Preface

The purpose of this book is to develop a generative theory of shape that has two properties we regard as fundamental to *intelligence* – (1) *maximization of transfer*: whenever possible, new structure should be described as the transfer of existing structure; and (2) *maximization of recoverability*: the generative operations in the theory must allow maximal inferentiability from data sets. We shall show that, if generativity satisfies these two basic criteria of intelligence, then it has a powerful mathematical structure and considerable applicability to the computational disciplines.

The requirement of intelligence is particularly important in the generation of *complex* shape. There are plenty of theories of shape that make the generation of complex shape unintelligible. However, our theory takes the opposite direction: we are concerned with the *conversion of complexity into understandability.* In this, we will develop a *mathematical theory of understandability.*

The issue of understandability comes down to the two basic principles of intelligence - maximization of transfer and maximization of recoverability. We shall show how to formulate these conditions group-theoretically. (1) Maximization of transfer will be formulated in terms of wreath products. Wreath products are groups in which there is an upper subgroup (which we will call a *control group*) that transfers a lower subgroup (which we will call a *fiber group*) onto copies of itself. (2) maximization of recoverability is insured when the control group is symmetry-breaking with respect to the fiber group.

A major part of this book is the invention of classes of wreath-product groups that describe, with considerable insight, the generation of *complex* shape; e.g., in computer vision and computer-aided design. These new groups will be called *unfolding groups.* As the name suggests, such a group works by *unfolding* the complex shape from a structural core. The core will be called an *alignment kernel.* We shall see that any complex object can be described as having an alignment kernel, and that the object can be generated from this kernel by *transferring* structure from the kernel out to become the parts of the object.

A significant aspect of all the groups to be invented in this book, is that they express the object-oriented nature of modern geometric programming. In this way, the book develops an *object-oriented theory of geometry.* For example, we will develop an algebraic theory of object-oriented *inheritance.*

Our generative theory of shape is significantly different from current generative theories (such as that of Stiny and Gips) which are based on production rules. In our theory, shape generation proceeds by *group extensions.* The algebraic theory therefore has a very different character. Briefly speaking: *Features correspond to symmetry groups, and addition of features corresponds to group extensions.*

The major application areas in the book are *visual perception*, *robotics,* and *computer-aided design.* In visual perception, our central principle is that an intelligent perceptual system (e.g., the human perceptual system) is structured as an n-fold wreath product $G_1 \circledS G_2 \circledS \ldots \circledS G_n$. In previous publications, we have put forward several hundred pages of empirical psychological evidence, to demonstrate the correctness of this view for the human system. We shall see that the fact that the visual system is structured as a wreath product, has powerful consequences on the way in which perception *organizes* the world into cohesive structures. Chapter 5 shows how the perceptual *groupings* can be systematically predicted from the wreath product $G_1 \circledS G_2 \circledS \ldots \circledS G_n$.

Chapter six develops a group theory of robot manipulators. We require the group theory to satisfy three fundamental constraints: (1) Perceptual and motor systems should be representationally equivalent. (2) The group linking base to effector cannot be $SE(3)$ (which is rigid) but a group that we will call *semi-rigid*; i.e., allowing a breakdown in rigidity at a specific set of points. (3) The group must encode the *object-oriented* structure.

The theory of robotic kinematics continues in two ways: (1) within the theory of mechanical CAD in Chapter 14; and (2) in the theory of relative motion (in visual perception, computer animation, and physics) given in Chapter 9.

Chapter ten begins the analysis of static CAD by developing a theory of surface primitives, showing that, in accord with the theory of recoverability, the standard primitives of CAD (and visual perception) can be systematically elaborated in terms of what we call *iso-regular groups.* Such groups are n-fold wreath products $G_1 \circledS G_2 \circledS \ldots \circledS G_n$, in which each level G_i is an isometry group and is cyclic or a one-parameter Lie group. To go from such structures to non-primitive objects, one then uses either the theory of splines given later in the book, or the theory of unfolding groups given in Chapters 11, 12 and 13.

The basic properties of an unfolding group are that it is a wreath product in which the control group acts (1) *selectively* on only part of its fiber, and (2) by *misalignment*. In many significant cases, the fiber is the direct product $G_1 \times \ldots \times G_n$ of the symmetry groups G_i of the primitives, i.e., the iso-regular groups, and any fiber copy corresponds to a configuration of objects. The fiber copy in which the object symmetry groups $G_1, \ldots, G_n$ are maximally aligned with each other is called the *alignment kernel*. The action of the control group, in transferring fibers onto each other is to successively misalign the symmetry groups. This gives an *unfolding* effect.

Chapter 14 then presents a lengthy and systematic analysis of mechanical CAD using the above theory. We work through the main stages of MCAD/CAM: part-design, assembly, and machining. For example, in part-design, we give an extensive algebraic analysis of sketching, alignment, dimensioning, resolution, editing, sweeping, feature-addition, and intent-management.

Chapter 15 then carries out an equivalent analysis of the stages of architectural CAD. Then, Chapter 16 gives an advanced algebraic theory of solid structure, Chapter 17 gives a theory of spline-deformation as automorphic actions on groups; and Chapter 18 provides an equivalent analysis for sweep structures.

Chapter 20 examines the conservation laws of physics, in terms of our generative theory, although the next volume will be devoted almost entirely to the geometric foundations of physics. Chapter 21 gives a theory of sequence generation in music.

Finally, Chapters 2, 8, and 22, examine in detail the fundamental differences between our theory of geometry and Klein's Erlanger program. Essentially, in our theory, the recoverability of generative operations from the data set means that the shape acts as a memory store for the operations. More strongly, we will argue that *geometry is equivalent to memory storage*. This is fundamentally opposite to the Erlanger approach in which geometric objects are defined as invariant under actions. If an object is invariant under actions, the actions are not recoverable from the object. We demonstrate that our approach to geometry is the appropriate one for modern computational disciplines such as computer vision and CAD, whereas the Erlanger approach is inadequate and leads to incorrect results.

June 2001 Michael Leyton

Springer

http://www.springer.com/978-3-540-42717-9

A Generative Theory of Shape
Leyton, M.
2001, XV, 549 p., Softcover
ISBN: 978-3-540-42717-9