesma representação e elementos que pertencem a órbitas diferentes têm representações diferentes. A representação de forma proposta na tese atinge estes objectivos.

Descrevemos como calcular eficientemente a representação proposta usando programação dinâmica e terminamos descrevendo experiências que ilustram as propriedades provadas.

Palavras-chave: Representação de forma, Reconhecimento de forma, Invariante completo, Biespectro, Polinómios simétricos, Teoria de grupos, Acção de grupo.

Abstract

We address the representation of two-dimensional shapes in its most general form, i.e., arbitrary sets of points. Examples of these shapes arise in multiple situations, in the form of sparse sets of representative landmarks, or dense sets of image edge points. Our goal are recognition tasks, where the key is balancing two contradicting demands: shapes that differ by rigid transformations or point relabeling should have the same representation (invariance), but geometrically distinct shapes should have different representations (completeness).

We introduce a new shape representation that marries properties of the symmetric polynomials and the bispectrum. Like the power spectrum, the bispectrum is insensitive to signal shifts; however, unlike the power spectrum, the bispectrum is complete. Particular sets of symmetric polynomials, the so-called elementary ones and the power sums, are complete and invariant to variable relabeling. We show that these polynomials of the shape points depend on the shape orientation in a way that enables interpreting them in the frequency domain and building from them a bispectrum. The result is a shape representation that is complete and invariant to rigid transformations and point relabeling.

We describe the shape representation problem in a very general way by using concepts of group theory. The concept of shape is determined by the definition of the required shape-preserving transformations (e.g., point relabeling and/or geometric ones) through group actions. Shapes are then identified with the orbits of the actions of those groups and shape representation amounts to representing those orbits. This way, as pretended, elements that belong to the same orbit have the same representation and elements that belong to different orbits have different representations. The proposed shape representation attains this goal.

We describe how the proposed representation can be efficiently computed from the shape points using dynamic programming and end by describing experiments that illustrate the proved properties.

Keywords: Shape representation, Shape recognition, Complete invariant, Bispectrum, Symmetric polynomials, Group theory, Group action.

Contents

List of Figures

Chapter 1

Introduction

We start by introducing and motivating the problem of shape representation. Then, after reviewing current approaches and their limitations, we briefly describe our work. The chapter ends with an outline of the thesis content.

1.1 Problem and Motivation

In many cases, an object can be recognized by its shape alone. For example, a human would hardly mistake, even without any texture or color information, the shape of a chicken for the shape of a dog. In this thesis, we address the problem of representing two-dimensional (2D) shapes, having in mind shape retrieval tasks, i.e., finding, in a shape database, the shapes that are similar to a query shape.

For us, a 2D shape is an arbitrary set of points in the plane. In some approaches, researchers only consider shapes that are well-described by closed contours. Although this makes the representation problem significantly easier, it also severely limits the kinds of shapes that can be dealt with. In fact, in many real-life scenarios (e.g., trademark retrieval), the underlying shapes contain multiple contours, lines, and/or small isolated regions that are better modeled as simple 2D points.

Comparing two collections of points is difficult because, as it happens in general shape recognition problems, they are related by unknown geometric transformations (due to different position, orientation, and size) and permutation (due to the absence of labels for the points). For example, although all the 2D shapes in Figure 1.1 are the same, this is not easily captured from the corresponding lists of 2D point coordinates.

The difficulties summarized in the previous paragraph, but also the fact that modern machine learning algorithms require more than the capability of comparing pairs of shapes, motivates the search for a representation that enable shapes to be treated as points in an abstract shape space, where machine learning algorithms can be applied. Naturally, the representation must be *invariant* to geometric transformations and point permutation but also *complete*, in the sense of fully describing the underlying shape, i.e., shapes that differ only by a rigid geometric transformation or point relabeling are mapped to the same point in the abstract shape space while shapes not related by these transformations are mapped to

Figure 1.1: Each footprint is an instance of the same shape, i.e., the same set of 2D points is displayed with distinct scales, orientations, and positions. When attempting to recognize the shape from the list of 2D point coordinates of an instance, besides these geometric distortions, there is an additional difficulty (hidden when the shapes are displayed): the fact that the point lists have in general distinct orders.

different points in the abstract shape space. This is illustrated in Figure 1.2.

1.2 Current Approaches and their Limitations

Shape representation has deserved the attention of several researchers in the past and complete surveys can be found in [1, 2, 3, 4]. In the sequel, we summarize the limitations of a few meaningful approaches.

Although connected regions can be represented by a one-dimensional contour, which is easier to code, see, e.g., [5], this is not the case of general shapes, i.e., arbitrary sets of points in the plane. The statistical theory of shape [6] addresses this problem in situations where the points are labeled (usually in small number, denoted by landmarks). However, the problem remains for reasonably large sets of points without labels or natural ordering, e.g., those arising from automatic edge/corner/interest-point detection.

When dealing with point clouds, translation and scale are easily taken care of through normalization. However, this is not the case of rotation and permutation, whose simultaneous estimation leads to a non-convex problem. Iterative methods such as the Iterative Closest Point (ICP) [7] or its probabilistic versions based on Expectation-Maximization (EM), e.g., [8], tackle this problem but suffer from the usual sensitivity to the initialization, exhibiting uncertain convergence. When the relative orientation of the shapes to compare is known, the estimation of the permutation relating the point sets can be casted into a convex optimization problem [9]. However, normalizing a point set with respect to rotation is harder than it could seem at first sight. In fact, although theoretically sustained moment-based methods have been proposed (see [10] and the references therein), degenerate cases have been successively identified, showing that these methods can be sensitive to the noise and motivating subsequent research, e.g., [11].

Reference [12] proposes a representation that overcomes the need to compute correspondences between shape points. It is permutation-invariant and complete, but it is not rotation invariant, requiring pairwise alignments for shape comparison. Moment-based representations of image patterns have been

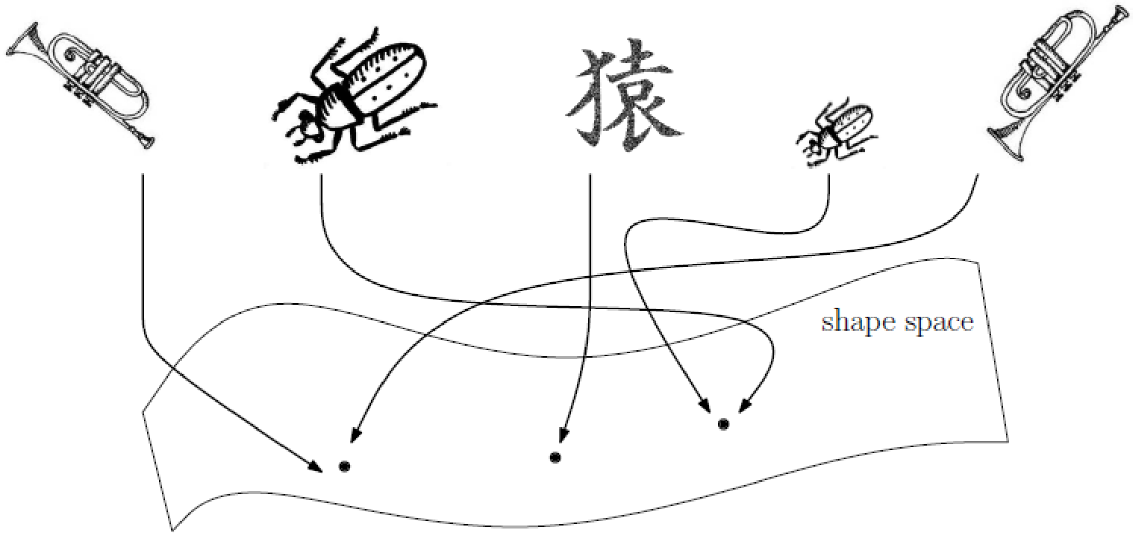

Figure 1.2: Pictorial view of the problem addressed in this thesis: how to represent shapes in such a way that distinct shapes are mapped to distinct points in a shape space, but instances of the same shape (differing by geometric transformations and point re-ordering) are mapped to the same point?

used since the sixties due to their geometric invariance properties but their completeness only recently have been focus of attention [10].

1.3 Proposed Approach

In this thesis we introduce a new shape representation that is complete and invariant with respect to geometric transformations and point permutations. We draw on properties of the *symmetric polynomials* and the *bispectrum*. The symmetric polynomials, extensively studied in algebra and in combinatorics, are motivated by the permutation invariance/completeness. The bispectrum, which has received attention in signal processing, is motivated by the geometric invariance/completeness.

It can be shown that particular subclasses of symmetric polynomials (the *power sums* and the so-called *elementary symmetric polynomials*) on a set of variables suffice to determine them up to a permutation. This enables us to factor out the permutation of the shape points in an efficient manner, while guaranteeing the completeness of the proposed shape representation scheme.

To obtain complete invariance with respect to shape rotation, we draw inspiration from the bispectrum. While the power spectrum of a signal is insensitive to signal shifts but does not uniquely determine the underlying signal, its bispectrum inherits the invariance and determines the signal, up to a shift. We show that shape rotation affects the symmetric polynomials in a similar way as a signal shift affects the coefficients of its Fourier series. Based on this property, we propose a shape representation that consists in a bispectrum computed from the symmetric polynomials, being then complete and invariant with respect to point permutation and shape orientation (translation and scale are taken care of through normalization).

We believe that the connection just summarized is the most insightful part of our work. Our approach links two well-known mathematical objects that have received extensive attention in the research literature. The complete invariance of the bispectrum with respect to shifts has been used in image processing, but not to represent arbitrary sets of points. For example, in reference [13] image rotations are transformed into shifts in the polar domain and reference [14] uses the bispectrum of one-dimensional image projections (Radon transform). Although the completeness of bispectrum even inspired other authors to extend its applicability beyond commutative groups [15, 16], to the best of our knowledge, none addresses the representation of 2D shapes with the generality we do here.

In spite of the key aspect just referred, our work lead to other original contributions, which are singled out in a synthetic way in the following list:

- Shapes as unordered sets of 2D points and their representations via groups and group actions.
- Using particular subsets of symmetric polynomials to represent 2D shapes in a way that factors out point label permutations.
- The homogeneity property of the symmetric polynomials in general and its relation to Fourier analysis.
- Efficient computation of the monomial symmetric polynomials using dynamic programming.
- Using the bispectrum computed from symmetric polynomials to represent 2D shapes in a way that also factors out rotations.

The representation based on elementary symmetric polynomials was presented in [17]; reference [18] summarizes our work.

1.4 Thesis Organization

The remaining of the thesis is structured as follows.

In Chapter 2, we formulate the problem of shape representation with generality. Shape is defined as what remains after factoring out the action of a group of transformations. The formalization of the shape representation problem uses thus concepts of group theory, motivating the need to introduce the notions of group and group action. These concepts provide a rigorous way to interpret shapes as orbits of particular group actions. They also enable elegant derivations of the invariance properties presented in subsequent chapters.

Chapter 3 deals with the symmetric polynomials. We derive properties of such polynomials, shedding light on why these are interesting for the problem of shape representation. We particularize two subclasses of symmetric polynomials, namely, the power sums and the elementary symmetric ones. Two important properties of these polynomials are completeness and homogeneity. The completeness allows us to represent an arbitrary set of points up to a permutation. The homogeneity allows us to link the symmetric polynomials to Fourier analysis. Finally, we study how the symmetric polynomials can be computed, deriving an efficient approach based on dynamic programming.

In Chapter 4, we address the bispectrum. We show how signal shifts affect their frequency representation, making the connection with what happens in the case of the symmetric polynomials of a 2D shape. We illustrate the desired properties of the bispectrum by contrasting with the power spectrum: while the latter is invariant to shifts but incomplete, the former inherits the invariance but exhibits completeness.

Chapter 5 describes the proposed representation. First, we introduce the group action that defines the shape-preserving transformations, formalizing the concept of shape. The transformations that we consider are translation, rotation, scaling, and permutation of the labels. They are successively factored out and invariance and completeness properties of the intermediate representations are analytically shown. The final invariant and complete representation is attained by composing the partial invariants.

In Chapter 6, we illustrate the properties of the proposed representation with experiments such as shape classification in the presence of noise, automatic clustering of binary images, and classification of shapes extracted from real trademark images with simple edge detection.

Chapter 7 concludes the thesis. We summarize our approach and the properties of the proposed representation and end by outlining research paths that emerged from our work.

In the appendices, we include topics that would hinder the presentation in the main body of the thesis. Appendix A discusses the notion of shape dissimilarity and its implications. In Appendix B, we study the impact on the elementary symmetric polynomials and the power sum symmetric polynomials of a perturbation of their arguments. In Appendix C, we extend the proposed representation to also include invariance and completeness with respect to reflections by defining a new shape dissimilarity measure.

Chapter 2

Shape Representation

In this chapter, we start by discussing the problem of shape representation from an intuitive standpoint, raising questions that need formal addressing. The discussion intends to make the reader aware of the freedom that exists in the definition of shape.

We then present concepts of group theory and use them to formulate the problem of shape representation. These concepts will be rather abstract at first, but we will see instantiations of them in Chapter 3 and Chapter 4, where we deal with invariants to a particular group actions, and in Chapter 5, where we specify a definition of shape and construct the corresponding representation.

2.1 First Remarks

Our work concerns the representation two-dimensional (2D) shapes, that are given as ordered sets of points in the plane, which we call shape representatives (sometimes we call them just representatives or, when it is clear from the context, shapes). When we talk about shapes there is usually some invariance involved. For example, a square is still a square if it is rotated, scaled and translated arbitrarily. The invariance that captures the notion of shape will be defined through a set of shape-preserving transformations, which have the property that, if we apply them to particular shape representative, the transformed shape representative has the same shape as the untransformed one. We will define the shape preserving transformation through the action of a group.

The shape representatives that we consider live in $\mathbb{C}^N$, where N is the number of shape points. For now, we consider that the number of points is fixed at the same value for all shapes. This is mostly for convenience reasons and will be dealt with later. The identification of $\mathbb{C}$ with $\mathbb{R}^2$ is trivial and, therefore, no information is lost by working in this space instead. The choice of $\mathbb{C}$ is motivated by its more favorable algebraic properties.

By using $\mathbb{C}^N$ to represent representatives of shapes, we realize that they are inherently labeled due to the distinction between coordinates. This will allow us to treat the problem in a very general manner, beginning with arbitrary ordered sets of points in the plane. For example, we can consider that each of the distinct labeled sets of points represents a different shape. This effectively corresponds to having only the identity as a shape-preserving transformation.

The shape s is identified with its set of shape representatives $R_s \subset \mathbb{C}^N$, which contains all the shape representatives $r_s \in \mathbb{C}^N$ that are instantiations of the shape s. Each of the coordinates of r_s is simply a labeled point in the plane:

$$r_s = \begin{bmatrix} z_1 \\ \vdots \\ z_N \end{bmatrix} \in \mathbb{C}^N. \tag{2.1}$$

We define shape by partitioning the space of shape representatives $\mathbb{C}^N$ into equivalence classes. Each of the resulting equivalence classes corresponds to a different shape. We will define the equivalence classes through the action of a group that defines the shape-preserving transformations. Here, two elements r_s and $r_{s'}$ in the space of shape representatives are representatives of the same shape if and only if they are related by a shape-preserving transformation, i.e., if there is a shape-preserving transformation that maps one representative to the other.

In our initial example, where we had only the identity as a shape-preserving transformation, each representative r_s is related (by the identity shape-preserving transformation) only with itself and with no other different representative r'_s. This means that each $r_s \in \mathbb{C}^N$ is identified with a different shape and that the set of all shape representatives R_s for a shape s has a single element r_s (it is a singleton set). This is a trivial example.

In more interesting cases, we have nontrivial transformations that we want to deem as shape-preserving. Some commonly considered shape-preserving transformations are the rigid ones, where two shape representatives represent the same shape if there is a translation, rotation, and reflection in the plane $\mathbb{C}$, that maps one representative to the other. Another interesting set of shape-preserving transformations are the permutations of the labels, where two representatives represent the same shape if they differ by a permutation of the coordinates. This gives rise to unlabeled shapes, i.e., shapes whose points do not have natural labels.

In general, considering a set of shape-preserving transformations, two representatives correspond to the same shape if there is a shape-preserving transformation that maps one to the other. Equivalently, two shapes s and s' are equal if and only if $R_s = R_{s'}$. We will use shape and its identification with a set of representatives, interchangeably.

Since a shape s can be identified with its set of representatives R_s, we may think of representing s by explicitly storing R_s. Nonetheless, this is usually not possible (note that the set of representatives is finite for the case of permutations of the labels, but infinite for the case of rigid transformations; even in the case of the unlabeled shapes, the explicit storage of the R_s is intractable because the set has $N!$ elements.) Figure 2.1 illustrates this difficulty. If the shape-preserving transformations are rotations and permutations of the labels, all the shape representatives in Figure 2.1 belong to the same set of representatives and, therefore have to be represented as such. To capture shape, we need to somehow encode this relation of belonging of a given shape representative to a given set of shape representatives.

A more convenient way to represent the sets of representatives is to map the problem to some space where each set of representatives can be represented by a point in that space, which identifies a shape

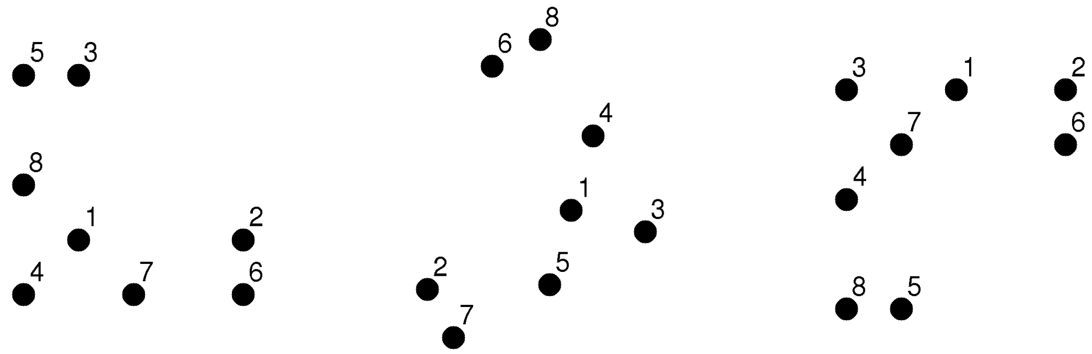

Figure 2.1: Examples of shape representatives related by rotation and permutation of the labels. If the shape preserving transformations include rotations and permutations of the labels, all these representatives belong to the set of representatives of the same shape. To represent shape we need to capture this relation.

(see Figure 1.2). We require this map to keep all shape information and to be computable just from one representative of the shape (and not just from the full set of representatives). The motivation for these requirements is obvious since, given two different representatives, knowing if they correspond to the same shape or not boils down then to the evaluation of this mapping and to the comparison of the results in the new space.

Stating that all shape information is kept means that, besides the shape-preserving transformations, no other information is factored out. In this case, two representatives will be mapped to the same point in this new space if and only if they correspond to the same shape. This amounts to having a complete and invariant representation for shapes. We call this new space, the space of shape representations $\mathcal{R}$ and the mapping that matches each shape representative to its shape representation, the shape representation mapping $\rho : \mathbb{C}^N \to \mathcal{R}$.

2.2 Group Theory

We outlined what we would like to do for solving the shape representation problem without ever describing how we might go about constructing the specified objects. Now, we formalize the concepts of shape-preserving transformation, shape representative set, space of shape representations, and shape representation mapping. We begin by introducing notions of group theory that will help us in the task.

A group is a tuple $(G, \cdot)$, where G is a set, sometimes called the underlying set, and $\cdot$ is a mapping $G \times G \to G$, satisfying the following axioms:

Closure: For any x and y in G, xy is also in G;

Associativity: For any x, y, and z in G, $x(yz) = (xy)z$;

Existence of identity: There is an unique element of G, denoted e and called the identity element, such that, for any g in G, $eg = ge = g$;

Existence of inverses: For any g in G, there is a corresponding element, denoted g^{-1} and called the inverse of g, such that $gg^{-1} = g^{-1}g = e$.

Note that although we use the multiplicative notation to denote the group operation, that does not mean that the group is necessarily commutative, i.e., for every g_1 and $g_2 \in G$, g_1g_2 may not be equal to g_2g_1. It is common practice to refer to a group $(G, \cdot)$ by just G. We will also adopt this usage, however, it must be kept in mind that a group is identified, not just by its underlying set G, but both by its underlying set G and its group operation $\cdot$. The simplest group has just the identity element. It is possible to specify a finite group G by identifying a set $\{g_1, \dots g_{|G|}\}$ (where $|G|$ is the number of elements in the group, usually called the order of the group) and explicitly writing a multiplication table with $|G|^2$ entries that defines the group operation. Obviously, the operation defined by the multiplication table has to satisfy the group axioms, otherwise the resulting structure is not a group.

The seemingly simple four group axioms above give rise to an exceedingly rich mathematical structure on which there is an extensive body of knowledge under the field of group theory and related subfields. Even though, at first, the notion of a group may seem like a rather abstract one, we can easily come up with several concrete examples:

1. Complex numbers under addition, $(\mathbb{C}, +)$;
2. Nonzero complex numbers under multiplication, $(\mathbb{C} \setminus \{0\}, \cdot)$;
3. Vector space under addition, e.g., $(\mathbb{R}^N, +)$ and $(\mathbb{C}^N, +)$;
4. Real N-by-N matrices of unit determinant under matrix multiplication, called the special orthogonal group and denoted by $SO(N)$;
5. Permutations of n symbols under composition, called the symmetric group and denoted by S_n.

A subgroup H of a group G is a group on its own right, but we call it a subgroup for tying its definition to the larger group. The underlying set of the subgroup H is simply a subset of the underlying set of the group G. The group operation for the subgroup H is the same as the one of the group G. This allows us to specify smaller groups.

A way to build a larger group G given two smaller ones G_1 and G_2 is by taking their direct product $G_1 \times G_2$. The underlying set of G is the direct product of the underlying sets of the groups G_1 and G_2. The group operation of the new group G is constructed from the group operations of G_1 and G_2 by having the group operation of G_1 act on the part of G that pertains to G_1 and by having the group operation of G_2 act on the part of G that pertains to G_2. More concretely, if g_1 and g_2 are in G, where $g_1 = (g_1^1, g_1^2)$ and $g_2 = (g_2^1, g_2^2)$, and where g_1^1 and g_2^1 are in G_1 and g_1^2 and g_2^2 are in G_2. The product of g_1g_2 is given by $(g_1^1g_2^1, g_1^2g_1^2)$. We can easily verify that the product group G verifies the group axioms. Taking the direct product of some groups basically amounts to stacking them together. No new information is added besides the one already contained in the component groups. The commutativity of the group $G = G_1 \times G_2$ depends on the commutativity of its components groups G_1 and G_2. Nonetheless, every element $g = (g^1, g^2)$ of G can be written as the product of the elements $g_1 = (g^1, e^2)$ and $g_2 = (e^1, g^2)$, where e^1 and e^2 are the identities of the component groups G_1 and G_2, respectively, which commute, i.e., we have $g = g_1g_2 = g_2g_1$.

Now, we present the concept of a group action. This will bring us one step closer to formalizing the notion of shape-preserving transformations. A group G is said to act on a set X when there is a map $\phi : G \times X \to X$ satisfying:

Identity map: $\phi(e, x) = x$ for every x in X;

Group Homomorphism: $\phi(g_1, \phi(g_2, x)) = \phi(g_1 g_2, x)$, for every g_1 and g_2 in G and every x in X.

When it results in no confusion, we use the group element g to denote the action by g. In this case, $\phi(g, x)$ is denoted by $g(x)$. The action itself is determined by the choice of the map ϕ. The second condition means that we can commute between taking the product $g_1 g_2 \in G$ and acting by this element or acting, successively, by g_2 and then by g_1. The satisfaction of these properties by ϕ turns G into a transformation group and X into a G-set. Note that every group acts on itself by group multiplication. In this case, the set X is the underlying set of G. Another interesting fact is that the trivial action is a valid action for every group, i.e., having $\phi(g, x) = x$ for every $g \in G$ and every $x \in X$. It can be trivially verified that this satisfies the properties of a group action irrespective of the group G.

As a nontrivial example, we can make the permutation group S_N act on $\mathbb{C}^N$ by defining the map

$$\phi(\pi, r_s) = \begin{bmatrix} z_{\pi(1)} \\ \vdots \\ z_{\pi(N)} \end{bmatrix}, \tag{2.2}$$

where $\pi \in S_N$ and $r_s \in \mathbb{C}^N$. This action permutes the labels of the points of the representatives according to the element π of the group of permutations S_N. This is what we call the natural action of the symmetric group S_N on space of representatives $\mathbb{C}^N$. The identity property is trivially verified because the identity permutation does not change the labels. The homomorphism property is verified:

$$\phi(\pi_1, \phi(\pi_2, r_s)) = \begin{bmatrix} z_{\pi_1(\pi_2(1))} \\ \vdots \\ z_{\pi_1(\pi_2(N))} \end{bmatrix} = \begin{bmatrix} z_{\pi_1\pi_2(1)} \\ \vdots \\ z_{\pi_1\pi_2(N)} \end{bmatrix} = \phi(\pi_1\pi_2, r_s), \tag{2.3}$$

for any $\pi_1, \pi_2 \in S_N$ and $r_s \in \mathbb{C}^N$.

Defining the shape-preserving transformations through the action of a group automatically endows the set of shape-preserving transformations with group properties: the identity transformation is a shape-preserving transformation; the inverse of a shape-preserving transformation exists, being also a shape-preserving transformation; the composition of two shape-preserving transformations is a shape-preserving transformation. These properties are derived from the definitions of group and group action.

Considering $\phi(g, x)$, if we fix an element $x \in X$ and let g run over all the elements in G, we obtain the orbit $\mathcal{O}_x$ of x. Formally,

$$\mathcal{O}_x = \{\phi(g, x) | g \in G\}. \tag{2.4}$$

The orbits of the elements of X are either disjoint or the same and its union is the whole set X. We present a sketch of the proof of this fact. For each $x \in X$, the orbit $\mathcal{O}_x$ is nonempty since it has at least x

in it. Therefore, the union of all the orbits is the whole set X. The equality or disjointness can be proved by noting that if we have the orbits $\mathcal{O}_x$ and $\mathcal{O}_{x'}$, generated by elements x and $x' \in X$, and $\mathcal{O}_x \cap \mathcal{O}_{x'} \neq \emptyset$, then, there is an element $y \in \mathcal{O}_x \cap \mathcal{O}_{x'}$. By the definition of orbit, this implies that $y = \phi(g, x) = \phi(g', x')$, for some $g, g' \in G$. Acting with g^{-1} on both sides and using the properties of a group action, we get $\phi(g^{-1}, \phi(g, x)) = \phi(e, x) = x = \phi(g^{-1}g', x')$. This means that x is in the same orbit as x'. But this implies that the orbits are the same, since a element of the orbit runs over all the elements of the orbit when it is acted by the group.

The fact that the orbits of the action partition X means that they define a set of equivalence classes on X (in general, a set of equivalence classes on X is basically a partition of X). Two elements of X are equivalent if and only if they belong to the same orbit. This means that two elements are equivalent if and only if one can be mapped to the other by acting with some element of G. This is the equivalence relation that arises from the partition of X with the orbits.

An equivalence relation $\sim$ is a binary relation $X \times X$ satisfying the following properties:

Reflexive: $a \sim a$, for all a in X;

Symmetric: If $a \sim b$, then $b \sim a$, for all a, b in X;

Transitive: If $a \sim b$ and $b \sim c$, then $a \sim c$, for all a, b and c in X.

By having a set of equivalence classes on X, we get an equivalence relation on the elements of X that is given by: two elements of X are equivalent if and only if they belong to the same equivalent class. The converse is also true. An equivalence relation on X induces a set of equivalence classes on X where the equivalence class of some element of X is the set of all elements of X that are equivalent to it.

Another important concept that arises when we talk about group actions is the concept of a stabilizer. The stabilizer is related to the concept of an orbit. Both orbits and stabilizers are indexed by elements of X, but, while the elements of an orbit are elements of the set X, the elements of a stabilizer are elements of the group G. The stabilizer of an element $x \in X$ is the set of all group elements $g \in G$ that act on x by leaving it fixed. The stabilizer of $x \in X$ is denoted by G_x and it is formally defined as

$$G_x = \{g \in G | \phi(g, x) = x\}. \tag{2.5}$$

The notation G_x is to emphasize that the stabilizer is a subgroup of G. (We leave the proof of this to the reader.)

The group theoretical concepts just presented are but a tiny fraction of all group theory and abstract algebra. More information can be found in references such as [19, 20, 21].

2.3 Shapes as Orbits

As anticipated above, we define the shape-preserving transformations, which determine the notion of shape, through the action of some group on the set of shape representatives $\mathbb{C}^N$. The definitions of the desired group and action depend on the shape-preserving transformations that we want to consider. In most cases of interest, the group can be easily constructed by taking some smaller groups as building blocks and putting them together by taking the direct product. Given the group, the desired group action is also easy to build.

The set R_s, containing all the representatives of a shape s, is simply the orbit of any of its representatives r_s under the action of the group of shape-preserving transformations. Each orbit is a subset of $\mathbb{C}^N$ and the set of all orbits is a partition of $\mathbb{C}^N$. Two shapes are equivalent if they correspond to the same orbit or, equivalently, if given a representative of each shape, there is an action by some element of the group of shape-preserving transformations that maps one to the other. The space of all orbits is written as $\mathbb{C}^N/\sim$, where $\sim$ is the equivalence relation induced by the group action defining the shape-preserving transformations.

The space of all orbits is an example of a quotient space. The operation by which we obtain a quotient space is called quotienting out by an equivalence relation. An equivalence class is identified with a single point in this new space. Sometimes, as an intuitive explanation of how this quotienting out process works, it is said that the points in an equivalence class are glued or collapsed together into a single point. The space resulting from the collapse of all the equivalence classes is the quotient space. Figure 2.2 illustrates the notion of an equivalence class and a quotient set.

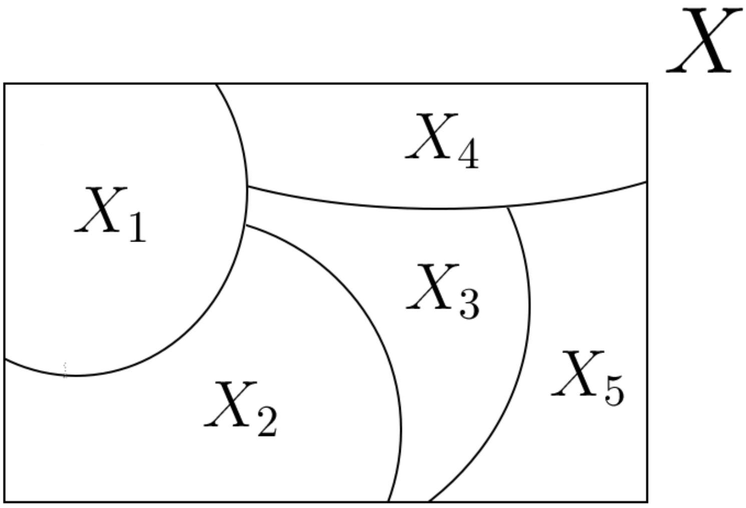

Figure 2.2: The set X is partitioned into the equivalence classes X_1, X_2, X_3, X_4, and X_5. The set of the five equivalence classes is an example of a quotient space $X/\sim$ and each of the equivalence classes is an element of this new space. The elements of X that belong to the same equivalence class are identified with the same point in quotient space. This identification is called the projection of X into the quotient space $X/\sim$.

A shape is naturally identified with its orbit, which contains all its representatives. We could, in principle, represent a shape by amassing all the elements of its orbit but, as we have already noted in Section 2.1, this is not computationally tractable in general. This is where the notion of a shape representation mapping comes into play.

A way to represent orbits is to find a function ρ that maps shape representatives in $\mathbb{C}^N$ to the space of shape representations $\mathcal{R}$. We require the mapping $\rho : \mathbb{C}^N \to \mathcal{R}$ to take the same value for all the elements of an orbit, i.e., it is constant when restricted to a particular orbit. This property is called invariance. We are indirectly factoring out the equivalence relations induced by the group action through the evaluation of the shape representation mapping ρ. The other key property is completeness. This property requires that two shape representations $\rho(r_s)$ and $\rho(r_{s'})$ are equal only if r_s and $r_{s'}$ are equivalent as defined by the group action. Our goal is to a find a complete and invariant representation. Figure 2.3 illustrates the partition of the space of shape of representatives $\mathbb{C}^N$ into shapes and the invariance and completeness of the shape representation mapping $\rho : \mathbb{C} \to \mathcal{R}$.

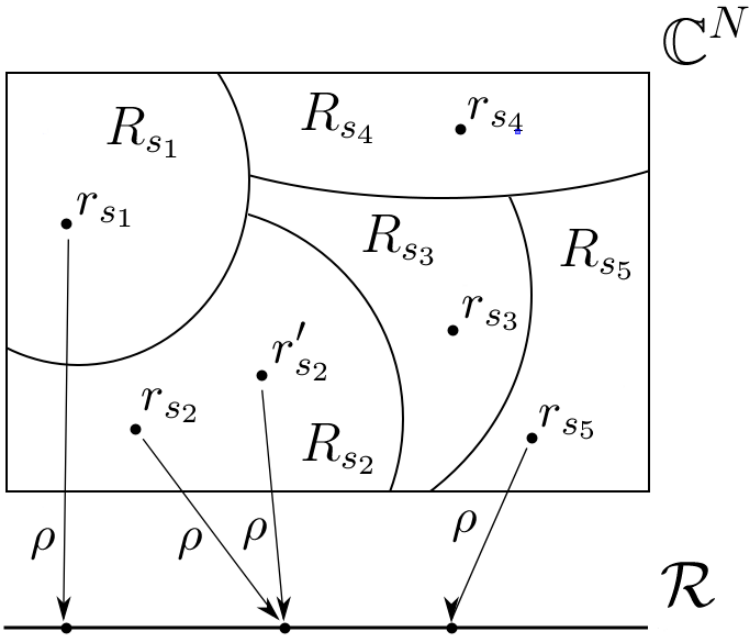

Figure 2.3: The space of shape representatives $\mathbb{C}^N$ is partitioned into five shapes. Each of these shapes has several shape representatives. To capture the shape information, the shape representation mapping $\rho : \mathbb{C}^N \to \mathcal{R}$ has to take the same value for all shape representatives of a shape (invariance), which is illustrated by $\rho(r_{s_2}) = \rho(r'_{s_2})$, and different values for shape representatives of different shapes (completeness), which is illustrated by $\rho(r_{s_1}) \neq \rho(r_{s_2}) \neq \rho(r_{s_5})$.

We first define a map $\rho_{\mathbb{C}^N}^{\mathbb{C}^N/\sim}$ that assigns to each representative r_s its corresponding orbit R_s (this is the projection onto the equivalence class). This map is surjective because every orbit has at least one representative. A second map is $\rho^{\mathcal{R}}_{\mathbb{C}^N/\sim}$, which assigns to each of the orbits R_s a point $\rho^{\mathcal{R}}_{\mathbb{C}^N/\sim}(R_s)$ in the space of shape representations $\mathcal{R}$. The final shape representation map is $\rho^{\mathcal{R}}_{\mathbb{C}^N}$: to each representative r_s, it assigns a point $\rho(r_s)$ that encodes the orbit on which r_s is present and therefore, the shape that r_s is a representative of. This map can be seen as the composition of the two previously defined maps:

$$\rho = \rho^{\mathcal{R}}_{\mathbb{C}^N} = \rho^{\mathcal{R}}_{\mathbb{C}^N/\sim} \circ \rho_{\mathbb{C}^N}^{\mathbb{C}^N/\sim}. \tag{2.6}$$

Even though in practice the representation ρ may not be computed as the composition of the mappings expressed in (2.6), it is an interesting idea to keep in mind because the invariance comes from the fact that $\rho_{\mathbb{C}^N}^{\mathbb{C}^N/\sim}$ maps any element in an orbit to that orbit and the completeness comes from requiring the injectiveness of $\rho^{\mathcal{R}}_{\mathbb{C}^N/\sim}$. Anyway, the computation of ρ as a composition of the mappings would imply dealing with whole orbits at an intermediate step, which is not tractable in general.

As for the properties of invariance and completeness, invariance has received attention, being completeness much harder to guarantee in general. In fact, many researchers introduced invariants to some group actions and experimentally validated if they actually are descriptive enough, i.e., if they keep enough relevant information about shape. Note that by itself the problem is nontrivial. We could just represent all the representatives of all the shapes by the same constant. Even though this representation is clearly invariant to any group action that we can conceive, it obviously does not keep any shape information, being therefore, useless.

Summarizing, in this chapter we approached the problem of shape representation from an abstract perspective by using notions of group theory. We start by identifying the group and the corresponding action on $\mathbb{C}^N$. This defines the notion of shape. For the identified group action, the shape representation mapping ρ has to be constructed. The process of construction of ρ will usually involve putting together invariants to form the invariant to the full group action. Its construction will be elaborated in subsequent chapters by presenting invariants that can serve as building blocks and then, finally, constructing an actual representation for a particular case. Remember that the representation mapping $\rho : \mathbb{C}^N \to \mathcal{R}$ is evaluated in the space of representatives $\mathbb{C}^N$ and does not involve explicitly the orbits. In our work, the space of shape representations $\mathcal{R}$ will be identified with $\mathbb{C}^M$, for some M that represents the dimension of the space of shape representations $\mathcal{R}$, and the shape dissimilarity measure on $\mathcal{R}$ will be the Euclidean distance on $\mathbb{C}^M$. We refer the reader to Appendix A for a discussion on how one may define the notion of shape dissimilarity.

Chapter 3

Symmetric Polynomials

A symmetric polynomial is a polynomial that is invariant to any permutation, i.e., relabeling, of its variables. In this chapter, we define the symmetric polynomials and study their properties. The two types of symmetric polynomials that will be more important to us are the elementary symmetric polynomials and the power sum symmetric polynomials. This is due to the fact that they completely represent a set of points in $\mathbb{C}$ apart from arbitrary permutations, allowing, therefore to build complete invariants to the action of the symmetric group.

We also present the monomial symmetric polynomials, which subsume both the elementary symmetric polynomials and the power sum symmetric polynomials. Monomial symmetric polynomials are interesting because, even though their general case will not be extensively used for the construction of shape invariants, they provides a framework to better reason about symmetric polynomials.

The concepts presented in this chapter will be used in Chapter 5, where we deal with a concrete group action defining the shape-preserving transformations and propose the corresponding shape representation mapping.

3.1 Definition

A polynomial $s : \mathbb{C}^N \to \mathbb{C}$ is called a symmetric polynomial if it is invariant to any permutation of its variables. This concept can be considered for polynomials in arbitrary fields, nonetheless, since in this work we are interested in $\mathbb{C}$, we will only consider polynomials in this field. For a reference in symmetric polynomials, see [22].

In a formal way, with the symmetric group S_N acting on the set of variables $z_1, \dots, z_N$ of the polynomial $s : \mathbb{C}^N \to \mathbb{C}$ (each element of S_N yields a labeling for the variables), we say that s is symmetric if and only if

$$s(z_1, \dots, z_N) = s(z_{\pi(1)}, \dots, z_{\pi(N)}), \tag{3.1}$$

for all π in S_N, where $z_1, \dots, z_N \in \mathbb{C}$ and π denotes, simultaneously, an element of the symmetric group S_N and the permutation that it induces. It may also be said that a symmetric polynomial is stable under the action of the symmetric group. Some examples of symmetric polynomials in N variables are:

$$p_1(z_1, \ldots, z_N) = z_1 + \ldots + z_N; \tag{3.2}$$

$$p_3(z_1, \ldots, z_N) = z_1^3 + \ldots + z_N^3; \tag{3.3}$$

$$s(z_1, \ldots, z_N) = z_1 + z_1^3 + \ldots + z_N + z_N^3; \tag{3.4}$$

$$e_2(z_1, \ldots, z_N) = z_1 z_2 + \ldots + z_1 z_N + z_2 z_3 + \ldots + z_2 z_N + \ldots + z_{N-1} z_N. \tag{3.5}$$

It is immediately clear that the first three polynomials (3.2), (3.3), and (3.4) are symmetric. The fourth one (3.5) is also symmetric since it is given by the sum of all products of pairs of the variables $z_1, \ldots, z_N$. A permutation of the variables amounts to reordering the terms in the sum, but that does not change the polynomial. It can also be seen that, summing or multiplying two arbitrary symmetric polynomials also yields a symmetric polynomial. (In fact, the symmetric polynomials form a ring [22].)

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

$$D(d_1, \ldots, d_N) = \sum_{n=1}^{N} d_n. \tag{3.6}$$

For example, each of the polynomials (3.2), (3.3), and (3.5) has only monomials of a given order: 1, 3 and 2, respectively.

It is clear that if a symmetric polynomial has a monomial $(d_1, \ldots, d_N)$, it also has to have all the monomials that can be generated by a permutation of the indexes, i.e., $(d_{\pi(1)}, \ldots, d_{\pi(N)})$, for any π in S_N. In fact, this is a perfectly natural way of generating symmetric polynomials. The polynomials generated in this way are called the monomial symmetric polynomials.

In this chapter, we explore three subclasses of symmetric polynomials that will be of interest to our work: the monomial symmetric polynomials, the power sum symmetric polynomials, and the elementary symmetric polynomials.

3.2 Monomial Symmetric Polynomials

The monomial symmetric polynomials are the most general type of symmetric polynomials that we consider. To construct them, we first pick an monomial $(d_1, \ldots, d_N)$, with $d_1 \geq \ldots \geq d_N$, i.e., we require $d_1, \ldots, d_N$ to be given in non-increasing order. This monomial indexes a symmetric polynomial and will be called its indexing monomial. 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 $(d_1, \ldots, d_N)$ and $(d_1', \ldots, d_N')$ originating the same symmetric polynomial (this would only occur if $d_1, \ldots, d_N$ and $d_1', \ldots, d_N'$ were related by a permutation). Naturally, a monomial that is not given in non-increasing order can be appropriately sorted, without changing the polynomial that results from the construction process.

From the indexing monomial $(d_1, \ldots, d_N)$, through the action of the symmetric group S_N, we generate all the monomials that can be obtained by a permutation of the variables. (See Section 2.2 for notions of group theory.) Summing all these monomials, we obtain the indexed symmetric polynomial

$$s'_{(d_1,\ldots,d_N)}(z_1, \ldots, z_N) = \sum_{\pi \in S_N} z_{\pi(1)}^{d_1} \ldots z_{\pi(N)}^{d_N}. \tag{3.7}$$

The action of the symmetric group S_N on the variables of this polynomial amounts to a reordering of the monomials in the sum, leaving the polynomial unchanged. As an example of the construction process (3.7), the indexing monomial $(3, 2, 1)$ gives rise to the polynomial

$$s'_{(3,2,1)}(z_1, z_2, z_3) = z_1^3 z_2^2 z_3 + z_1^3 z_2 z_3^2 + z_1^2 z_2^3 z_3 + z_1^2 z_2 z_3^3 + z_1 z_2^3 z_3^2 + z_1 z_2^2 z_3^3. \tag{3.8}$$

Note that (3.8) sums all possible monomials that can be generated as a permutation of the indexing monomial $(3, 2, 1)$.

When not all $d_1, \ldots, d_N$ are distinct, there are permutations that leave the indexing monomial fixed, meaning that the action of S_N on the indexing monomial $(d_1, \ldots, d_N)$ has nontrivial stabilizer $G_{(d_1,\ldots,d_N)}$, i.e, there are other group elements besides the identity that act on the indexing monomial $(d_1, \ldots, d_N)$ by leaving it unchanged. The permutations $\pi \in S_N$ in the stabilizer $G_{(d_1,\ldots,d_N)}$ are those that satisfy $d_1 = d_{\pi(1)}, \ldots, d_N = d_{\pi(N)}$. Each of the different monomials $z_{\pi(1)}^{d_1} \ldots z_{\pi(N)}^{d_N}$ appears a number of times equal to the order of the stabilizer $G_{(d_1,\ldots,d_N)}$. In the construction, to cancel this factor, we divide by the order of the stabilizer, $|G_{(d_1,\ldots,d_N)}|$. As an example, consider the indexing monomial $(3, 1, 1)$ which gives rise to the polynomial

$$\begin{aligned} s'_{(3,1,1)}(z_1, z_2, z_3) &= z_1^3 z_2 z_3 + z_1^3 z_2 z_3 + z_1 z_2^3 z_3 + z_1 z_2 z_3^3 + z_1 z_2^3 z_3 + z_1 z_2 z_3^3 \\ &= 2z_1^3 z_2 z_3 + 2z_1 z_2^3 z_3 + 2z_1 z_2 z_3^3. \end{aligned} \tag{3.9}$$

The stabilizer $G_{(3,1,1)}$ has the identity permutation and the permutation that exchanges the ones in the indexing monomial $(3, 1, 1)$, therefore $|G_{(3,1,1)}|$ equals two.

From the indexing monomial $(d_1, \ldots, d_N)$, we now count how many different monomials are summed in (3.7). Let P be the number of distinct values that are taken by $d_1, \ldots, d_N$, and consider the pairs of variables $(v_1, b_1), \ldots, (v_P, b_P)$, where v_n denotes one of the values taken by $d_1, \ldots, d_N$, and b_n denotes how many times v_n occurs in $d_1, \ldots, d_N$. Obviously, $\sum_{n=1}^{P} b_n = N$. (For example, for the indexing monomial $(3, 1, 1)$, we have $P = 2$ and $v_1 = 3$, $b_1 = 1$, $v_2 = 1$, and $b_2 = 2$.)

If all the $d_1, \ldots, d_N$ are different, we have $P = N$ and $b_1 = \ldots = b_N = 1$. In this case, the stabilizer is trivial and there will be $N!$ different monomials in the sum. If we have repeated values in $d_1, \ldots, d_N$, we have $P < N$ and some of the values of $b_1, \ldots, b_P$ will be larger than one.

The order of the stabilizer for the general case can be determined by noticing that there are: $b_1!$ ways to permute the b_1 occurrences of v_1, $b_2!$ ways to permute the b_2 occurrences of v_2, and so on. None of these permutations change the monomial $(d_1, \ldots, d_N)$. Therefore, we can generate an element of the stabilizer $G_{(d_1,\ldots,d_N)}$ by picking one of $b_1!$ permutations, and then picking one of $b_2!$ permutations, and so on. This yields

$$|G_{(d_1,\ldots,d_N)}| = b_1! \ldots b_P!. \tag{3.10}$$

From this follows that there are $\frac{N!}{b_1!\ldots b_P!}$ distinct monomials to sum.

The process of construction of the monomial symmetric polynomial corresponding to the indexing monomial $(d_1, \ldots, d_N)$ is summarized by

$$s_{(d_1,\ldots,d_N)}(z_1, \ldots, z_N) = \frac{1}{|G_{(d_1,\ldots,d_N)}|} \sum_{\pi \in S_N} z_{\pi(1)}^{d_1} \ldots z_{\pi(N)}^{d_N}. \tag{3.11}$$

It is straightforward to prove that the resulting polynomial is symmetric. An element π' of the symmetric group S_N acting on the polynomial variables, leads to

$$\begin{aligned} \pi'(s_{(d_1,\ldots,d_N)}(z_1, \ldots, z_N)) &= \frac{1}{|G_{(d_1,\ldots,d_N)}|} \sum_{\pi \in S_N} z_{\pi'\pi(1)}^{d_1} \ldots z_{\pi'\pi(N)}^{d_N} \\ &= \frac{1}{|G_{(d_1,\ldots,d_N)}|} \sum_{\sigma \in S_N} z_{\sigma(1)}^{d_1} \ldots z_{\sigma(N)}^{d_N}, \end{aligned} \tag{3.12}$$

where $\sigma = \pi'\pi$. Since (3.12) is the same as (3.11), the polynomial is symmetric. All monomials in a monomial symmetric polynomial (3.11) have the same order $D(d_1, \ldots, d_N) = d_1 + \ldots + d_N$, so we can talk unequivocally of the order of the polynomial as being the order of its indexing monomial.

For the indexing monomial $(0, \ldots, 0)$, we have $|G_{(0,\ldots,0)}| = |S_N| = N!$ and $\sum_{\pi \in S_N} z_1^0 \ldots z_N^0 = |S_N| = N!$, so, from (3.11), we obtain $s_{(0,\ldots,0)}(z_1, \ldots, z_N) = 1$.

We have mostly considered examples of symmetric polynomials which are also monomial symmetric polynomials, being the only exception the example (3.4), which is the sum of the monomial symmetric polynomials (3.2) and (3.3). In fact, it can be shown that all symmetric polynomials can be written as linear combinations of monomial symmetric polynomials [22]. The power sum symmetric polynomials and the elementary symmetric polynomials are the particular cases of the monomial symmetric polynomials that we use to represent shape.

3.3 Power Sum Symmetric Polynomials

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

$$p_k(z_1, \ldots, z_N) = \sum_{n=1}^{N} z_n^k. \tag{3.13}$$

Among the examples of symmetric polynomials that were given in Section 3.1, (3.2) and (3.3) are power sum symmetric polynomials for $k = 1$ and $k = 3$, respectively.

The power sum symmetric polynomial of order k in N variables is the particular case of a monomial symmetric polynomial with the indexing monomial $(k, 0, \dots, 0)$:

$$p_k(z_1, \dots, z_N) = s_{(k,0,\dots,0)}(z_1, \dots, z_N). \tag{3.14}$$

Note that, for $k = 0$, we obtain $p_0(z_1, \dots, z_N) = s_{(0,\dots,0)}(z_1, \dots, z_N) = 1$.

3.4 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 of the N variables, i.e., the sum of all different monomials of order k involving each of the variables $z_1, \dots, z_N$ zero or one times. Therefore, the sum is composed by N-choose-k monomials. This definition can be formalized as

$$e_k(z_1, \dots, z_N) = \sum_{(i_1,\dots,i_N) \in I_k^N} z_1^{i_1} \dots z_N^{i_N}, \tag{3.15}$$

where I_k^N is the set of the tuples $(i_1, \dots, i_N)$ satisfying $i_1, \dots, i_N \in \{0, 1\}$ and $\sum_{n=1}^N i_n = k$. This set is indexed by the number N of variables and the order k of the polynomial, and its elements encode every possible way of taking k elements from a set of N elements. If $k = 0$, the set I_k^N has exactly one element since there is exactly one way of taking zero elements from a set of N (the only way is by taking none). In this case, we have $e_0(z_1, \dots, z_N) = 1$. If $k > N$, the set I_k^N has no elements (there are no ways of taking more than N elements from a set of N), leading to $e_k(z_1, \dots, z_N) = 0$.

Among the examples of symmetric polynomials that were given in Section 3.1, (3.2) and (3.5) are elementary symmetric polynomials for $k = 1$ and $k = 2$, respectively (the polynomial in (3.2) is, simultaneously, a power sum and an elementary symmetric polynomial).

As it happens in the case of the power sum symmetric polynomials, the elementary symmetric polynomial of order k in N variables is a particular case of a monomial symmetric polynomial where the indexing monomial, in this case, is $(1, \dots, 1, 0, \dots 0)$ (the first k numbers are ones and the remaining $N - k$ numbers are zeros):

$$e_k(z_1, \dots, z_N) = s_{(1,1,\dots,1,0,\dots 0)}(z_1, \dots, z_N). \tag{3.16}$$

3.5 Completeness

The elementary symmetric polynomials in variables the $z_1, \dots, z_N$ are closely related to the coefficients of a monic polynomial with roots $z_1, \dots, z_N$ (monic means that the coefficient of the polynomial of largest order is equal to one). The relation is important because it enables building a complete invariant to the symmetric group by using the elementary symmetric polynomials. Formally, the relation that holds is

$$\prod_{n=1}^N (t - z_n) = \sum_{k=0}^N (-1)^k e_k(z_1, \dots, z_N) t^{N-k}, \tag{3.17}$$

which states that the coefficients of the monic polynomial are exactly the elementary symmetric polynomials apart from an alternating sign pattern.

Expression (3.17) is easily derived by thinking that, when expanding the product, to each term $(t - z_n)$ we can associate a binary variable that encodes the choice between multiplying by t or multiplying by $-z_n$. Let zero denote the first choice and one denote the second choice. As we expand $(t - z_1)\dots(t - z_N)$, we can encode each of the resulting terms as a vector of N of these choices, resulting in a total of 2^N different choice vectors. Each of the terms that involve k one choices has $N - k$ zero choices, resulting in a term that is a product of t^{N-k} and k of the variables $-z_1, \dots, -z_N$. Collecting all the terms with the same order t^{N-k}, we obtain, apart from the sign pattern, exactly those encoded by the elements of the set I_k^N in (3.15).

Since the coefficients of a polynomial uniquely identify its roots $z_1, \dots, z_N$ apart from a permutation, the values of the elementary symmetric polynomials $e_k(z_1, \dots, z_N)$, with $k = 1, \dots, N$, uniquely identify $z_1, \dots, z_N$ apart from a permutation. This is useful for building invariants when we want to consider shapes with unlabeled points (shapes that are induced by the shape-preserving transformations given by the symmetric group acting on $\mathbb{C}^N$ by permuting the coordinates). Figure 3.1 illustrates the invariance and completeness of the elementary symmetric polynomials to permutation.

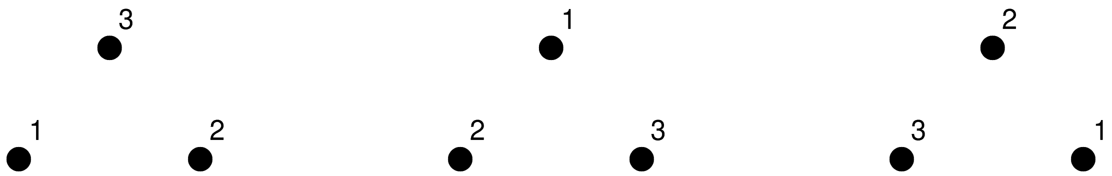

Figure 3.1: Each of the shape representatives in the above figure has points with coordinates -1, 1, and j, differing by permutation of the labels. Nonetheless, they all yield the same result when we evaluate the elementary symmetric polynomials: $e_1(-1, 1, j) = e_1(j, -1, 1) = e_1(1, j, -1) = j$, $e_2(-1, 1, j) = e_2(j, -1, 1) = e_2(1, j, -1) = -1$, and $e_3(-1, 1, j) = e_3(j, -1, 1) = e_3(1, j, -1) = -j$. Furthermore, $-1, 1, j$ is the only unordered set of three points yielding this result.

In a surprising manner, the completeness of elementary symmetric polynomials extends to the power sum symmetric polynomials due to the so-called Newton's identities. These identities (see, e.g., [22]) state that

$$ke_k(z_1, \dots, z_N) + \sum_{r=1}^{k} (-1)^r \, e_{k-r}(z_1, \dots, z_N) \, p_r(z_1, \dots, z_N) = 0, \tag{3.18}$$

which, by straightforward algebraic manipulation, yields

$$e_k(z_1, \dots, z_N) = \frac{1}{k} \sum_{r=1}^{k} (-1)^{r-1} \, e_{k-r}(z_1, \dots, z_N) \, p_r(z_1, \dots, z_N). \tag{3.19}$$

From expression (3.19), it is clear that, if we have the values of the power sum symmetric polynomials $p_k(z_1, \dots, z_N)$, with $k = 1, \dots, N$, we can compute the values of the elementary symmetric polynomials $e_k(z_1, \dots, z_N)$, with $k = 1, \dots, N$, in a recursive way. By rearranging the identities (3.19), we can also compute the values of the power sum symmetric polynomials $p_k(z_1, \dots, z_N)$, with $k = 1, \dots, N$, from the values of the elementary symmetric polynomials $e_k(z_1, \ldots, z_N)$, with $k = 1, \ldots, N$, proving that the power sum symmetric polynomials and the elementary symmetric polynomials are the same in with respect to completeness.

3.6 Homogeneity

Besides the completeness, the other property that will be key for our work is homogeneity. This property is valid for all the symmetric polynomials that only have monomials of a fixed order. This implies that it is valid for the monomial symmetric polynomials and, therefore, for the power sum and elementary symmetric polynomials.

The property states that if we have $z'_n = \alpha z_n$, with $n = 1, \ldots, N$, where α is some constant in $\mathbb{C}$, the following holds:

$$s_{(d_1,\ldots,d_N)}(z'_1, \ldots, z'_N) = \alpha^{D(d_1,\ldots,d_N)} s_{(d_1,\ldots,d_N)}(z_1, \ldots, z_N). \tag{3.20}$$

It can be proved by the following sequence of equalities:

$$\begin{aligned} s_{(d_1,\ldots,d_N)}(z'_1, \ldots, z'_N) &= \frac{1}{|G_{(d_1,\ldots,d_N)}|} \sum_{\pi \in S_N} {z'}_{\pi(1)}^{d_1} \ldots {z'}_{\pi(N)}^{d_N} && (3.21) \\ &= \frac{1}{|G_{(d_1,\ldots,d_N)}|} \sum_{\pi \in S_N} (\alpha z_{\pi(1)})^{d_1} \ldots (\alpha z_{\pi(N)})^{d_N} \\ &= \frac{1}{|G_{(d_1,\ldots,d_N)}|} \sum_{\pi \in S_N} z_{\pi(1)}^{d_1} \ldots z_{\pi(N)}^{d_N} \alpha^{\sum_{n=1}^{N} d_n} && (3.22) \\ &= \alpha^{D(d_1,\ldots,d_N)} \frac{1}{|G_{(d_1,\ldots,d_N)}|} \sum_{\pi \in S_N} z_{\pi(1)}^{d_1} \ldots z_{\pi(N)}^{d_N} && (3.23) \\ &= \alpha^{D(d_1,\ldots,d_N)} s_{(d_1,\ldots,d_N)}(z_1, \ldots, z_N), && (3.24) \end{aligned}$$

where (3.21) is just the definition (3.11), (3.22) is obtained by simple algebraic manipulations, (3.23) uses the definition of the order of a monomial (3.6), and (3.24) uses the definition of the monomial symmetric polynomials (3.11).

The homogeneity property (3.20) states that when we multiply all the variables by a constant, the obtained monomial symmetric polynomials are the original ones multiplied by the same constant raised to the power of the order of the polynomial. The case of interest to our work are constants of the type $\alpha = e^{j\theta}$, with $\theta \in [0, 2\pi)$. Multiplying a point in the complex plane by a factor $e^{j\theta}$ corresponds to rotating counter-clockwise the point around the origin by an angle of θ radians. For $z'_n = e^{j\theta} z_n$, with $n = 1, \ldots, N$, the homogeneity property (3.20) reduces to

$$s_{(d_1,\ldots,d_N)}(z'_1, \ldots, z'_N) = e^{jD(d_1,\ldots,d_N)\theta} s_{(d_1,\ldots,d_N)}(z_1, \ldots, z_N). \tag{3.25}$$

If we consider two sets of points $z_1, \ldots, z_N$ and $z'_1, \ldots, z'_N$, such that $z'_n = z_n e^{j\theta}$, with $n = 1, \ldots, N$, i.e., the points $z'_1, \ldots, z'_N$ are related to $z_1, \ldots, z_N$ by a counter-clockwise rotation of θ radians around the origin of $\mathbb{C}$, the monomial symmetric polynomials evaluated for these two sets of points are related by $s_{(d_1,\ldots,d_N)}(z'_1,\ldots,z'_N) = s_{(d_1,\ldots,d_N)}(z_1,\ldots,z_N)e^{jD(d_1,\ldots,d_N)\theta}$. As we will see in the following chapter, this is similar to the way the Fourier coefficients of a signal change with a time-shift. This similarity enables making a bridge between the symmetric polynomials and spectral methods. Figure 3.2 illustrates the homogeneity for a particular case of monomial symmetric polynomials: the elementary symmetric polynomials.

Figure 3.2: On the left, the shape representative r_s, with points -1, 1, and j. On the right, the shape representative r'_s with points $-j$, j, and -1 (r'_s is obtained by rotating r_s around the origin by $\frac{\pi}{2}$). Evaluating the elementary symmetric polynomials e_1, e_2, and e_3 on the points of the shape representatives yields: for r_s, $e_1(-1,1,j) = j$, $e_2(-1,1,j) = -1$, and $e_3(-1,1,j) = -j$; for r'_s, $e_1(-j,j,-1) = -1$, $e_2(-j,j,-1) = 1$, and $e_3(-j,j,-1) = -1$. The elementary symmetric polynomials for r_s and r'_s are related by the homogeneity property (3.25), which reduces in this case to $e_k(-1,1,j) = e^{jk\frac{\pi}{2}}e_k(-j,j,-1)$, with $k = 1,2,3$.

3.7 Efficient Computation

An apparent problem concerning the usage of the monomial symmetric polynomials in practice is the computation of $s_{(d_1,\ldots,d_N)}(z_1,\ldots,z_N)$. A direct implementation of (3.11) has to sum $\frac{N!}{b_1!\ldots b_P!}$ terms (as discussed in Section 3.2). For the elementary symmetric polynomials, this approach is computationally intractable, even for small N and k, e.g., the evaluation of $e_k(z_1,\ldots,z_N)$ for $N = 50$ and $k = 10$ involves the sum of more than 10 billion different monomials.

Fortunately, dynamic programming provides an efficient solution. To see this, we have to understand how the evaluation of $s_{(d_1,\ldots,d_N)}(z_1,\ldots,z_N)$ decomposes in overlapping subproblems. We first remember the notation $v_1,\ldots,v_P$ and $b_1,\ldots,b_P$, introduced in Section 3.2, where v_n is the n-th of P different values taken by $d_1,\ldots,d_N$ and b_n is the number of occurrences of v_n in $d_1,\ldots,d_N$, with $n = 1,\ldots,P$. For compactness, we now denote the monomial $(d_1,\ldots,d_N)$ by λ and the monomial obtained from $(d_1,\ldots,d_N)$ by removing value v by $\lambda \setminus v$. For example, if we have the monomial $\lambda = (3,2,1,1)$, $\lambda \setminus 2 = (3,1,1)$.

With the introduced notation, definition (3.11) is rewritten as

$$s_\lambda(z_1,\ldots,z_N) = \frac{1}{|G_\lambda|}\sum_{\pi\in S_N} z^{d_1}_{\pi(1)}\ldots z^{d_N}_{\pi(N)}. \tag{3.26}$$

Noting that each element π in S_N maps label N to one of the labels $1,\ldots,N$, we can partition set of permutations S_N into N sets, such that all the permutations in each of these sets map the label N to the same label L, where $L = 1, \dots, N$. This allows rewriting (3.26) as

$$s_\lambda(z_1, \dots, z_N) = \frac{1}{|G_\lambda|} \left(z_N^{d_1} \sum_{\pi \in S_{N-1}} z_{\pi(1)}^{d_2} \dots z_{\pi(N-1)}^{d_N} + \dots + z_N^{d_N} \sum_{\pi \in S_{N-1}} z_{\pi(1)}^{d_1} \dots z_{\pi(N-1)}^{d_{N-1}} \right) \tag{3.27}$$

$$= \frac{1}{|G_\lambda|} \left(z_N^{d_1} |G_{\lambda \setminus d_1}| s_{\lambda \setminus d_1}(z_1, \dots, z_{N-1}) + \dots + z_N^{d_N} |G_{\lambda \setminus d_N}| s_{\lambda \setminus d_N}(z_1, \dots, z_{N-1}) \right), \tag{3.28}$$

where the last equality is obtained by using (3.26) again. Since $d_1, \dots, d_N$ take only P different values, rearranging the terms in (3.28), we obtain

$$s_\lambda(z_1, \dots, z_N) = \frac{1}{|G_\lambda|} \left(b_1 z_N^{v_1} |G_{\lambda \setminus v_1}| s_{\lambda \setminus v_1}(z_1, \dots, z_{N-1}) + \dots + b_P z_N^{v_P} |G_{\lambda \setminus v_P}| s_{\lambda \setminus v_P}(z_1, \dots, z_{N-1}) \right). \tag{3.29}$$

Now, looking at the orders of the stabilizers in (3.29) , we have $|G_\lambda| = b_1! \dots, b_P!$ (see Section 3.2) and $|G_{\lambda \setminus v_n}| = b_1! \dots, b_{n-1}!(b_n - 1)! b_{n+1}! \dots b_P!$ (justified by minus one occurrence of v_n in the monomial $\lambda \setminus v_n$, with $n = 1, \dots, P$). This implies that

$$|G_\lambda| = b_1! \dots, b_P! = b_n b_1! \dots, b_{n-1}!(b_n - 1)! b_{n+1}! \dots b_P! = b_n |G_{\lambda \setminus v_n}|, \tag{3.30}$$

allowing the simplification of (3.29), resulting in

$$s_\lambda(z_1, \dots, z_N) = z_N^{v_1} s_{\lambda \setminus v_1}(z_1, \dots, z_{N-1}) + \dots + z_N^{v_P} s_{\lambda \setminus v_P}(z_1, \dots, z_{N-1}). \tag{3.31}$$

The recurrence relation (3.29) decomposes the evaluation of $s_\lambda(z_1, \dots, z_N)$ into P subproblems: solving for $s_{\lambda \setminus v_n}(z_1, \dots, z_{N-1})$, with $n = 1, \dots, P$, and then combining the results as in (3.31). The computational gain comes from storing the results and using them when the subproblems reappear. Note that the choice of z_N for the partition condition of S_N in (3.27) was arbitrary. Partitioning the set of permutations S_N using an arbitrary label L, where $L = 1, \dots, N$, is equally valid (we used $L = N$ in (3.27) and (3.28)), yielding the decomposition

$$s_\lambda(z_1, \dots, z_N) = z_L^{v_1} s_{\lambda \setminus v_1}(z_1, \dots, z_{L-1}, z_{L+1}, \dots, z_N) + \dots + z_L^{v_P} s_{\lambda \setminus v_P}(z_1, \dots, z_{L-1}, z_{L+1}, \dots, z_N), \tag{3.32}$$

where $z_1, \dots, z_{L-1}, z_{L+1}, \dots, z_N$ equals $z_2, \dots, z_N$ and $z_1, \dots, z_{N-1}$ for $L = 1$ and $L = N$, respectively.

We now particularize the decomposition (3.31) for the case of the power sum symmetric polynomials and elementary symmetric polynomials. For the power sum symmetric polynomials, decomposition (3.31) is written as

$$\begin{aligned} p_k(z_1, \dots, z_N) &= s_{(k,0,\dots,0)}(z_1, \dots, z_N) \\ &= z_N^k s_{(0,\dots,0)}(z_1, \dots, z_{N-1}) + z_N^0 s_{(k,0,\dots,0)}(z_1, \dots, z_{N-1}) \\ &= z_N^k + p_k(z_1, \dots, z_{N-1}), \end{aligned} \tag{3.33}$$

with $k > 0$, where equality (3.33) is obtained by using the equalities $s_{(0,\dots,0)}(z_1, \dots, z_N) = 1$ and

$s_{(k,\ldots,0)}(z_1,\ldots,z_N) = p_k(z_1,\ldots,z_N)$, for $N > 0$ (see Section 3.2). Expression (3.33) is uninteresting in terms of computational efficiency because all the $p_k(z_1,\ldots,z_n)$, with $k = 1,\ldots,N-1$, computed during the evaluation of $p_N(z_1,\ldots,z_N)$, appear only once (there is no gain in storing them for later).

For the elementary symmetric polynomials, decomposition (3.31) leads to

$$\begin{aligned} e_k(z_1,\ldots,z_N) &= s_{(1,\ldots,1,0,\ldots0)}(z_1,\ldots,z_N) \\ &= z_N^1 s_{(1,\ldots,1,0,\ldots0)}(z_1,\ldots,z_{N-1}) + z_N^0 s_{(1,\ldots,1,0,\ldots0)}(z_1,\ldots,z_{N-1}) \\ &= z_N e_{k-1}(z_1,\ldots,z_{N-1}) + e_k(z_1,\ldots,z_{N-1}), \end{aligned} \tag{3.34}$$

with $k > 0$, where the last equality is obtained by using the definition of the elementary symmetric polynomials as a monomial symmetric polynomial (see Section 3.4). For the elementary symmetric polynomials, the use of decomposition (3.34) yields an enormous computational gain when compared to the direct use of the definition (3.15).

To provide intuition, we use the two-dimensional array depicted in Figure 3.3. It is indexed by the natural numbers, where the two dimensions i and j are identified with N and k, respectively. Entry (i,j) stores the result of $e_j(z_1,\ldots,z_i)$. Column $j = 0$ is not included because $e_0(z_1,\ldots,z_i) = 1$, for all $i \in \mathbb{N}$. Furthermore, since $e_k(z_1,\ldots,z_N) = 0$, for $k > N$, we just have to consider entries (i,j) with $j \leq i$. The computation of $e_j(z_1,\ldots,z_i)$ according to the decomposition (3.34) can be performed in constant time if entries $(i-1,j-1)$ and $(i-1,j)$ have been previously computed. If not, decomposition (3.34) has to be recursively used until we reach a position (i',j') where we can evaluate (3.34) directly from the array (which eventually happens because $e_0(z_1,\ldots,z_N) = 1$, for all N, and $e_k(z_1,\ldots,z_N) = 0$, for $k > N$). If the array is initially empty, i.e., no results have yet been computed, the computation of $e_k(z_1,\ldots,z_N)$ requires computing $k(N-k+1)$ positions. If we want to evaluate $e_k(z_1,\ldots,z_N)$, for $k = 1,\ldots,N$, an additional computational gain comes from the fact that the required entries of the array just have to be computed once and are reused many times. In this case, we have a total of $\frac{N(N+1)}{2}$ entries involved.

To contrast the computational cost of evaluating $e_k(z_1,\ldots,z_N)$ by the definition (3.15) with the one of using decomposition (3.34) with techniques of dynamic programming, we recall the example of the beginning of this section: for $N = 50$ and $k = 10$, using definition (3.15) requires the computation of more than 10 billion terms while using the techniques proposed in this section only requires the computation of 410 terms (assuming that the array is initially empty, this number is $k(N-k+1)$).

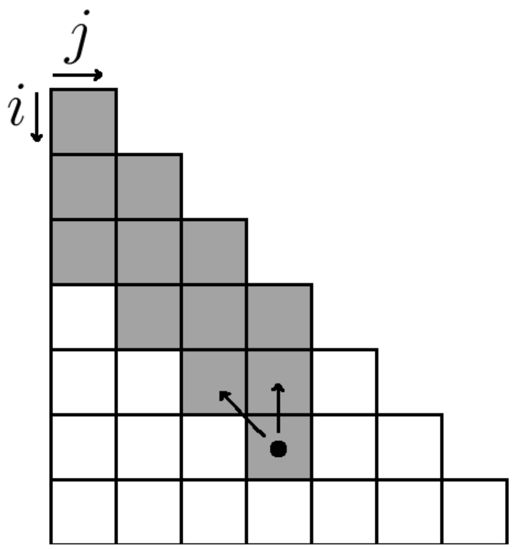

Figure 3.3: Entry (i, j) of the array stores $e_j(z_1, \ldots, z_i)$. If the array is initially empty, the computation of $e_4(z_1, \ldots, z_6)$ through decomposition (3.34) uses the 12 positions marked in gray (the arrows from entry $(6, 4)$ to entries $(5, 3)$ and $(5, 4)$ represent the dependency of the computation of $e_4(z_1, \ldots, z_6)$ on the values of those entries, $e_3(z_1, \ldots, z_5)$ and $e_4(z_1, \ldots, z_5)$).

Chapter 4

Spectral Invariants

The chapter discusses the problem of representing a continuous-time complex-valued periodic signal apart from a time-shift. We first introduce the Fourier series coefficients, which uniquely identify the signal under mild technical conditions, and study how they change by shifting the signal. We then present shift invariants that have been studied in the signal processing literature: the power spectrum and the bispectrum. We verify invariance and discuss completeness. The spectral invariants presented in this chapter and the permutation invariants of Chapter 3 will be used in Chapter 5 to build a shape representation that is invariant and complete with respect to permutations of the labels and geometric transformations. Finally, we briefly discuss the broader family of higher-order spectral invariants, to which the power spectrum and the bispectrum belong.

4.1 Fourier Series

A continuous-time complex-valued periodic signal x of period T is, under mild technical conditions, uniquely determined by the coefficients $c_k(x)$, with $k \in \mathbb{Z}$, of its Fourier series:

$$x(t) = \sum_{k=-\infty}^{+\infty} c_k(x) e^{-j\frac{2\pi}{T}kt}. \tag{4.1}$$

The coefficients of the Fourier series are given in terms of x by

$$c_k(x) = \frac{1}{T}\int_T x(t) e^{-j\frac{2\pi}{T}kt} dt. \tag{4.2}$$

The Fourier series has several properties; the one that impacts our work is the behavior of the coefficients with a time-shift of the signal x. If a signal x' is a shifted version of a signal x, i.e., if $x'(t) = x(t + t_0)$, the coefficients of the Fourier series of x' are related to the ones of the Fourier series of x:

$$
\begin{aligned}
c_k(x') = &= \frac{1}{T}\int_T x'(t)e^{-j\frac{2\pi}{T}kt}dt \\
&= \frac{1}{T}\int_T x(t+t_0)e^{-j\frac{2\pi}{T}kt}dt \\
&= \frac{1}{T}\int_T x(t')e^{-j\frac{2\pi}{T}k(t'-t_0)}dt' && (4.3)\\
&= e^{j\frac{2\pi}{T}kt_0}c_k(x). && (4.4)
\end{aligned}
$$

Equality (4.3) is obtained by making the change of variables $t' = t + t_0$ and (4.4) uses the fact that $e^{j\frac{2\pi}{T}kt_0}$ does not depend on t'. Expression (4.4) is surprisingly similar to expression (3.25) obtained in the previous chapter. In both cases (rotating a set of planar points and shifting a periodic signal), the change in the representation (the symmetric polynomials and the coefficients of the Fourier series) is just a phase difference that is proportional to the order.

4.2 Power Spectrum

A way to represent a continuous-time complex-valued periodic signal x apart from a time shift is to factor out the corresponding phase difference induced on the coefficients of the Fourier series. A complete invariant representation requires that two signals have the same representation if and only if they are related by a shift. The power spectrum $P_k(x)$ of a continuous-time complex-valued periodic signal x is the squared absolute value of its coefficients of the Fourier series, i.e.,

$$
P_k(x) = |c_k(x)|^2 = c_k(x)c_k(x)^*, \tag{4.5}
$$

where $*$ denotes complex conjugation.

The invariance of the power spectrum with respect to signal shifts is easily verified from (4.5). If $x'(t) = x(t + t_0)$, the power spectrum of x' is

$$
\begin{aligned}
P_k(x') &= c_k(x')c_k(x')^* \\
&= c_k(x)e^{j2\pi k\frac{t_0}{T}}c_k(x)^*e^{-j2\pi k\frac{t_0}{T}} && (4.6)\\
&= c_k(x)c_k(x)^* \\
&= P_k(x), && (4.7)
\end{aligned}
$$

where equality (4.6) is obtained by using property (4.4).

Unfortunately, the power spectrum is not complete: it keeps all the information about the power of each frequency component in the signal (motivating the designation of power spectrum), but no information about the relative phases between these components. Shifting a signal x of period T with coefficients of the Fourier series $c_k(x)$ by t_0 results in the multiplication of each coefficient $c_k(x)$ by $e^{j\frac{2\pi}{T}kt_0}$. Only the joint multiplication of each of the coefficients $c_k(x)$ by $e^{jk\theta}$ results in a shift of the signal x. If this condition is not met, the resulting signal is not a shifted version of the original signal. Thus, the power spectrum is not complete because any two signal x and x' that have the same power at all the frequencies (i.e., $|c_k(x)| = |c_k(x')|$, for all $k \in \mathbb{Z}$), irrespective of their temporal cohesion, have the same power spectrum. This means that, for a signal x, each of the coefficients of the Fourier series $c_k(x)$ can be multiplied by a term $e^{j\theta_k}$, with an arbitrary $\theta_k \in [0, 2\pi)$ for each $k \in \mathbb{Z}$, and still leave the power spectrum unchanged. This is equivalent to shifting each of the frequency components of x independently. Figure 4.1 illustrates the invariance of the power spectrum with respect to signal shifts, but also its incompleteness.

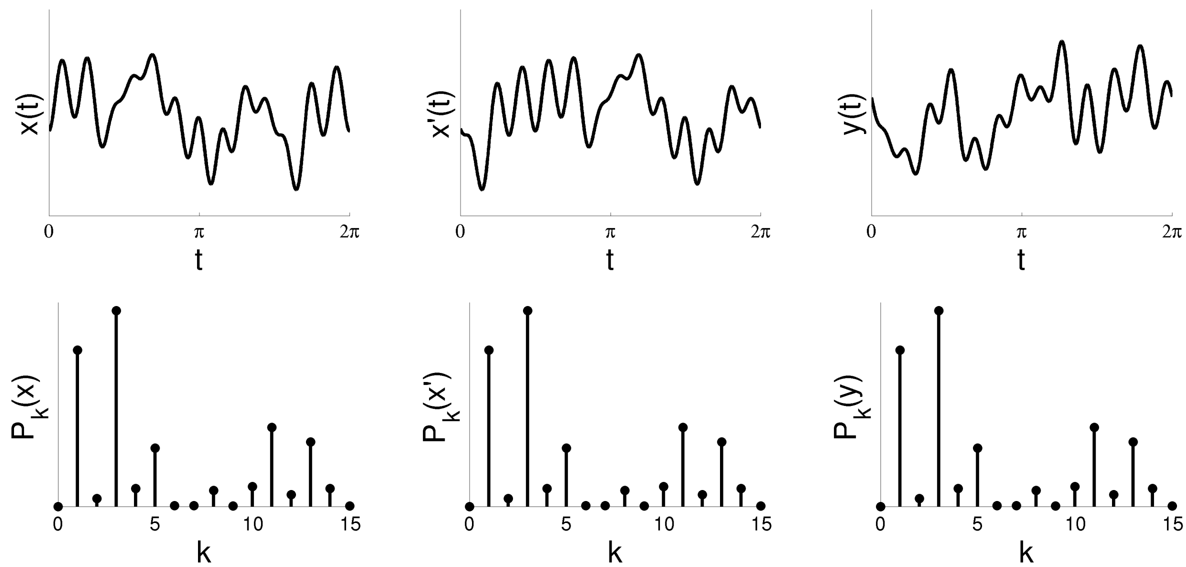

Figure 4.1: On top, three real-valued periodic signals, x, x', and y. On bottom, the corresponding power spectra, $P_k(x)$, $P_k(x')$, and $P_k(y)$. Signals x and x' only differ by a time shift and their power spectra are the same, illustrating its invariance. However, the power spectra of y, which is not a shifted version of x and x', is also the same, illustrating its incompleteness.

4.3 Bispectrum

The bispectrum $b_{k_1,k_2}(x)$ of a continuous-time complex-valued periodic signal x is defined as

$$b_{k_1,k_2}(x) = c_{k_1}(x)c_{k_2}(x)c_{k_1+k_2}(x)^*, \tag{4.8}$$

where $k_1, k_2 \in \mathbb{Z}$. The bispectrum is symmetric with respect to k_1 and k_2, i.e., $b_{k_1,k_2}(x) = b_{k_2,k_1}(x)$. If we think of the bispectrum as an infinite grid, the main diagonal is the symmetry axis of $b_{k_1,k_2}(x)$.

As for the the power spectrum, the invariance of the bispectrum with respect to signal shifts is readily verified from its definition (4.8). Let x' be a shifted by t_0 version of x, i.e., $x'(t) = x(t+t_0)$. The bispectrum of x' is

$$\begin{aligned} b_{k_1,k_2}(x') &= c_{k_1}(x')c_{k_2}(x')c_{k_1+k_2}(x')^* \\ &= c_{k_1}(x)e^{j\frac{2\pi}{T}k_1t_0}c_{k_2}(x)e^{j\frac{2\pi}{T}k_2t_0}c_{k_1+k_2}(x)^*e^{-j\frac{2\pi}{T}(k_1+k_2)t_0} \qquad (4.9) \\ &= c_{k_1}(x)c_{k_2}(x)c_{k_1+k_2}(x)^* \\ &= b_{k_1,k_2}(x), \qquad (4.10) \end{aligned}$$

where equality (4.9) is obtained by using property (4.4).

The bispectrum, contrary to the power spectrum, preserves the phase information of the signal apart from an arbitrary shift and, therefore, it is complete (under mild conditions; see references [23, 24]). There are reconstruction algorithms that recover the signal apart from an arbitrary shift (see reference [25]). The completeness comes at the price of an higher dimensional representation for the signal x: while the power spectrum is linear in the number of coefficients of the Fourier series $c_k(x)$, the bispectrum is quadratic. Nonetheless, the symmetry along the main diagonal allows us to keep just the half of the bispectrum $b_{k_1,k_2}(x)$ with $k_1 \geq k_2$. Furthermore, if the signal x has nonzero coefficients of the Fourier series $c_k(x)$ only for $k = 1, \ldots, K$, we just have to keep the coefficients of the bispectrum $b_{k_1,k_2}(x)$, with $k_1 \geq k_2$, $k_1 \geq 1$, and $k_1 + k_2 \leq N$. See Figure 4.2 for an illustration of the invariance and completeness properties of the bispectrum. The bispectra of the signal in Figure 4.2 are upper triangular because the signals represented only has nonzero coefficients of Fourier series for $k = 1, \ldots, 15$. For more information about the bispectrum see references [26, 13, 27, 28, 29, 30]).

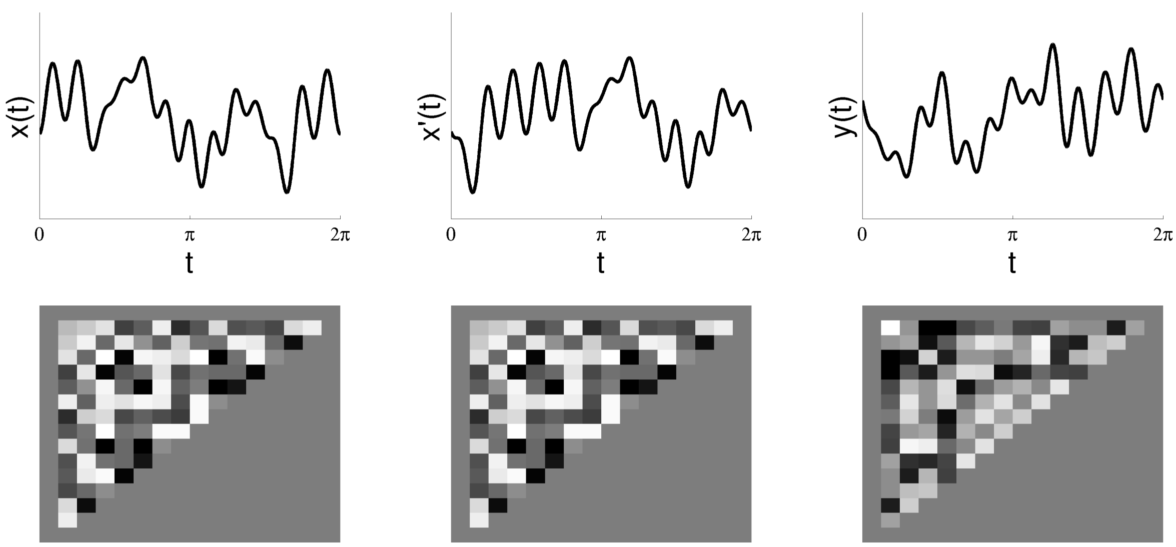

Figure 4.2: On top, three real-valued periodic signals, x, x', and y. On bottom, the corresponding phase of the bispectra $\arg b_{k_1,k_2}(x)$, $\arg b_{k_1,k_2}(x')$, and $\arg b_{k_1,k_2}(y)$. Signals x and x' only differ by a time shift and their bispectra are the same, illustrating its invariance. Contrary to the power spectrum (see Figure 4.1), the bispectrum of the signal y is different from the one of the signals x and x', illustrating its completeness.

4.4 Higher-Order Spectra

The power spectrum and the bispectrum are members of a larger family of invariants called higher-order spectra. The spectrum of order l of a continuous-time complex-valued periodic signal x of period T with coefficients of the Fourier series $c_k(x)$, with $k \in \mathbb{Z}$ is defined as

$$h_{k_1,\ldots,k_l}(x) = c_{k_1+\ldots+k_l}(x)^* \prod_{n=1}^{l} c_{k_n}(x), \tag{4.11}$$

where $l \in \mathbb{N}$ and $k_1, \ldots, k_l \in \mathbb{Z}$. For $l = 1, 2$, (4.11) is, respectively, the power spectrum and the bispectrum.

All the elements in the higher-order spectra family are shift-invariant. The process of verifying invariance is similar to what was done for the power spectrum and the bispectrum.

The autocorrelation function of order l of a signal x of period T is

$$a_{t_1,\ldots,t_l}(x) = \int_T x(t) \prod_{n=1}^{l} x(t+t_n) dt, \tag{4.12}$$

where $t_1, \ldots, t_l \in \mathbb{R}$. The higher-order spectra arise as the coefficients of the higher-dimensional Fourier series of the higher-order autocorrelation functions, i.e.,

$$h_{k_1,k_2,\ldots,k_l}(x) = \mathcal{F}\{a_{t_1,\ldots,t_l}(x)\}_{k_1,\ldots,k_l}, \tag{4.13}$$

where $\mathcal{F}\{a_{t_1,\ldots,t_l}(x)\}_{k_1,\ldots,k_l}$ denotes the coefficient $k_1, \ldots, k_l$ of the l-dimensional Fourier series of the higher-order autocorrelation $a_{t_1,\ldots,t_l}(x)$ (see reference [23]).

In this short remark, we only meant to put the power spectrum and the bispectrum in perspective. See references [31, 32, 33, 34] for more information on higher-order spectra.

Chapter 5

Proposed Representation

In this chapter, we build a shape representation that is complete and invariant to translation, rotation, scaling and permutation. We present the group and the group action that encode these transformations and factor them out in turn. The invariants introduced in Chapter 3 and Chapter 4 are used to deal with permutation and rotation. We show the invariance and completeness properties of the final shape representation proposed.

5.1 Group Action

The shape-preserving transformations that we consider are translation, scaling, rotation, and permutation of the labels. The group G that encodes these transformations is constructed as the product of the symmetric group S_N, the group of complex numbers under addition $(\mathbb{C},+)$, and the group of nonzero complex numbers under multiplication $(\mathbb{C},\cdot)$. The nonzero complex numbers under multiplication $(\mathbb{C},\cdot)$ can be further decomposed as the product of the group of positive real numbers under multiplication $(R^+,\cdot)$ and the group of complex numbers with absolute value one under multiplication $SO(2)$. We write the decomposition of G as a product of its component groups as

$$G = S_N \times (\mathbb{C},+) \times (\mathbb{C},\cdot) \times (R^+,\cdot) \times SO(2). \tag{5.1}$$

The group operation of G is the one that arises from the product construction (see Section 2.2). We denote an element of G by g or, when we want to make evident each of the component elements of the product, by (π, z^t, α, z^r), where each of the component elements belongs to the corresponding component group of G as in decomposition (5.1).

The symmetric group S_N accounts for permutation of the labels; the complex numbers under addition $(\mathbb{C},+)$ for translation; the complex numbers with absolute value one under multiplication $SO(2)$ for rotation (the elements of $SO(2)$ are of the form $e^{j\theta}$, with $\theta \in [0, 2\pi)$); and the positive real numbers under multiplication $(R^+,\cdot)$ for scaling. The desired group action on the space of shape representatives $\mathbb{C}^N$ is

$$\phi(g, r_s) = \phi((\pi, z^t, \alpha, z^r), r_s) = \alpha z^r (\pi(r_s) - \bar{r}_s) + \bar{r}_s + z^t, \tag{5.2}$$

where $\bar{r}_s$ is the centroid of the shape representative r_s (see definition (2.1)), i.e., $\bar{r}_s = \frac{1}{N}\sum_{n=1}^{N} z_n$. From now on, we will use the bar notation to denote the mean of a vector quantity. There is some abuse of notation in (5.2) since we sum vectors with scalars. What is meant is that the scalar is summed coordinate-wise to the vector. The action of π on r_s is denoted by $\pi(r_s)$. This is the usual coordinate label permutation induced by the an element π of the symmetric group S_N.

In group action (5.2), the term $\alpha z^r(\pi(r_s) - \bar{r}_s)$ has zero mean: the group action first centers the representative r_s at the origin and only then rotates and scales it. After that, r_s is recentered at the original position plus the translation z^t. The term $\bar{r}_s + z^t$ is the mean of the group action $\phi(g, r_s)$, i.e., $\bar{\phi}(g, r_s) = \bar{r}_s + z^t$. Using this last remark about the mean, we can rewrite (5.2) as

$$\phi(g, r_s) = \alpha z^r(\pi(r_s) - \bar{r}_s) + \bar{\phi}(g, r_s), \tag{5.3}$$

or, alternatively, as

$$\phi(g, r_s) - \bar{\phi}(g, r_s) = \alpha z^r(\pi(r_s) - \bar{r}_s). \tag{5.4}$$

The commutativity of permutation with scaling and translation allows the following rearrangements of (5.4):

$$\phi(g, r_s) - \bar{\phi}(g, r_s) = \alpha z^r(\pi(r_s) - \bar{r}_s) = \alpha z^r(\pi(r_s - \bar{r}_s)) = \pi(\alpha z^r(r_s - \bar{r}_s)). \tag{5.5}$$

We now prove that (5.2) defines a valid group action for G (see Section 2.2). The identity property is obviously verified. As for the homomorphism property, we have

$$\begin{aligned}
\phi(g_1, \phi(g_2, r_s)) &= \alpha_1 z_1^r(\pi_1(\phi(g_2, r_s)) - \bar{\phi}(g_2, r_s)) + \bar{\phi}(g_2, r_s) + z_1^t && (5.6)\\
&= \alpha_1 z_1^r(\pi_1(\phi(g_2, r_s) - \bar{\phi}(g_2, r_s))) + \bar{\phi}(g_2, r_s) + z_1^t && (5.7)\\
&= \alpha_1 z_1^r(\pi_1(\alpha_2 z_2^r \pi_2(r_s - \bar{r}_s)) + \bar{\phi}(g_2, r_s) + z_1^t && (5.8)\\
&= \alpha_1\alpha_2 z_1^r z_2^r(\pi_1(\pi_2(r_s - \bar{r}_s)) + \bar{r}_s + z_1^t + z_2^t && (5.9)\\
&= \phi((\pi_1\pi_2, z_1^t z_2^t, \alpha_1\alpha_2, z_1^r z_2^r), r_s)) && (5.10)\\
&= \phi(g_1 g_2, r_s), && (5.11)
\end{aligned}$$

where $\bar{\phi}(g_2, r_s) = \bar{r}_s + z_2^t$. To derive equalities (5.6), (5.10), and (5.11), we use the definition of the group action (5.2). To derive (5.7) and (5.9), we use the commutativity properties of permutation expressed in (5.5). To derive (5.8), we use (5.4).

In the following sections, we build a complete invariant shape representation by factoring out each of components of the group action (5.2).

5.2 Translation Invariance

Translation can be factored out by centering the representative at the origin. The translation invariant is denoted by $\rho^t : \mathbb{C}^N \to \mathbb{C}^N$. It is evaluated for a shape representative r_s as

$$\rho^t(r_s) = r_s - \bar{r}_s, \tag{5.12}$$

where $\bar{r}_s$ is the centroid of the representative r_s. We use similar notation for the other invariants, i.e., the usage of the first letter of a transformation as a superscript means that the map is invariant to that transformation.

It is straightforward to verify the invariance of ρ^t to translation:

$$\begin{aligned}\rho^t(\phi(g, r_s)) &= \phi(g, r_s) - \bar{\phi}(g, r_s) \\ &= \alpha z^r(\pi(r_s) - \bar{r}_s),\end{aligned} \tag{5.13}$$

where equality (5.13) comes from the group action rearrangement (5.3). Since (5.13) does not depend on the translation component z^t of the action (5.2), we conclude that ρ^t is invariant to translation.

The completeness of ρ^t is proved by noting that two shape representatives r_s and r'_s have the same translation invariant representation, i.e., $\rho^t(r_s) = \rho^t(r'_s)$, if and only if $r'_s = r_s + z^t$ for some translation z^t in $\mathbb{C}$, which is obvious from the definition (5.12).

5.3 Scaling Invariance

To factor out the scaling part of the action (5.2), we normalize the mean energy $e : \mathbb{C}^N \to \mathbb{R}$, which is given by

$$e(r_s) = \sqrt{\frac{1}{N} \sum_{n=1}^{N} |z_n|^2}. \tag{5.14}$$

The scaling invariant $\rho^s : \mathbb{C}^N \to \mathbb{C}^N$ is given by

$$\rho^s(r_s) = \begin{cases} \frac{1}{e(r_s)}(r_s - \bar{r}_s) + \bar{r}_s & \text{if } r_s \neq 0; \\ 0 & \text{otherwise.} \end{cases} \tag{5.15}$$

From now on, we will assume that $r_s \neq 0$. The shape representative r_s only equals zero when all the points $z_1, \ldots, z_N$ are zero, which is not important in practice.

In (5.15), we use the mean energy instead of the total energy $e_T : \mathbb{C}^N \to \mathbb{R}$, which is given by

$$e_T(r_s) = \sqrt{\sum_{n=1}^{N} |z_n|^2}, \tag{5.16}$$

to provide some constancy to the scaling invariant ρ^s when the number of points of the shapes change.

To gain intuition in this respect, assume that we have a shape representative r_s with N points. It has total energy $e_T(r_s)$ and mean energy $e(r_s)$. If we now consider shapes of $2N$ points and generate a shape representative r'_s by simply repeating twice each of the points in r_s. (This is for illustrative purposes only. It does not happen in practice, although points may indeed be close to each other.) This new shape representative r'_s has total energy

$$e_T(r'_s) = \sqrt{\sum_{n=1}^{2N} |z'_n|^2} = \sqrt{2\sum_{n=1}^{N} |z_n|^2} = \sqrt{2}e_T(r_s), \tag{5.17}$$

i.e., the energy $e_T(r'_s)$ changes when we change the number of points of the shapes. Nonetheless, the mean energy remains constant:

$$e(r'_s) = \sqrt{\frac{1}{2N}\sum_{n=1}^{2N} |z'_n|^2} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} |z_n|^2} = e(r_s). \tag{5.18}$$

By normalizing with the mean energy (5.14), if we plot the two normalized shape representatives $\rho^s(r_s)$ and $\rho^s(r'_s)$ in the complex plane, we see that the points of the normalized shape representatives $\rho^s(r_s)$ and $\rho^s(r'_s)$ overlap. This would not happen if we have used the total energy.

A simultaneous invariant to translation and scaling is obtained by evaluating

$$\rho^{s,t} = \rho^s \circ \rho^t. \tag{5.19}$$

To verify the invariance of $\rho^{s,t}$ to translation and scaling, we write

$$\begin{aligned}
\rho^{s,t}(\phi(g, r_s)) &= \rho^s(\rho^t(\phi(g, r_s))) \\
&= \rho^s(\alpha z^r(\pi(r_s) - \bar{r}_s)) && (5.20) \\
&= \frac{1}{e(\alpha z^r(\pi(r_s) - \bar{r}_s))}\alpha z^r(\pi(r_s) - \bar{r}_s) && (5.21) \\
&= \frac{z^r}{e(r_s - \bar{r}_s)}(\pi(r_s) - \bar{r}_s), && (5.22)
\end{aligned}$$

where (5.20) comes from (5.13). Equality (5.21) comes from (5.15) along with the fact that $\alpha z^r(\pi(r_s) - \bar{r}_s)$ has zero mean. To obtain (5.22) we used the fact that the mean energy $e(\alpha z^r(\pi(r_s) - \bar{r}_s))$ is linear in α and does not depend on π and z^r, which can be easily derived from the definition (5.14). Equality (5.22) proves that $\rho^{s,t}$ is invariant to both translation and scaling because it does not depend on z^t and α, the corresponding components of the action (5.2).

The completeness of the invariant $\rho^{s,t}$ is a result of the fact that two shape representatives r_s and r'_s only have the same translation and scaling invariant representation (i.e. $\rho^{s,t}(r_s) = \rho^{s,t}(r'_s)$) if and only if $r'_s = \alpha r_s + z^t$ for some scale factor α in the positive real numbers R^+ and some translation in the complex numbers $\mathbb{C}$, which, again, is obvious from the normalizations (5.12) and (5.15).

The construction of invariants through composition of smaller invariants, as we have done with $\rho^{s,t}$ in (5.19), will be a recurring practice in the remaining of this chapter.

5.4 Permutation Invariance

The factorization of permutation part of the action (5.2) uses the elementary symmetric polynomials presented in Chapter 3. The permutation invariant representation $\rho^p : \mathbb{C}^N \to \mathbb{C}^N$ is defined as

$$\rho^p(r_s) = \begin{bmatrix} e_1(r_s) \\ \vdots \\ e_N(r_s) \end{bmatrix}, \tag{5.23}$$

where $e_1(r_s), \ldots, e_N(r_s)$ are the elementary symmetric polynomial evaluated at the points $z_1, \ldots, z_N$ of the shape representative r_s. As shown in Section 3.5, the ordered set of values of the elementary symmetric polynomials $e_1(r_s), \ldots, e_N(r_s)$ uniquely determine r_s up to an arbitrary permutation of its coordinates $z_1, \ldots, z_N$.

The invariant $\rho^{p,s,t}$ is given by

$$\rho^{p,s,t} = \rho^p \circ \rho^s \circ \rho^t. \tag{5.24}$$

To verify that $\rho^{p,s,t}$ is invariant to translation, scaling, and permutation, we write

$$\begin{aligned}
\rho^{p,s,t}(\phi(g, r_s)) &= \rho^p(\rho^{s,t}(\phi(g, r_s))) && (5.25)\\
&= \rho^p \left(\frac{z^r}{e(r_s - \bar{r}_s)} (\pi(r_s) - \bar{r}_s) \right) && (5.26)\\
&= \rho^p \left(\pi \left(\frac{z^r}{e(r_s - \bar{r}_s)} (r_s - \bar{r}_s \right) \right) && (5.27)\\
&= \rho^p \left(\frac{z^r}{e(r_s - \bar{r}_s)} (r_s - \bar{r}_s) \right). && (5.28)
\end{aligned}$$

Equalities (5.25) and (5.26) follow from definitions (5.24) and (5.19), respectively. To derive (5.27), we used the fact that permutation commutes with translation and scaling. Equality (5.28) follows from the invariance to permutation of ρ^p in (5.23). Equality (5.28) shows that $\rho^{p,s,t}$ is invariant to permutation, scaling, and translation. The only component of the group action (5.2) left to factor out is the rotation component z^r.

The completeness of invariant $\rho^{p,s,t}$ comes from the completeness of $\rho^{s,t}$ with respect to scaling and translation (Section 5.3), and the completeness of ρ^p with respect to permutation (Section 3.5). Note that the values $e_1(r_s), \ldots, e_N(r_s)$ of the elementary symmetric polynomials in (5.23) can be replaced by the values $p_1(r_s), \ldots, p_N(r_s)$ of the power sum symmetric polynomials without loss of completeness and invariance, as shown in Section 3.5. In Appendix B we study the behavior of the elementary symmetric polynomials and the power sum symmetric polynomials when their variables are perturbed. For what is required to deal with rotation in the next section, each elementary symmetric polynomial e_k can be substituted by any monomial symmetric polynomial as long as the order of the indexing monomial is equal to k (see Section 3.6). Therefore, an alternative permutation invariant $\rho'^p : \mathbb{C}^N \to \mathbb{C}^N$ is

$$\rho'^p(r_s) = \begin{bmatrix} s_{\lambda_1}(r_s) \\ \vdots \\ s_{\lambda_N}(r_s) \end{bmatrix}, \tag{5.29}$$

where λ_k is an indexing monomial of order k, i.e., $D(\lambda_k) = k$, with $k = 1, \ldots, N$. However, for the invariant ρ'^p in (5.29), we are not aware that any completeness properties can be proved.

Notice that translation invariance imposes $\bar{r}_s = 0$ and $e_1(r_s) = N\bar{r}_s = 0$. We can thus remove e_1 from the permutation invariant ρ^p since it does not contain any shape information. The same happens if we use power sum symmetric polynomials or, in fact, any monomial symmetric polynomial of order 1 since the only indexing monomial of order 1 is $(1, 0, \ldots, 0)$. Therefore, representation $\rho^{p,s,t}(r_s)$ has $N-1$ nontrivial numbers, i.e., $\rho^{p,s,t} : \mathbb{C}^N \to \mathbb{C}^{N-1}$.

To make the permutation invariant ρ^p insensitive to the point density of the shapes, we divide the elementary symmetric polynomial e_k by the number of monomials summed, *i.e.*, N-choose-k. (This normalization was left out of (5.23) to avoid unnecessary clutter.) If we had used the power sum symmetric polynomials, we would have to divide by N. Moreover, in the case of a generic monomial symmetric polynomial we would have to divide by $\frac{N!}{b_1! \ldots b_N!}$ (see Section 3.2). Naturally, this normalization does not affect the completeness of the permutation invariant.

Making the invariants independent to the number of points of the shapes is interesting because, in practice, we may have shapes with different numbers of points. Obviously the completeness does not hold in this case but, since the representation does not depend on the number of points, we can represent shape by using the invariants and still expect good results for shapes that are similar but have different number of points.

5.5 Rotation Invariance

The factorization of rotation in this section is linked to the usage of the permutation invariant ρ^p of the previous section. Notice that this did not happen when we considered other the invariants. We could just have, for example, translation and permutation invariance by using $\rho^{p,t} = \rho^p \circ \rho^t$, as presented in Section 5.2 and Section 5.4.

By the homogeneity property (3.25), we know that the elementary symmetric polynomials e_k change with shape rotation (see Section 3.6) in a way that resembles the change in the Fourier series coefficients with a signal shift (see Section 4.1) (Remember from the homogeneity property (3.25) that, for $r'_s = r_s e^{j\theta}$, $e_k(r'_s) = e_k(r_r)e^{jk\theta}$, where $k = 1, \ldots, N$ and $\theta \in [0, 2\pi))$. We can then interpret the values of the elementary symmetric polynomials $e_k(r_s)$ as being the coefficients of the Fourier series of a continuous-time complex-valued periodic signal x of period T,

$$c_k(x) = \begin{cases} e_k(r_s) & \text{if } 1 \le k \le N; \\ 0 & \text{otherwise.} \end{cases} \tag{5.30}$$

The elementary symmetric polynomial e_0 is identically equal to one, being non-informative. Choosing a period $T = 2\pi$ for x makes that the rotation of the shape representative r_s by θ analogous to shifting signal x by θ. The problem of obtaining rotation invariance reduces to the problem of finding a complete shift-invariant representation for signals.

Now, concerning the invariant $\rho^{p,s,t}$ of the previous section, the coordinate k of $\rho^{p,s,t}(\phi(g, r_s))$, which we denote by $\{\rho^{p,s,t}(\phi(g, r_s))\}_k$, changes with the rotation z^r the following way:

$$\left\{\rho^{p,s,t}(\phi(g, r_s))\right\}_k = \left\{\rho^p\left(\frac{z^r}{e(r_s - \bar{r}_s)}(r_s - \bar{r}_s)\right)\right\}_k \tag{5.31}$$

$$= e_k\left(\frac{z^r}{e(r_s - \bar{r}_s)}(r_s - \bar{r}_s)\right) \tag{5.32}$$

$$= (z^r)^k e_k\left(\frac{1}{e(r_s - \bar{r}_s)}(r_s - \bar{r}_s)\right), \tag{5.33}$$

where $z^r = e^{j\theta}$, with $\theta \in [0, 2\pi)$. Equalities (5.31) and (5.32) come from (5.28) and (5.23). Equality (5.33) is obtained by the homogeneity of the elementary symmetric polynomials (3.25).

From equality (5.33), we see that the invariant $\rho^{p,s,t}$ depends only on the rotation part z^r of the action (5.2). Furthermore, by multiplying r_s by $e^{j\theta}$, the representation $\rho^{p,s,t}(r_s)$ changes in the same way as the coefficients of the Fourier series of a signal of period 2π changes with a forward-shift by θ, with $\theta \in [0, 2\pi)$ (see Section 4.1). This enables us to use the spectral invariants introduced in Chapter 4 to factor out rotation.

Starting with $\rho^{p,s,t}$, rotation invariance is then obtained by defining the invariant $\rho^{r,p,s,t}$ as

$$\left\{\rho^{r,p,s,t}(r_s)\right\}_{k_1,k_2} = b_{k_1,k_2}(\rho^{p,s,t}(r_s)), \tag{5.34}$$

where $b_{k_1,k_2}(\rho^{p,s,t}(r_s))$ is the coefficient of order k_1, k_2 of the bispectrum computed with $\rho^{p,s,t}(r_s)$ as the coefficients of the Fourier series and $k_1 \ge 2$, $k_1 \ge k_2$, and $k_1 + k_2 \le N$. The conditions of k_1 and k_2 are due to the definition of $\rho^{p,s,t}$ and the symmetry of the bispectrum.

The invariance and completeness of the representation $\rho^{r,p,s,t}$ is immediate from (5.33), the parallel to the effect that a signal shift has on the coefficients of the Fourier series, and the statements of Section 4.3, about the invariance and completeness of the bispectrum. Figure 5.1 illustrates the invariance and completeness of the invariant map $\rho^{r,p,s,t}$ to the group action (5.2).

The shape representation mapping ρ for the group action (5.2) is $\rho^{r,p,s,t}$. The invariance to rotation remains valid if we substitute the bispectrum by the power spectrum, however, the completeness is obviously lost (see Section 4.2).

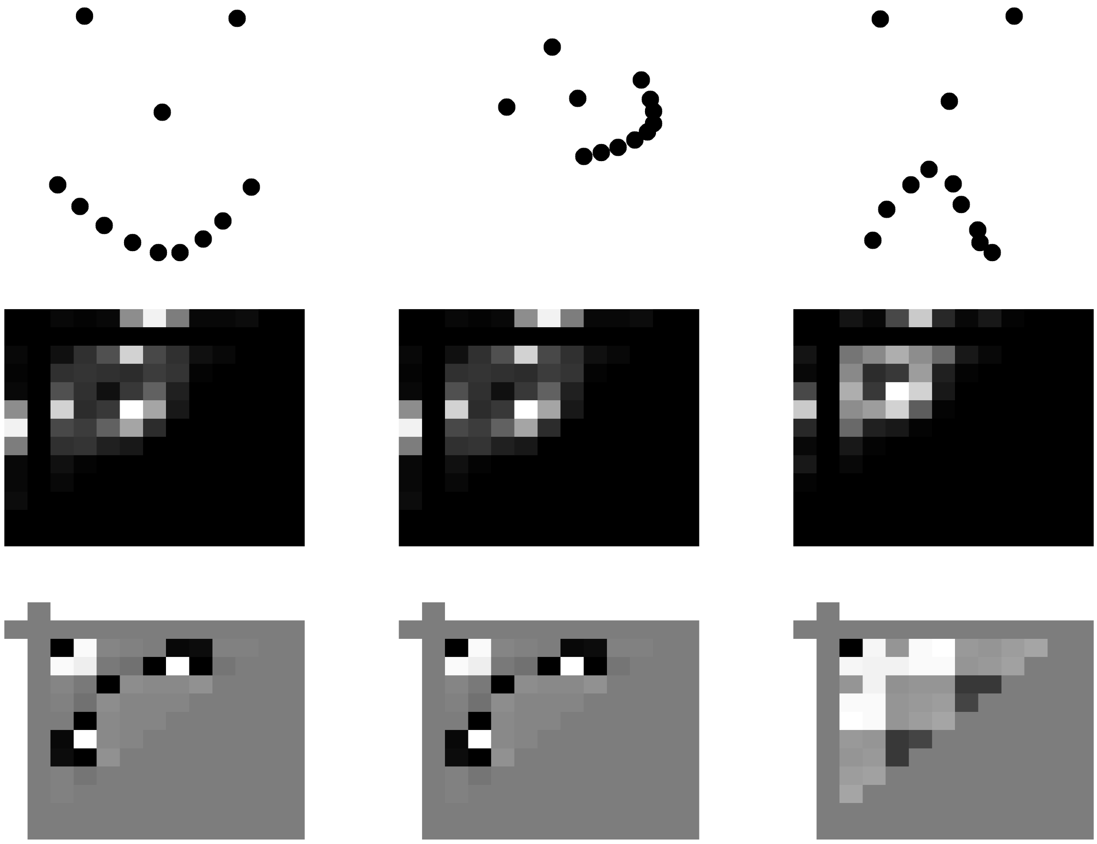

Figure 5.1: Top: shape representatives r_s and r'_s, related by a shape-preserving transformation (i.e., translation, scaling, permutation and rotation), and a distinct shape representative $r_{s'}$. Middle and bottom: the corresponding representations (magnitude and phase), illustrating that the proposed representation $\rho = \rho^{r,p,s,t}$ is simultaneously complete and invariant, i.e., $\rho(r_s) = \rho(r'_s) \neq \rho(r_{s'})$).

5.6 Computational Summary

In this chapter, we introduced the group inducing the shape-preserving transformations (Section 5.1) and built a complete invariant shape representation (from Section 5.2 to Section 5.5). The final representation was built by composing invariants.

Given a shape representative $r_s \in \mathbb{C}^N$, we want to compute its representation $\rho(r_s)$. We start by removing the mean (see (5.12)) and normalizing energy (see (5.15) and (5.14)). This yields $\rho^{s,t}(r_s)$, which is the same only for shape representatives r'_s that are related to r_s by a translation and scale factor.

Then, we evaluate the permutation invariant ρ^p (see (5.23)) on the scaling and translation invariant representation $\rho^{s,t}(r_s)$ of representative r_s, which amounts to evaluating the elementary symmetric polynomials $e_k(\rho^{s,t}(r_s))$, for $k = 2, \ldots, N$ (see Section 3.7 for how to efficiently evaluate the elementary symmetric polynomials). Each of the values of the elementary symmetric polynomials $e_k(\rho^{s,t}(r_s))$ is then divided by N-choose-k (the number of different monomials summed). This yields the representation $\rho^{p,s,t}(r_s)$, which is a complete invariant to permutation, scaling and translation.

Finally, the coordinates of the permutation, scaling and translation invariant $\rho^{p,s,t}(r_s)$ are used as coefficients of the Fourier series to compute the bispectrum. We just need to compute the coefficients k_1, k_2 of the bispectrum that satisfy $k_1 \geq 2$, $k_1 \geq k_2$, and $k_1 + k_2 \leq N$. This factors out the rotation, which is the only part of the group action that remains after the computation of $\rho^{p,s,t}(r_s)$. This yields the final shape representation $\rho^{r,p,s,t}(r_s)$, where $\rho^{r,p,s,t}$ is our shape representation mapping ρ, which is invariant to the full group action (5.2).

The alternatives to the usage of the elementary symmetric polynomials, described in Section 5.4, and the bispectrum, described in Section 5.5, can be employed without compromising the invariance of the representation ρ, but with the referred implications on completeness. In Appendix C, we propose an extension to the proposed representation that enables dealing with shape reflections.

Chapter 6

Experimental Results

In this chapter, we illustrate the properties of the representation proposed in Chapter 5, using distinct scenarios. To conduct the experiments, we developed a (MATLAB coded) software package that enables specifying shapes either in terms of sets of 2D points or sets of images. In the latter case, shapes are extracted by using edge detection or thresholding. The experiments we single out in this chapter illustrate how nearest neighbor shape classification behaves in the presence of noise, the automatic clustering of binary images, and the capability of dealing with shapes that are extracted from real images with simple edge detection.

6.1 Robustness to Noise

We used a database of four simple shape representatives (see Figure 6.1) and performed 5000 tests that consisted in classifying randomly disturbed, (i.e., translated, rotated, and scaled) noisy versions of the shape representatives in the database. A disturbed shape representative r'_s with representation $\rho(r'_s)$ is classified as having the same shape as the shape representative r_s in database with the closest representation $\rho(r_s)$. The distance is given by the Euclidean distance in $\mathbb{C}^M$, where M is the size of the representation. The classification rule is then

$$\arg\min_{r_s \in \mathcal{D}} ||\rho(r_s) - \rho(r'_s)||, \tag{6.1}$$

where $\mathcal{D}$ is the database of shape representatives. Remember that the representations $\rho(r_s)$ for the shape representatives r_s in the database are computed as described in Section 5.6.

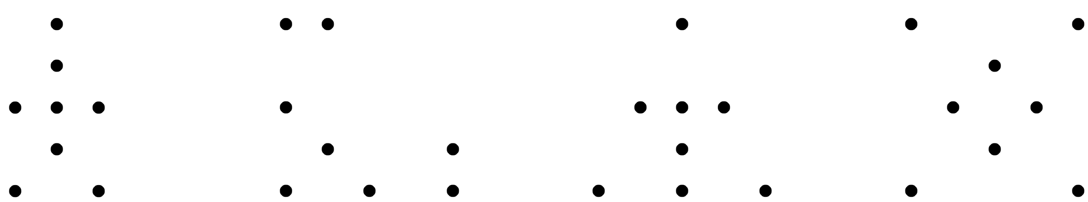

Figure 6.1: Noise-free shape representatives in the database.

The plot in Fig. 6.2 shows the percentage of correct classifications as a function of the noise level, showing 100% correct retrievals with noise standard deviation up to $\sigma = 0.25$, which is high enough to produce the perceptually misleading shape representatives in Figure 6.3 (displayed with the same size and orientation as the ones in Figure 6.1).

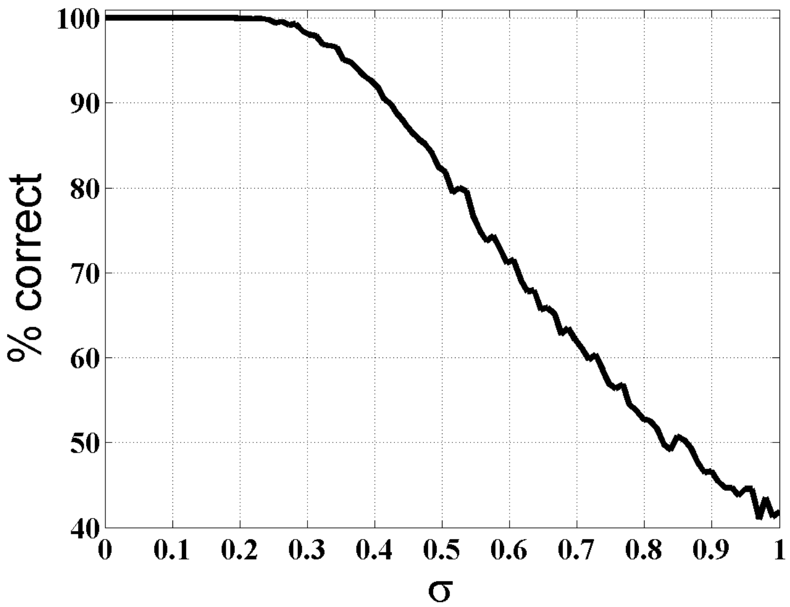

Figure 6.2: Shape classification accuracy as a function of the standard deviation of the noise.

In Appendix B, we study analytically the behavior of the elementary symmetric polynomials and of the power sum symmetric polynomials when their variables are perturbed, providing insight into the behavior of the representation ρ.

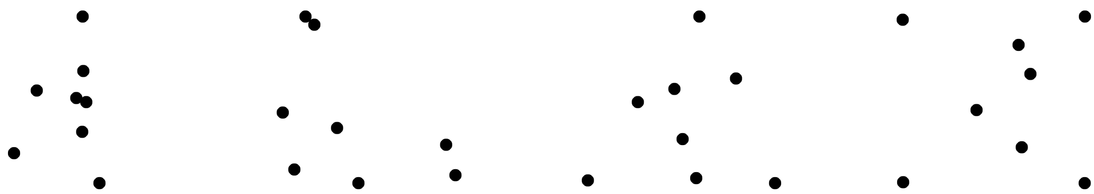

Figure 6.3: Noisy versions of the shape representatives in the database (Figure 6.1) with $\sigma = 0.25$, the approximate limit for 100% correct classifications (Figure 6.2).

6.2 Shape Clustering

The task described in the previous section is based on the comparison of the representation of pairs of shape representatives, thus it could be alternatively approached by attempting to compute the transformation between them (a non-trivial problem in general, as discussed in Section 1.2). In this section, we consider clustering in the space of shape representations. In clustering, all data points are unlabeled and we rely on the assumption that the data points of the same class have similar representations to group them into clusters.

Most algorithms for clustering, require data represented in a way that factors out relevant transformations, so that statistics such as means, variances, etc, can be computed. To illustrate that our representation is adequate for these kinds of tasks, we use $30{\times}30$ binary images obtained by thresholding gray-level images of digits with random orientations. The shape representatives extracted from these images (shown in Figure 6.4) are simply the sets of points corresponding to image pixels of value 1, which are not exactly related by a geometric transformation, due to the coarse discretization and binarization.

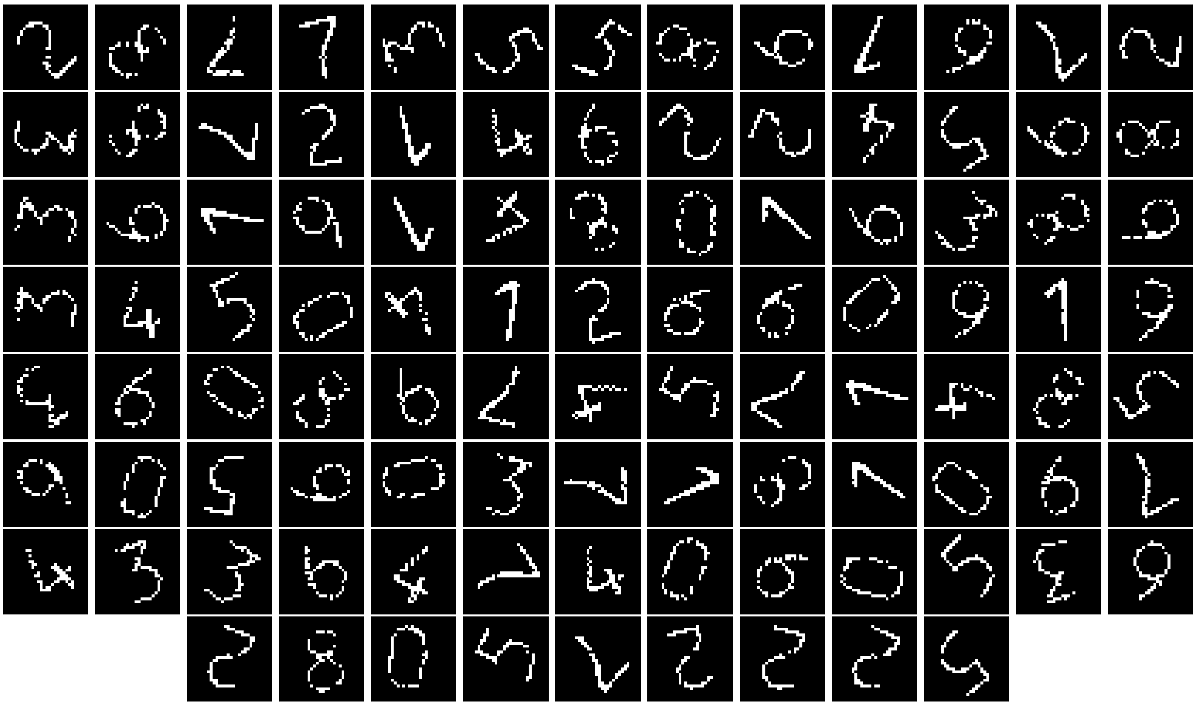

Figure 6.4: Unlabeled images of digits to group into clusters.

Figure 6.5 displays the result of a standard method (hierarchical K-means [35]) used to automatically cluster the representations of the images in Figure 6.4. Note that the images corresponding to digits "6" and "9" are grouped into the same cluster, which is not surprising, since they only differ by distinct orientations of the same geometric pattern, thus having similar representations.

6.3 Trademark Classification

Finally, we describe an experiment where the shape representatives to classify are the edges of real images. Basically, we used (hand-held) webcam images of trademark logos (see the three examples of the top of Figure 6.6). The shapes to classify are given by the (Canny [36]) edge maps of these images (see the examples on the bottom of Figure 6.6). Besides the distinct positions, sizes, and orientations of the logos, other disturbances come from the only approximate perpendicularity of the camera axis to the paper plane, which originates geometrically distorted shapes, and the sensitivity of the edge detection to illumination, resolution, etc. In spite of these disturbances, we were able to successfully classify several of the images by directly comparing the proposed representations of the corresponding edge maps. As in

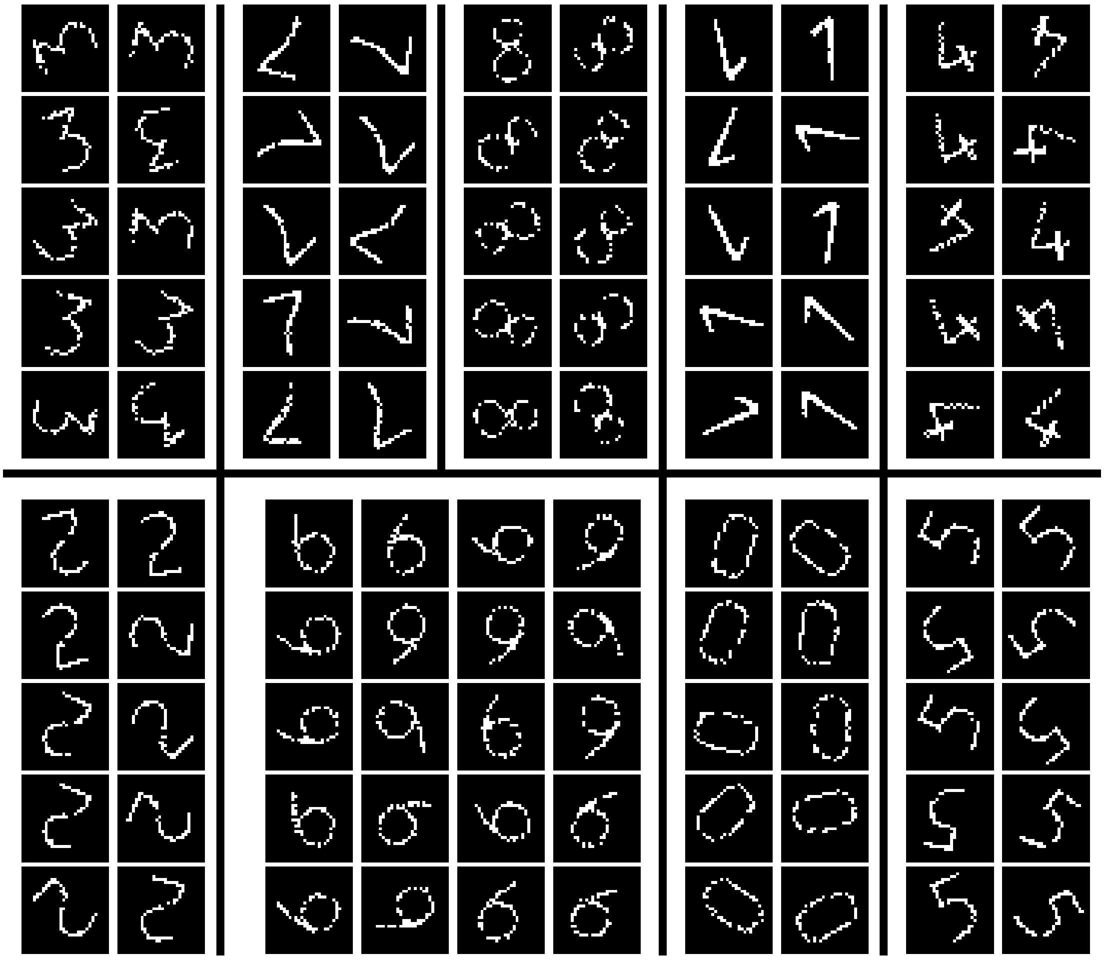

Figure 6.5: Automatic clustering of the binary images in Figure 6.4.

Section 6.1, we used a database containing just one shape representative for each of the different logos considered. The classification rule was again (6.1), i.e., each shape representative is classified as having the same shape as the shape representative in the database with the closest representation. Examples of images correctly classified are shown in Figure 6.7.

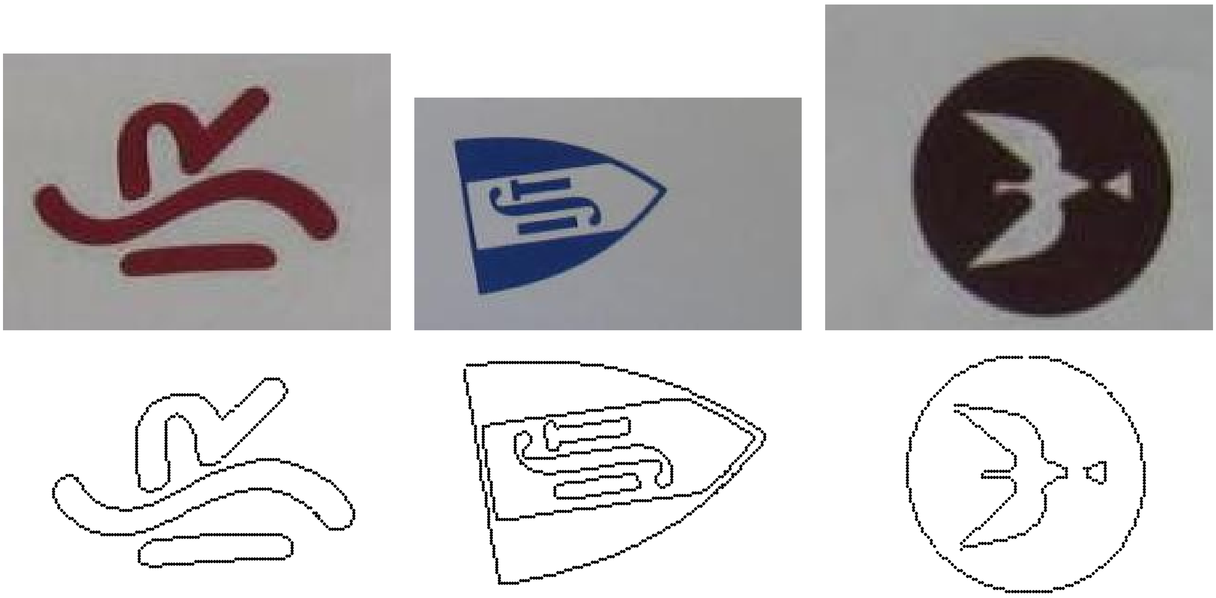

Figure 6.6: Top: three examples of images captured with an hand-held webcam. Bottom: the corresponding edge maps. The shapes to classify are given by the coordinates of the black points in these maps.

Figure 6.7: Examples of webcam images of logos correctly recognized.

Chapter 7

Conclusion

In this thesis we dealt with the problem of representing two-dimensional shapes described by arbitrary sets of points in the plane. We started by framing the problem using group theoretical concepts. Through the action of the group that encodes the shape-preserving transformations on the space of shape representatives, we define shapes as the orbits of the action. The problem of representing shapes reduces to representing orbits.

We discussed the difficulties inherent with dealing with full orbits and concluded that a good approach to the problem would be to find a map from the space of representatives to the shape representation space. For this map to be useful, we required it to be invariant to the group action and complete. These two properties together imply that the two shape representatives have the same representation if and only if they belong to the same orbit.

We presented the symmetric polynomials, which are invariants with respect to permutation of the variables. Furthermore, we have seen that the elementary symmetric polynomials and the power sum symmetric polynomials are complete invariants to permutation. We derived an interesting connection between the symmetric polynomials and the coefficients of the Fourier series of a periodic signal, motivating the introduction of spectral invariants, as the power spectrum and the bispectrum, to factor out rotation. While the power spectrum is invariant to signal shifts but not complete, the bispectrum is both invariant and complete. This allowed the factorization of rotation and permutation of the labels in a complete manner.

Building on these facts, we proposed a shape representation that is complete and invariant to translation, rotation, scaling, and permutation of the labels. The invariants to each of the transformations are constructed and subsequently used, until we arrive to an invariant to the full action of the desired group.

Finally, we illustrated the capabilities of the proposed representation with experimental results. We presented a retrieval example, where we analyzed the robustness of the representation to the noise affecting the point positions; a shape clustering example, where the proposed representation is used as a feature vector by a simple clustering algorithm; and an example of shape recognition with real images, where we compute edge maps from the images and then classify them using the proposed representation.

Our work has uncovered questions that deserve further exploration. We single out the following ones:

- How does the proposed representation deal with subsampling? Is it robust or does it vary dramatically with it? Is there a normalization for the representation that mitigates the dependency on the number of points?
- What kind of statistical analysis can be performed? Are there any analytical results that can be derived for the statistics of the representation when noise with a given distribution is added to the points?
- How to estimate shape in the representation space in an optimal way? i.e., if we have several observations of a single shape, how can we compute an estimate of shape in order to reduce the effect of the noise?
- How to reconstruct the shape, apart an arbitrary shape-preserving transformation, from the computed representation in an efficient way?
- Are there any advantages on using symmetric polynomials besides the elementary symmetric polynomials and the power sum symmetric polynomials?

Appendix A

Shape Dissimilarity

In Chapter 2, we presented the problem of shape representation using group theory. The notion of shape is defined through the action of a group on the space of shape representatives $\mathbb{C}^N$. We denote the space of orbits by $\mathbb{C}^N/\sim$, where $\sim$ is the equivalence relation induced by the group action.

To compare shapes we need to address the notion of dissimilarity on the space of orbits $\mathbb{C}^N/\sim$. A natural way to do so would be to define a metric on this space, leading to a metric space. A metric $d : X \times X \to \mathbb{R}$ on a set X satisfies the following axioms:

Non-Negative: $d(x, y) \geq 0$, for all x, y in X;

Identity of Indiscernibles: $d(x, y) = 0$ if and only if $x = y$, for all x, y in X;

Symmetric: $d(x, y) = d(y, x)$, for all x, y in X;

Triangle Inequality: $d(x, z) \leq d(x, y) + d(y, z)$, for all x, y, z in X.

We could attempt to define dissimilarity on the shape space $\mathbb{C}^N/\sim$ by using a metric on the space of shape representatives (which is easy to come up with). The distance between two shapes would be given by the distance between the closest pair of representatives, i.e.,

$$\tilde{d}_{\mathbb{C}^N/\sim}(R_s, R_{s'}) = \inf_{r_s \in R_s, r_{s'} \in R_{s'}} d_{\mathbb{C}^N}(r_s, r_{s'}), \tag{A.1}$$

where $\tilde{d}_{\mathbb{C}^N}$ is the dissimilarity measure on the shape space $\mathbb{C}^N/\sim$ and $d_{\mathbb{C}^N}$ is the metric on the space of the shape representatives $\mathbb{C}^N$. However, the dissimilarity measure defined by (A.1) is not a metric in general. While the first three axioms of a metric are verified, the triangular inequality is not. This can be understood intuitively by the fact that moving along the orbits does not change the distance. Consider two orbits that are further apart, but each of them is much closer to a third orbit (the third orbit has a representative close to one of the first orbit and another representative close to one of the second orbit). This creates a ”bridge” between the two orbits, violating the triangular inequality. Due to this difficulty, we drop the requirement of having a metric on the shape space $\mathbb{C}^N/\sim$.

The definition of dissimilarity (A.1) has another problem, since its computation involves full orbits, which is intractable in general (see Section 2.3). To solve this problem, we can consider the map $\rho : \mathbb{C}^N \to \mathcal{R}$

from the space of shape representatives $\mathbb{C}^N$ to a space of shape representations $\mathcal{R}$, which encodes all the orbit information without dealing with all its elements, as introduced in Section 2.3. If the shape representation mapping ρ is complete and invariant, it is immediate that any metric on the space of shape representations $\mathcal{R}$ is a dissimilarity measure on the space of shapes $\mathbb{C}^N/\sim$, which satisfies the first three axioms of a metric on $\mathbb{C}^N/\sim$.

The fact that our dissimilarity measure involves a metric on the space of shape representations $\mathcal{R}$ should not concern us here since in the cases that we will deal with, the space of shape representations $\mathcal{R}$ will be $\mathbb{C}^M$ and for this case, a metric is readily available (we use the Euclidean distance).

Appendix B

Perturbation Analysis

The monomial symmetric polynomials are sums of products of the arguments, which makes nontrivial to derive how a perturbation of the arguments propagates. In this appendix, we study how the elementary symmetric polynomials and the power sum symmetric polynomials change with perturbations of the arguments.

Both the elementary symmetric polynomials and the power sum symmetric polynomials are particular cases of monomial symmetric polynomials (see Section 3.4 and Section 3.3, respectively). The decomposition for monomial symmetric polynomials, derived in Section 3.7, provides a method to efficiently evaluate them. We denote the perturbation in z_n as $\Delta z_n \in \mathbb{C}$ and the perturbed versions of z_n as z'_n, where $z'_n = z_n + \Delta z_n$, with $n = 1, \ldots, N$.

For the elementary symmetric polynomials e_k, the decomposition (3.31) reduces to (3.34), which we recall here:

$$e_k(z_1, \ldots, z_N) = z_N e_{k-1}(z_1, \ldots, z_{N-1}) + e_k(z_1, \ldots, z_{N-1}). \tag{B.1}$$

Remember that decomposition (B.1) can also be written as

$$e_k(z_1, \ldots, z_N) = z_n e_{k-1}(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N) + e_k(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N), \tag{B.2}$$

where $n = 1, \ldots, N$ (See equality (3.32)). (For $n = 1$ and $n = N$, $z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N$ means $z_2, \ldots, z_N$ and $z_1, \ldots, z_{N-1}$, respectively.) Note that equation (B.2) is linear in each of the variables $z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N$ when we fix z_n. The partial derivative of $e_k(z_1, \ldots, z_N)$ with respect to z_n is

$$\frac{\partial e_k}{\partial z_n}(z_1, \ldots, z_N) = e_{k-1}(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N), \tag{B.3}$$

where $n = 1, \ldots, N$. Equality (B.3) is clear from (B.2). Consequently, the gradient of $e_k(z_1, \ldots, z_N)$ is

$$\nabla e_k(z_1, \ldots, z_N) = \begin{bmatrix} e_{k-1}(z_2, \ldots, z_N) \\ \vdots \\ e_{k-1}(z_1, \ldots, z_{N-1}) \end{bmatrix}. \tag{B.4}$$

The first-order approximation of $e_k(z'_1, \ldots, z'_N)$ is given by

$$\begin{aligned} e_k(z'_1, \ldots, z'_N) &\approx e_k(z_1, \ldots, z_N) + \nabla e_k(z_1, \ldots, z_N)^T \Delta z \\ &\approx e_k(z_1, \ldots, z_N) + e_{k-1}(z_2, \ldots, z_N)\Delta z_1 + \ldots + e_{k-1}(z_1, \ldots, z_{N-1})\Delta z_N, \end{aligned} \tag{B.5}$$

where $z'_n = z_n + \Delta z_n$, with $n = 1, \ldots, N$ and $\Delta z = [\Delta z_1, \ldots, \Delta z_N]^T$. It is curious that perturbing the variable z_n by Δz_n induces a perturbation on the value of the elementary symmetric polynomial $e_k(z_1, \ldots, z_N)$ that depends only on the value of the elementary symmetric polynomial $e_{k-1}(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N)$ — which does not depend on the perturbed variable z_n — and on the perturbation Δz_n.

For the power sum symmetric polynomials p_k, we can write

$$p_k(z_1, \ldots, z_N) = z_n^k + p_k(z_1, \ldots, z_{n-1}, z_{n+1} \ldots, z_N), \tag{B.6}$$

where $n = 1, \ldots, N$ (see equality (3.32)). The partial derivative of p_k with respect to z_n is then

$$\frac{\partial p_k}{\partial z_n}(z_1, \ldots, z_N) = k z_n^{k-1}, \tag{B.7}$$

where $n = 1, \ldots, N$. Consequently, The gradient of p_k is

$$\nabla p_k(z_1, \ldots, z_N) = \begin{bmatrix} k z_1^{k-1} \\ \vdots \\ k z_N^{k-1} \end{bmatrix}. \tag{B.8}$$

The first-order approximation of $p_k(z'_1, \ldots, z'_N)$, which is valid for small perturbations $\Delta z_1, \ldots, \Delta z_N \in \mathbb{C}$, is given by

$$p_k(z'_1, \ldots, z'_N) \approx p_k(z_1, \ldots, z_N) + \nabla p_k(z_1, \ldots, z_N)^T \Delta z \tag{B.9}$$

$$\approx p_k(z_1, \ldots, z_N) + k z_1^{k-1} + \ldots + k z_N^{k-1}, \tag{B.10}$$

where $z'_n = z_n + \Delta z_n$, with $n = 1, \ldots, N$ and $\Delta z = [\Delta z_1, \ldots, \Delta z_N]^T$. Distinctly to what happens with the elementary symmetric polynomials, perturbing z_n induces a perturbation on the value of the power sum symmetric polynomials $p_k(z_1, \ldots, z_N)$, that only depends on the perturbed variable z_n, and on no other.

We now consider arbitrary perturbations (not necessarily small) of the variables $z_1, \ldots, z_N$. We have by the definition of the elementary symmetric polynomials (3.15):

$$e_k(z'_1, \ldots, z'_N) = \sum_{(i_1, \ldots, i_N) \in I_k^N} z_1'^{i_1} \ldots z_N'^{i_N} \tag{B.11}$$

$$= \sum_{(i_1, \ldots, i_N) \in I_k^N} (z_1 + \Delta z_1)^{i_1} \ldots (z_N + \Delta z_N)^{i_N}. \tag{B.12}$$

Now, let us look at an element $(i_1, \ldots, i_N) \in I_k^N$. Let us assume, without loss of generality, that it is

$(1,\ldots,1,0\ldots,0)$, i.e., $i_1,\ldots,i_k = 1$ and $i_{k+1},\ldots,i_N = 0$. The corresponding monomial expands as

$$(z_1 + \Delta z_1)\ldots(z_k + \Delta z_k) = z_1 \ldots z_k + \sum_{\substack{b_1,\ldots,b_k \in \{0,1\} \\ b_1,\ldots,b_k \neq 0}} z_1^{1-b_1}\Delta z_1^{b_1} \ldots z_k^{1-b_k}\Delta z_k^{b_k}. \tag{B.13}$$

This means that each of the monomials that is summed in (B.11) originates $2^k - 1$ perturbation terms. Since there is a total of $\frac{N!}{k!(N-k)!}$ different monomials summed in (B.12), there is a total of $(2^k-1)\frac{N!}{k!(N-k)!}$ perturbation terms.

For the value of the power sum symmetric polynomials $p_k(z'_1,\ldots,z'_N)$, we have

$$\begin{aligned}
p_k(z'_1,\ldots,z'_N) &= \sum_{n=1}^{N} z'^k_n \\
&= \sum_{n=1}^{N} (z_n + \Delta z_n)^k \\
&= \sum_{n=1}^{N}\sum_{i=0}^{k} \begin{pmatrix} N \\ k \end{pmatrix} z_n^i \Delta z_n^{k-i} && \text{(B.14)} \\
&= \sum_{n=1}^{N} z_n^k + \sum_{n=1}^{N}\sum_{i=0}^{k-1} \begin{pmatrix} N \\ k \end{pmatrix} z_n^i \Delta z_n^{k-i} && \text{(B.15)} \\
&= p_k(z_1,\ldots,z_N) + \sum_{n=1}^{N}\sum_{i=0}^{k-1} \begin{pmatrix} N \\ k \end{pmatrix} z_n^i \Delta z_n^{k-i}, && \text{(B.16)}
\end{aligned}$$

where (B.14) comes from the binomial formula, (B.15) is obtained by singling out from the sum the term $i = k$, and (B.16) is obtained by using the the definition of the power sum symmetric polynomials (3.13).

The behavior of the elementary symmetric polynomials (seen in the equations (B.5) and (B.13)) and of the power sum symmetric polynomials (seen in the equations (B.10) and (B.16)) do not make clear which ones perform better in terms of dealing with perturbations. This is one of the questions that we leave for future work.

Appendix C

Reflection Invariance

In Chapter 5 we propose a shape representation mapping $\rho = \rho^{r,p,s,t}$ that is invariant to translation, scaling and rotation and permutation of the labels. We now study how the shape representation changes with reflections and propose an extension to also accommodate invariance with respect to an arbitrary reflection by redefining the shape dissimilarity measure.

Any reflection in the complex plane can be decomposed in the following sequence of operations:

- a translation, to make the reflection axis pass through the origin;
- a rotation, to align the reflection axis with the real axis;
- a reflection about the real axis;
- the inverse rotation;
- the inverse translation.

Since our shape representation mapping ρ is invariant to translation and rotation, we can assume that the shape representatives are centered at the origin and that the reflection axis is the real axis. Thus, we consider, without loss of generality, only a reflection about the real axis.

A reflection about the real axis in the complex plane corresponds to complex conjugating the points of the shape representative r_s. Let r_s^* be the result of this reflection:

$$r_s^* = \begin{bmatrix} z_1^* \\ \vdots, \\ z_N^* \end{bmatrix}. \tag{C.1}$$

We now show that the representation $\rho(r_s^*)$ is the complex conjugate of the representation $\rho(r_s)$. To derive this, we start by noting that the representation $\rho^{s,t}(r_s^*)$ can be written as a function of the representation

$\rho^{s,t}(r_s)$:

$$\rho^{s,t}(r_s^*) = \frac{1}{e(r_s^* - \bar{r}_s^*)}(r_s^* - \bar{r}_s^*) \tag{C.2}$$

$$= \frac{1}{e(r_s - \bar{r}_s)}(r_s - \bar{r}_s)^* \tag{C.3}$$

$$= \rho^{s,t}(r_s)^*, \tag{C.4}$$

where equality (C.3) comes from noting that conjugation does not affect the mean energy, since it only depends on the absolute values of the entries (see (5.14)), and equalities (C.2) and (C.4) come from the definition $\rho^{s,t}$ in (5.19).

We can verify that $\rho^{p,s,t}(r_s^*) = \rho^{p,s,t}(r_s)^*$ by the following sequence of equalities:

$$\rho^{p,s,t}(r_s^*) = \rho^p(\rho^{s,t}(r_s^*)) \tag{C.5}$$

$$= \rho^p(\rho^{s,t}(r_s)^*) \tag{C.6}$$

$$= \rho^p(\rho^{s,t}(r_s))^* \tag{C.7}$$

$$= \rho^{p,s,t}(r_s)^*, \tag{C.8}$$

where equality (C.5) comes from the definition (5.24) of $\rho^{p,s,t}$. Equality (C.6) comes from (C.4). Equality (C.7) comes from the fact that complex conjugation commutes with multiplication and addition in the complex numbers, and that $\rho^p(\rho^{s,t}(r_s^*))$ consists in the evaluation of a set of (symmetric) polynomials.

Finally, we conclude that $\rho(r_s^*) = \rho^{r,p,s,t}(r_s^*) = \rho^{r,p,s,t}(r_s)^* = \rho(r_s)^*$ because

$$\rho^{r,p,s,t}(r_s^*) = b_{k_1,k_2}(\rho^{p,s,t}(r_s^*)) \tag{C.9}$$

$$= b_{k_1,k_2}(\rho^{p,s,t}(r_s)^*) \tag{C.10}$$

$$= \{\rho^{p,s,t}(r_s)^*\}_{k_1}\{\rho^{p,s,t}(r_s)^*\}_{k_2}\{\rho^{p,s,t}(r_s)^*\}_{k_1+k_2}^* \tag{C.11}$$

$$= b_{k_1,k_2}(\rho^{p,s,t}(r_s))^*, \tag{C.12}$$

with $k_1 \geq 2$, $k_1 \geq k_2$, and $k_1 + k_2 \leq N$. Equality (C.9) comes from the definition (5.34) of $\rho^{r,p,s,t}$, equality (C.10) comes from (C.8), and equalities (C.11) and (C.12) come from the definition (4.8) of the bispectrum.

To extend the invariance of ρ to reflection, while maintaining completeness, we have to capture, for a shape representative r_s, the representations $\rho(r_s)$ and $\rho(r_s)^*$ into one. This can be done by redefining the dissimilarity measure in the following way: for two shape representatives r_s and $r_{s'}$ we compare the representation $\rho(r_{s'})$ with the representations $\rho(r_s)$ and $\rho(r_s^*)$ and output the distance that is the lowest. Since the ρ is complete, the dissimilarity between two shape representatives is zero if and only if the two representatives are related by a rotation, translation, scaling, reflection, and permutation of the labels.

We did not model reflection in the representation of Chapter 5 because this would imply dealing with more elaborated instantiations of the concepts, which could hinder the ideas that we wanted to convey. For example, there is no way to model reflections and rotations as a product group (these operations do not commute and so, the group capturing these operations cannot be commutative). To capture simultaneously rotation and reflection we would have to deal with shapes that are represented in $\mathbb{R}^{2\times N}$, instead of $\mathbb{C}^N$. This is due to the fact that we would have to consider orthogonal matrices, i.e., matrices with determinant plus one or minus one, instead of just special orthogonal matrices, i.e, matrices with determinant plus one. This would imply a more cumbersome action and an awkward transition when factoring out the permutation. By staying with the framework of Chapter 5, we are led to factor out reflection through the redefinition of shape dissimilarity and, since this does not fit in the invariant-based framework of the rest of this thesis, we decided to include the discussion in the appendix.

Bibliography

[1] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.

[2] R. Veltkamp and M. Hagedoorn. State of the art in shape matching. Technical report, 1999.

[3] R. Veltkamp. Shape matching: Similarity measures and algorithms. In *Shape Modeling and Applications, SMI 2001 International Conference on.*, pages 188–197. IEEE, 2001.

[4] D. Zhang and G. Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1–19, 2004.

[5] I. Bartolini, P. Ciaccia, and M. Patella. Warp: Accurate retrieval of shapes using phase of fourier descriptors and time warping distance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(1):142–147, 2005.

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

[7] P. Besl and N. McKay. Method for registration of 3-D shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

[8] G. McNeill and S. Vijayakumar. Hierarchical procrustes matching for shape retrieval. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 885–894. IEEE, 2006.

[9] T. Jebara. Images as bags of pixels. In *ICCV*, pages 265–272, 2003.

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

[11] J. Crespo and P. Aguiar. Revisiting complex moments for 2-D shape representation and image normalization. *Image Processing, IEEE Transactions on*, 20(10):2896–2911, 2011.

[12] J. Rodrigues, P. Aguiar, and J. Xavier. ANSIG—an analytic signature for permutation-invariant two-dimensional shape representation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[13] B. Sadler and G. Giannakis. Shift-and rotation-invariant object reconstruction using the bispectrum. *JOSA A*, 9(1):57–69, 1992.

[14] V. Chandran, B. Carswell, B. Boashash, and S. Elgar. Pattern recognition using invariants defined from higher order spectra: 2-D image inputs. *Image Processing, IEEE Transactions on*, 6(5):703–712, 1997.

[15] R. Kakarala. *Triple correlation on groups*. PhD thesis, University of California, Irvine, 1992.

[16] R. Kondor. *Group theoretical methods in machine learning*. PhD thesis, Columbia University, 2008.

[17] R. Negrinho and P. Aguiar. Shape representation via elementary symmetric polynomials: a complete invariant inspired by the bispectrum. In *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pages 3518–3522. IEEE, 2013.

[18] R. Negrinho and P. Aguiar. Shape representation via symmetric polynomials: a complete invariant inspired by the bispectrum. Submitted for publication, October 2013.

[19] M. Artin. *Algebra*. Pearson Education, 2011.

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

[21] S. Lang. *Algebra*. Springer, 2002.

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

[23] J. Yellott and G. Iverson. Uniqueness properties of higher-order autocorrelation functions. *JOSA A*, 9(3):388–404, 1992.

[24] R. Kakarala and G. Iverson. Uniqueness of results for multiple correlations of periodic functions. *JOSA A*, 10(7):1517–1528, 1993.

[25] H. Bartelt, A. Lohmann, and B. Wirnitzer. Phase and amplitude recovery from bispectra. *Applied Optics*, 23(18):3121–3129, 1984.

[26] J. Heikkila. A new class of shift-invariant operators. *Signal Processing Letters, IEEE*, 11(6):545–548, 2004.

[27] K. Kubicki and R. Kakarala. Experimental results of bispectral invariants discriminative power. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 8290, page 14, 2012.

[28] I. Jouny and R. Moses. The bispectrum of complex signals: definitions and properties. *Signal Processing, IEEE Transactions on*, 40(11):2833–2836, 1992.

[29] R. Kakarala. A group-theoretic approach to the triple correlation. In *Higher-Order Statistics, 1993., IEEE Signal Processing Workshop on*, pages 28–32. IEEE, 1993.

[30] R. Kakarala. Completeness of bispectrum on compact groups. Technical report, 2009.

[31] C. Nikias and J. Mendel. Signal processing with higher-order spectra. *Signal Processing Magazine, IEEE*, 10(3):10–37, 1993.

[32] J. Mendel. Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications. *Proceedings of the IEEE*, 79(3):278–305, 1991.

[33] C. Nikias. Higher-order spectral analysis. In *Engineering in Medicine and Biology Society, 1993. Proceedings of the 15th Annual International Conference of the IEEE*, pages 319–319. IEEE, 1993.

[34] A. Swami, G. Giannakis, and G. Zhou. Bibliography on higher-order statistics. *Signal Processing*, 60(1):65–126, 1997.

[35] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[36] J. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 8(6):679–698, 1986.

[37] R. Prokop and A. Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. *CVGIP: Graphical Models and Image Processing*, 54(5):438–460, 1992.

[38] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, 2002.

[39] A. Khotanzad and Y. Hong. Invariant image recognition by zernike moments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(5):489–497, 1990.

[40] J. Crespo, G. Lopes, and P. Aguiar. Principal moments for efficient representation of 2D shape. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 1085–1088. IEEE, 2009.

[41] D. Kendall. A survey of the statistical theory of shape. *Statistical Science*, 4(2):87–99, 1989.

[42] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Shape Modeling Applications, 2004. Proceedings*, pages 167–178. IEEE, 2004.

[43] Y. Abu-Mostafa and D. Psaltis. Image normalization by complex moments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (1):46–55, 1985.

[44] M. Boutin, K. Lee, and M. Comer. Lossless shape representation using invariant statistics: the case of point-sets. In *Signals, Systems and Computers, 2006. ACSSC'06. Fortieth Asilomar Conference on*, pages 984–988. IEEE, 2006.

[45] C. Teh and R. Chin. On image analysis by the methods of moments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 10(4):496–513, 1988.

[46] H. Kim and J. Kim. Region-based shape descriptor invariant to rotation, scale and translation. *Signal Processing: Image Communication*, 16(1):87–93, 2000.

[47] B. Sagan. *The Symmetric Group: Representations, Combinatorial Algorithms, and Symmetric Functions*. Springer, 2001.

[48] P. Borwein. *Polynominals and Polynomial Inequalities*. Springer, 1995.

[49] R. Kondor. A novel set of rotationally and translationally invariant features for images based on the non-commutative bispectrum. *arXiv preprint cs/0701127*, 2007.

[50] M. Bober. MPEG-7 visual shape descriptors. *Circuits and Systems for Video Technology, IEEE Transactions on*, 11(6):716–719, 2001.

[51] T. Sikora. The MPEG-7 visual standard for content description-an overview. *Circuits and Systems for Video Technology, IEEE Transactions on*, 11(6):696–702, 2001.