

Making grammars: From computing with shapes to computing with things



Terry Knight and George Stiny, Department of Architecture, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA

Recent interest in making and materiality spans from the humanities and social sciences to engineering, science, and design. Here, we consider making through the lens of a unique computational theory of design: shape grammars. We propose a computational theory of making based on the improvisational, perception and action approach of shape grammars and the shape algebras that support them. We modify algebras for the materials (basic elements) of shapes to define algebras for the materials of objects, or things. Then we adapt shape grammars for computing shapes to making grammars for computing things. We give examples of making grammars and their algebras. We conclude by reframing designing and making in light of our computational theory of making.
© 2015 Elsevier Ltd. All rights reserved.

Keywords: computational model(s), design theory, perception, reflective practice, shape grammar

The recent wave of interest in making, materiality, and material culture — the so-called ‘material turn’ and ‘new materialism’ (Coole & Frost, 2010; Dolphijn & van der Tuin, 2012) — in the social sciences and humanities has been paralleled by growing attention and research on new materials, making, and manufacturing processes in engineering, science, and design. While humanists and social scientists inquire into the subjective, embodied, situated relationships between people and material things, their engineer, scientist, and designer colleagues tend to focus on technological innovations and applications of advanced materials and fabrication devices.

We pursue a different tack in the terrain of making and material things. We consider making from a computational point of view. Our computational view intersects with some concerns above, but offers a distinct alternative to how we can think about and engage in making. Our view is rooted in computation — but computation beyond the narrow, digital kind of computation to a more general and perceptual kind in which people carry out operations with things that may only have digital approximations. In a similar vein, we consider making to be processes carried out by people to form material things. From this perspective, the kinds of making are extensive and diverse — ranging from drawing a picture on paper, to producing an image on a computer screen,

Corresponding author:
Terry Knight
tknight@mit.edu



www.elsevier.com/locate/destud
0142-694X *Design Studies* 41 (2015) 8–28
<http://dx.doi.org/10.1016/j.destud.2015.08.006>
© 2015 Elsevier Ltd. All rights reserved.

to weaving a basket, to 3D printing a model, to machining engine parts, to constructing a building.

In developing our computational approach to making, we also consider the relationship between making and designing, the latter often understood as an intellectual or cognitive activity resulting in a plan for action or making. Our approach collapses many of the dualisms associated with designing and making that originate with Aristotle's concept of *hylomorphism*. Hylomorphism regards creation as the imposition of an idea of form (*morphe*) upon passive material or matter (*hyle*). A reincarnation of hylomorphism, perhaps better known to architects, is Alberti's distinction between designing — as a 'pre-ordering of the lines and angles conceived in the mind' (Alberti, 1986 [1755]: p. 2) — and building. The anthropologist Tim Ingold and others have argued persuasively to overturn hylomorphic thinking and replace it with an outlook that foregrounds material processes of formation, as opposed to final products as materializations of preconceived ideas (Ingold, 2010: p. 92). Our computational approach promotes this outlook.

Specifically, we look at making through the lens of a unique computational theory: shape grammars. Since their introduction over forty years ago (Stiny & Gips, 1972), shape grammars have been identified with computational design. Within the field of computational design — beginning with its early origins in computer-aided design (CAD) up to the present day — research has focused on design and designing, as the field's name implies. Accordingly, considerable research has centered on what is taken to exist in, or issue from, the mind or the intellect, as in studies of 'design thinking', 'design reasoning', and 'design cognition'. Many of the more influential studies along these lines have been presented within the pages of this journal, and the important core ideas have been brought together very neatly in a recent book by this journal's editor-in-chief, Nigel Cross (2011). Shape grammar studies, on the other hand, have tended to focus on designs and their dynamic, perceptual properties, with less speculation about the thought processes behind their production. Shape grammars have provided a compelling alternative and overlapping theory to the many theories and tools, cognition-oriented or otherwise, within the field of computational design. Here, we propose that shape grammars also offer a natural basis for a computational theory of making.

First, we review shape grammars and the shape algebras that support their improvisational, perception and action approach to computing. Second, we propose a definition of making that follows from the shape grammar approach and is also aligned with contemporary theories of making as they have been articulated in the social sciences, humanities, and elsewhere. Third, we show how shape grammars can be adapted, through our definition of making, to what we call making grammars. Specifically, we modify algebras for the materials — or stuff — of shapes (points, lines, planes, solids) to define algebras for

the stuff of things, and we extend the idea of grammars for shapes to grammars for things. Fourth, we give some examples of making grammars and their algebras. And last, we revisit design and designing in relation to our computational theory of making.

1 *Shape grammars*

Shape grammars are rule-based systems for describing and generating designs (Knight, 1999; Stiny, 2006). Shape grammars are distinctive for their visual approach. They generate designs by computing directly with shapes in two or three dimensions, rather than with symbols, words, numbers, or other abstract structures that represent visual shapes indirectly. Over the years, shape grammars have been developed to tackle a continuum of design tasks from analysis to synthesis. They have been applied successfully in these tasks across virtually all areas of design from urban design to architecture to landscape design to craft to painting to product design to mechanical design. For examples in each of these areas, see, respectively: Beirão (2012), Duarte (2005), Stiny and Mitchell (1980), Muslimin (2010), Knight (1989), McCormack, Cagan, and Vogel (2004) and Agarwal, Cagan, and Stiny (2000).

The rules of a shape grammar have the general form: $A \rightarrow B$ where A and B are shapes and the arrow \rightarrow denotes ‘replace with’. A rule says: Look for the shape A , or a copy of it, in an existing shape S . The copy of A might be some spatial transformation of A (for example, a rotated, reflected, shifted, or scaled copy). If you can see the shape A or its copy, then subtract it from S and replace it with the shape B , or a copy of B using the same spatial transformation as for the copy of A . The result of applying the rule, that is, replacing A with B in the shape S , is the creation a new shape S' . Rules are typically conceived or described in terms of the changes they effect when they are applied — for example, moving a shape, adding a shape, dividing a shape, erasing a shape, and so on. The formal replacement operation (\rightarrow) is a general operation which subsumes all these possibilities.

When the rules of a shape grammar are applied recursively, beginning with an initial shape, a sequence of shapes called a computation is produced. A computation has the form: $S_0 \Rightarrow S_1 \Rightarrow S_2 \Rightarrow \dots \Rightarrow S_n$, where each S_i is a shape computed from the previous shape, beginning with the initial shape S_0 . The double arrow \Rightarrow represents the application of a rule. Shape grammars can be generalized to parametric shape grammars in which shapes in rules have parameters associated with them.

Figure 1 illustrates how the visual rules of a shape grammar work. Each of the two rules shown takes a shape — a square or an L-shape — and moves it in space along a diagonal. Registration marks fix the before and after locations of the shapes. The rules can apply using different transformations or orientations allowing for moves along a diagonal in different directions. Two example

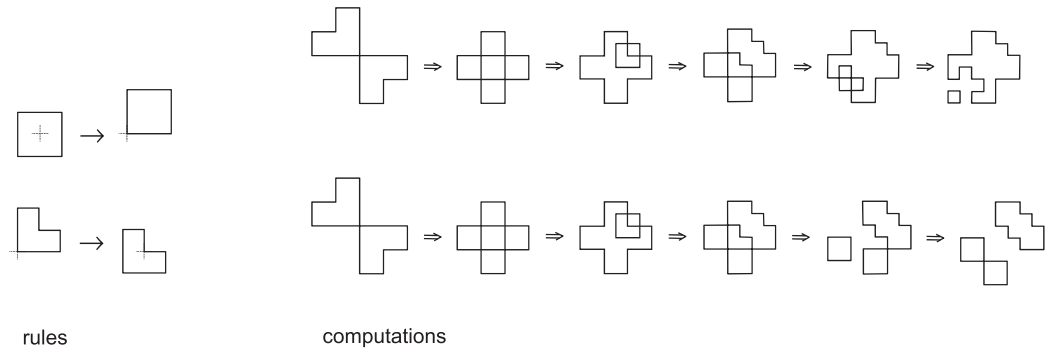


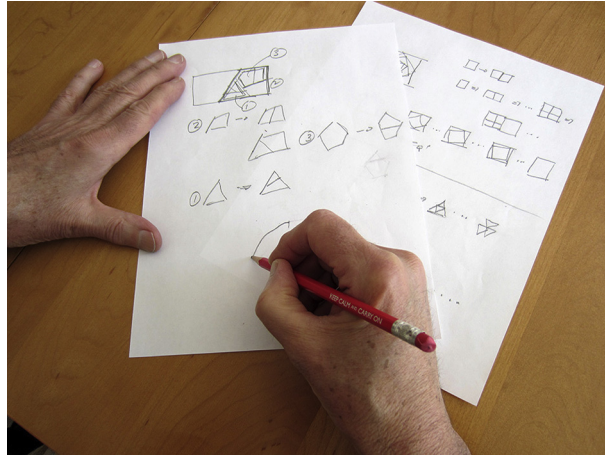
Figure 1 A shape grammar and two possible shape computations

computations, each beginning with a double L-shape, reveal outstanding features of this shape grammar and shape grammars in general. The rules are nondeterministic and, importantly, can apply to emergent shapes, shapes that emerge from the application of rules. The two computations shown are identical up through the first three steps, at which point the rules are applied to different emergent shapes. Many other computations are possible. In other words, the shape computations here, and shape computations in general, are *improvisational, perceptual, and action-oriented*. In each step of a computation, the user can choose on the spot what shapes to see and what to do next.

Shape grammars emulate what we do easily by hand with pencil and paper. Indeed, they can be developed and implemented entirely by hand (Figure 2). Implementation by machine is always possible, but often not straightforward. What is natural and easy to do with our eyes and hands — through seeing and doing (Stiny, 2006) — can be difficult to automate. But shape grammars are not without a rigorous formal foundation. Shape computations are underpinned by an algebraic theory that is a prerequisite to a general computer implementation.

Briefly, an algebra is formally a set of entities that can be changed one into another by operations and transformations, and that can be ordered in terms of one or more relations. Algebras are defined for different basic elements — points, lines, planes, and solids (Stiny, 2006). Basic elements combine by adding them together in terms of a sum operation (+), or subtracting one from another in terms of a difference operation (−). Spatial transformations (t) move basic elements around, reflect them, and change their size. Moreover, there is a part relation (\leq) to show when one basic element is contained in another one. With the part relation, the operations of sum and difference, and the spatial transformations, the way rules work in shape computations can be described in detail. A rule $A \rightarrow B$ applies to a shape S if there is a transformation t that makes A part of S, that is to say, $t(A) \leq S$. Then, the shape S'

Figure 2 Experimenting with shape grammars with paper and pencil



is defined in the formula $S' = (S - t(A)) + t(B)$. All of this corresponds exactly to how rules work, as described informally earlier.

2 Making

Shape grammars foreground action and perception in designing. Designing with shape grammars is about *doing* (*drawing*) and *seeing* with basic *spatial elements* to make *shapes*. We extend this idea of designing to propose a definition of making: Making is *Doing* and *Sensing* with *Stuff* to make *Things*.¹ This definition not only follows from shape grammar theory, but is also in the spirit of work in the social sciences, humanities, and elsewhere on the dynamic, embodied, improvisational, and material aspects of making. John Dewey in his landmark writings on aesthetics described art as a ‘process of doing or making’ with ‘some physical material, the body or something outside the body, with or without the use of intervening tools, and with a view to production of something visible, audible, or tangible’ (Dewey, 1934: p. 47). More recently, others have developed this idea, probing the various dimensions of making including materials (Ingold, 2007b; Lehmann, 2015), skilled practice (Ingold, 2010), the body (Malafouris, 2004), the senses (Howes, 2005), and more.

The components of making proposed here — doing, sensing, stuff, things — are defined informally and are intended to be in keeping with our intuitive understandings of these terms.

2.1 Doing and sensing

Doing is an action — an action by a person or possibly by a machine. Doing includes actions such as drawing, knotting, folding, typing, throwing, stomping, and so on. *Sensing* includes any one or more of our senses, in any of the various ways our senses have been defined. It includes touching, hearing, seeing, feeling, tasting, and so on. Sensing might also include sensing by a machine. We also

take some liberties here with the idea of sensing, and assume that sensing extends to perception. Both doing and sensing can be done with ‘tools’. Tools might be our bodies’ ‘tools’ such as our hands or our eyes. For example, we can do (knot) with our hands and we can sense (see) with our eyes. Tools might also be extensions of our bodies, for example, pencils or eyeglasses. We can do (draw) with a pencil and we can sense (see) with eyeglasses.

Doing and sensing are interrelated. Doing may involve sensing, and vice versa. The order and relations between the two, or between action and perception, have been a topic of much debate and research. Some of this work, for example Alva Noë’s (2004) compelling argument for the dependence of perception on action, might inform our future work.

2.2 *Stuff and things*

Stuff can be physical materials like gases, liquids, or solids (metal, fiber, plastic, wood, and so on) with properties that can be visual (color, texture, translucency, glossiness, and so on), acoustic, mechanical, geometric, and so on. A little more abstractly, stuff can be points, lines, planes, and solids. Stuff might be composed from other stuff. For example, watercolor paint is composed of pigment, binder, solvent, and other materials. In the abstract, stuff has indeterminate extent and parts. When used in making, stuff necessarily has determinate extent, but extent may or may not be appreciable or significant for the maker.

Things are finite objects made of stuff. For example, shapes are things made of line stuff; knots are things made of string stuff; paintings are things made of watercolor stuff (Table 1).

2.3 *Doing and sensing with stuff to make things*

When we make things with stuff, we usually engage multiple ways of doing and multiple modes of sensing. For the sake of simplicity, the examples we give in this paper focus on just one of the possible doings and just one of the possible sensings involved in the making of a thing. Following from the examples given in Table 1, Table 2 gives examples of doing and sensing with stuff to make things, showing one doing and one sensing for each example.

3 *Making grammars*

Shape grammars for computing or making shapes can be adapted, through our definition of making, to *making grammars* for computing or making

Table 1 Examples of things and the stuff with which they are made

<i>Things</i>		<i>Stuff</i>
Shapes	<i>made with</i>	Lines
Knots	<i>made with</i>	Strings
Paintings	<i>made with</i>	Watercolors

Table 2 Examples of doing and sensing with stuff to make things

<i>Doing</i>		<i>Sensing</i>		<i>Stuff</i>		<i>Things</i>
Drawing	<i>and</i>	Seeing	<i>with</i>	Lines	<i>to make</i>	Shapes
Knotting	<i>and</i>	Touching	<i>with</i>	Strings	<i>to make</i>	Knots
Painting	<i>and</i>	Seeing	<i>with</i>	Watercolors	<i>to make</i>	Paintings

things — that is, as means for computational making. The algebras for the stuff of shapes (points, lines, planes, solids) can be modified to define algebras for the stuff of things.

3.1 Algebras for stuff

Algebras for the stuff of shapes are defined in terms of what we can draw and see with spatial stuff. Algebras for the stuff of things generalize this idea to encompass other kinds of doings and sensings with other kinds of stuff. Algebras for stuff formalize properties of materials that are relevant to artists, designers, or makers — properties that may be very different from those of interest to scientists or technologists. For example, materials scientists study the core physical properties of materials, going back to their atomic and electronic structure. These properties are critical for understanding and classifying materials and for discovering and creating new materials, but they are not necessarily useful to artists, designers, or makers. We do not need to know all the details about the physical attributes of materials in order to work with them.

Studies of the sensory properties of materials, on the other hand, are more relevant to making but are few. Some studies are aimed at understanding how the sensory properties — the look, feel, taste, sound, smell, etc. — of materials are related to their physical properties. The ‘[Sensoaesthetic Materials](#)’ project at the Institute of Making at the University College London is an excellent example. Work of this kind is leading to a much richer understanding of materials, but is of limited value in working with materials. Other sensory studies of materials focus on the user’s or consumer’s perception of the materiality of a product or thing, not on the artist-designer-maker’s perceptions of the materials used in making the thing. (For example, see the discussion about the sensory attributes of products in [Ashby and Johnson \(2009\)](#): pp. 76–87.) In other words, the focus is on the sensory properties of a thing from a user’s point of view, not on the sensory properties of the materials from a maker’s point of view, in terms of what s/he can do with materials. Understanding users’ sensory impressions of a thing is a consideration in designing or making a thing, but it is not enough to tell us about how we sense and, importantly, what we can do with materials in making things.

The physical (mechanical, thermal, electrical, and so on) properties of materials which are central to materials science, as well as the sensory properties discussed in product design, are typically represented through quantitative

measurements and formulas. The doing and sensing properties of stuff that are central to making can be captured through algebras. These algebras may be thought of as phenomenal or alchemical. That is, our approach to algebras for stuff treats the sensing and doing properties of stuff as experiential properties, for example, as paint is described by the painter and philosopher Nigel Wentworth:

... a physical description of paint might well be possible and interesting, but it will not capture the paint as the painter employs it. As with tools, paint is something that only is what it is through being used. In other words paint is paint. ... To identify it and try to understand it in simple physical terms is to miss its very existence for what it is. ... To understand it properly as paint we must locate it within the context in which it is paint, and that is within the practice of painting (Wentworth, 2004: pp. 34–35).

Similarly, our approach to algebras for stuff treats the doing and sensing properties of stuff like substances in alchemy, for example, as described by Ingold in his reference to the art historian James Elkins’s discussion of painting:

The world according to alchemy, as art historian James Elkins explains, was not one of matter that might be described in terms of its molecular composition, but one of substances that were known by what they look and feel like, and by following what happens to them as they are mixed, heated or cooled. Alchemy, writes Elkins, ‘is the old science of struggling with materials, and not quite understanding what is happening’ (Elkins 2000: 19). His point is that this, too, is what painters have always done. Their knowledge was also of substances, and these were often little different from those of the alchemical laboratory. As practitioners, the builder, the gardener, the cook, the alchemist and the painter are not so much imposing form on matter as bringing together diverse materials and combining or redirecting their flow in the anticipation of what might emerge (Ingold, 2011: p. 213).

3.2 Grammars for things

With algebras for stuff as foundation, we adapt shape grammars for computing shapes to *making grammars* for computing things.

Like shape grammars, making grammars have rules of the form $A \rightarrow B$. A and B are things, and the arrow \rightarrow is a formal replacement operation. In terms of making, though, the arrow \rightarrow stands for some doing and/or sensing. A doing results in a physical or perceptible change in a thing. A sensing results in a change in the sensing of a thing. As mentioned earlier, doing and sensing are interrelated and one might involve the other. The replacement operation (\rightarrow) is a general operation which subsumes all kinds of doings and sensings, whether simultaneous or independent.

Also like the rules of a shape grammar, the rules of a making grammar can apply under different transformations. Depending on the things computed, the transformations might be the same as the spatial transformations used for shape grammars (p. 3) or they might be other kinds of transformations particular to the things computed. For the sake of simplicity, we consider only spatial transformations in the examples here.

The rules of a making grammar apply in the expected way to define a computation: $T_0 \Rightarrow T_1 \Rightarrow T_2 \Rightarrow \dots \Rightarrow T_n$. Each T_i is a thing computed from the previous thing, beginning with the initial thing T_0 . The double arrow \Rightarrow represents the application of a rule. In terms of making, the double arrow represents a particular doing and/or sensing to make a thing T_i into a thing T_{i+1} .

Making is a continuous, time-based process. A computation may be time-based, but it is not continuous. It segments a making process into parts marked by discrete moments. Each T_i is a thing in a moment — a critical moment, a rest moment, or otherwise — in a making process. It is the outcome of a chunk of the process. The times in between moments can be broken up into finer and finer parts, but some parts may not be describable and only individually enacted. The design researcher Mads Vedel Jensen has described the challenges of analyzing and representing a person's skilled movement through time: moments of rest can be identified, but equal weight must be given to the movements in-between these moments and to the continuity of the overall process (2005). Here, a computation according to the rules of a making grammar represents just one of many possible interpretations of a making process as a sequence of moments and the times in-between.

4 *Examples of making grammars*

We give three examples of making grammars to illustrate the possibilities and range of computational making, and to raise issues and questions for further work: drawing with lines (á la Stiny), knotting with string (á la Ingold), and painting with watercolors (á la Sargent). The first two examples are developed in some detail; the third is more of a proposal. Each example considers making with a particular kind of stuff to make particular things using particular doings and sensings. Each example highlights the need for a specialized algebra that reflects the doing and sensing properties of some stuff. Each example also considers how the doings and sensings with particular stuff might be represented in rules and computations.

4.1 *Drawing with lines (á la Stiny)*

This first example demonstrates that shape grammar computations are a kind of computational making. As we have noted, shape grammars are usually discussed in terms of designing. However, as a means for synthesis, they are a highly sensory, action-oriented kind of computational making. This example recasts the shape grammar of Figure 1 as a making grammar. It includes:

Things	shapes in 2D
Stuff	lines in 2D, represented in an algebra for lines
Doing	drawing (either drawing and erasing, or drawing with trace overlays)
Sensing	seeing (grasping, focusing attention) with the eyes

Figure 3 shows the making grammar. There are two sets of rules: seeing rules and drawing rules. The highlighted shapes in the rules are shapes that are the focus of seeing or ‘grasping’ with the eyes. The seeing rules apply to focus sight on — to visually grasp — a shape. They do not otherwise alter the shape. Seeing rules are similar to identity rules (Stiny, 1996), but here seeing rules explicitly identify what is visually grasped. The drawing rules are like the rules in Figure 1. They each apply to move a visually grasped shape along a diagonal. The computation is a series of alternating seeing and drawing moves. The computation is sequential but is separated into two columns to distinguish doing from sensing. This example is a much-simplified representation of an actual drawing process. The rules could be extended to allow, for example, for switching visual attention between different shapes before a drawing move is implemented, or for finer drawing moves for individual lines or their segments.

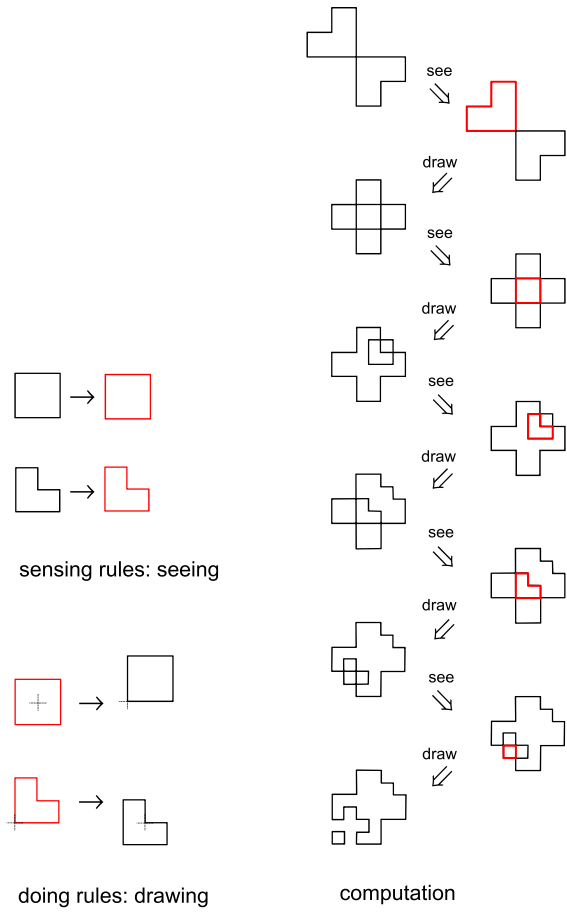
4.2 *Knotting with string (à la Ingold)*

Lines in a two-dimensional drawing space are now rendered as strings in a three-dimensional world. This example is inspired by the many affinities of lines with the sensory, improvisational, and emergent qualities of making with physical materials. It draws in part from the anthropologist Tim Ingold’s use of line as a rich metaphor for making and skilled practice, and more generally, for understanding the world (Ingold, 2007a, 2013a). For Ingold, lines and weaving embody the processes of growth, becoming, and formation that are intrinsic to making. Making is ‘like weaving a pattern from ever unspooling threads that twist and loop around one another, growing all the while without ever reaching completion’ (Ingold, 2013b: p. 26). And knotting is the means by which lines are held together in the making.

The knotting grammar we propose here also ties nicely to *kipu*, the knotted strings made by the Incas as a physical recordkeeping and communication language. Figure 4 shows one of the approximately six hundred extant *kipu*. *Khipu* knots are of three types: a single overhand knot, a long (multiple overhand) knot, and a figure eight knot. These knots are repeated in various sequences along strings, and the strings are hung from a primary cord. The precise meaning and syntax of *kipu* has been a source of speculation. Gary Urton, a leading researcher in the field, has proposed that *kipu* were an early, binary form of narrative writing (Urton, 2003).

Our knotting grammar is simple and general, and a step toward a grammar for *kipu* making. It generates single overhand knots and multiple overhand

Figure 3 Making grammar:
drawing with lines



(long) knots along a string. An overhand knot is one of the most basic knot forms and is the starting point for a variety of more complex knots (Figure 5).

An overhand knot can be repeated within itself to form multiple overhand knots: double, triple, quadruple and so on (Figure 6).

Overhand knots (single or multiple) can also be repeated in different locations along a string (Figure 7). In other words, iteration can occur at two levels: within a knot and along a string.

The knotting grammar encompasses both types of iteration. It includes:

Things	knotted strings in 3D, represented with an algebra outlined below
Stuff	strings
Doing	knotting (looping, pulling, etc.)
Sensing	touching (grasping, focusing attention, repositioning) with the hands

Figure 4 Khipu® President and Fellows of Harvard College, Peabody Museum of Archaeology and Ethnology, PM# 41-52-30/2938 (digital file# 96770002), DETAIL



Figure 5 A step in making an overhand knot

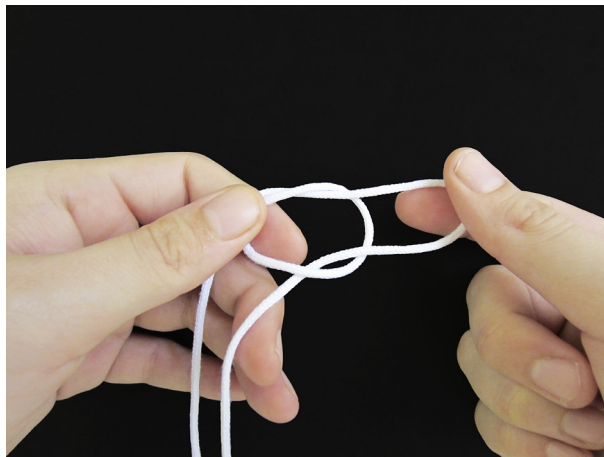


Figure 6 Single and multiple overhand knots (shown before and after tightening the knot)

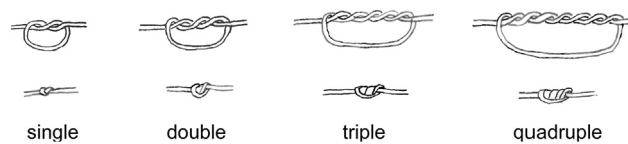
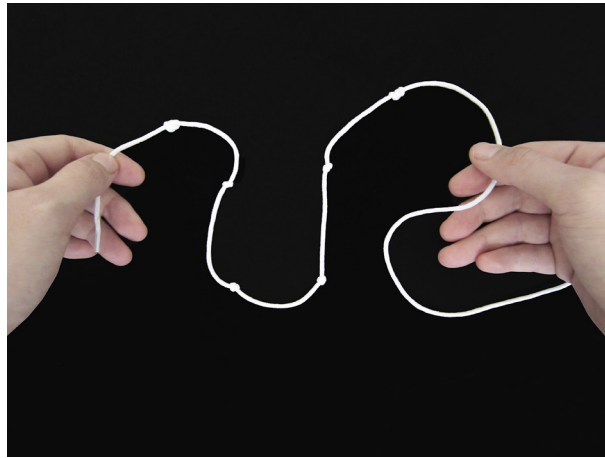


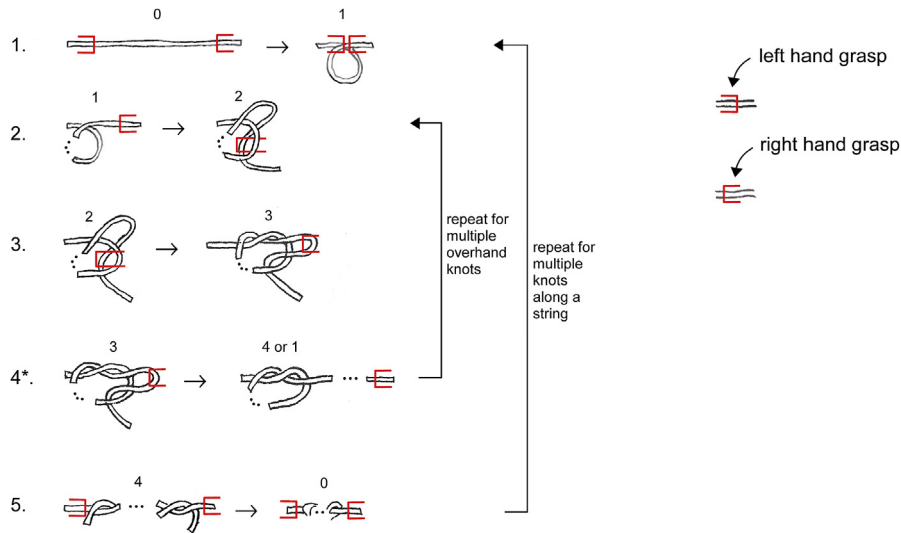
Figure 7 Repeated overhand knots on a string. From left to right: triple, single, single, single, triple knots



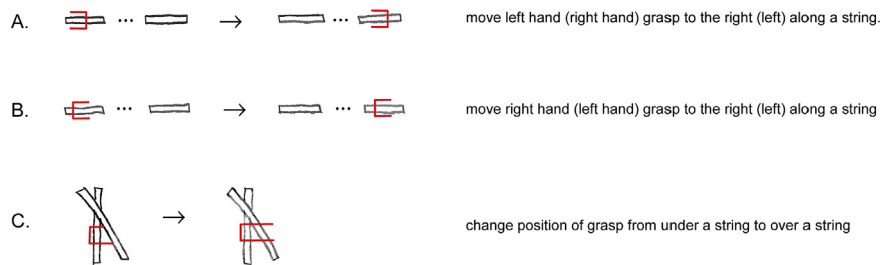
Note that we focus here on the sense of touch alone to simplify the example, and to highlight grasping with the hands as opposed to grasping with the eyes as in the previous example. Knotting is a highly tactile activity, but may involve other senses apart from touch, in particular, seeing. However, with a little practice, it is easy to make these knots entirely by hand and touch without the use of sight.

Figure 8 shows the knotting grammar. There are two sets of rules: knotting rules and touching or grasping rules. The brackets in both sets of rules indicate the locations of left and right hand grasps, using thumb and forefinger, along a string. The knotting rules apply to change the configuration of a string by looping and pulling the string in different ways to tie knots. The touching/grasping rules apply to change a grasp location along a string (rules A and B) or to change the position of a hand (rule C) from under a string to over a string. They do not change the configuration of the string. Rule 4* is both a knotting and grasping rule: it applies to pull a string through a loop while sliding or moving a grasp along the string. The labels 0, 1, 2, 3, and 4 in the doing rules control the sequence in which these rules apply. Sensing rules can be applied at any time. All of the rules apply under the same geometric transformations as shape grammars, so that knots can be made in different orientations and directions. The grammar is a highly schematized version of an actual knotting process. The rules capture natural stopping or stable points in a continuous tying process.

As mentioned earlier, an algebra is a set of entities that can be changed into others or that can be related to one another. Just as the entities in the algebras for shape grammars are points, lines, planes, or solids, the entities in the algebra for string grammars should be strings, or something like them in terms



doing rules: knotting



sensing rules: touching, grasping

Figure 8 Making grammar: knotting with string

of relevant properties. But intuitively, shape elements and strings are not quite the same. In particular, shape elements fuse in combination, whereas strings are independent in combination. Further, both shape elements and strings may have parts embedded in them, but shape elements can have parts embedded in them anywhere whereas the parts of strings are limited by knots and their locations. For example, two lines may combine to make a single, longer line, but two strings in combination are still two separate strings. Moreover, any line corresponds to a part of any other line, whereas a string corresponds to a part of another string only when their knots match up exactly. Such differences between shape elements and strings motivate our definition for an algebra for strings.

A little abstraction is useful for this purpose, keeping in mind the physical and tactile properties of strings. In the algebra, it is assumed that each string is a singleton set to recognize the independence of strings in combination. A string is a sequence $sk s \dots k s$ of segments s and knots k , where for n segments, there are always $n - 1$ knots. Knots cannot be at the ends of strings without trailing segments. Otherwise, the knots would unravel. Moreover, segments may vary in length. A segment is an internal segment if there are knots at both of its end-points. Otherwise the segment is an end segment. If a string contains at least one knot, then it has two end segments. Knots are indexed with integers to indicate the number of ties they contain — 1 is for a single knot, 2 for a double knot, and so on. The string $sk^1sk^3sk^1s$ has four segments divided by a single knot, a triple knot, and a single knot. A tying operation is defined in the algebra to add knots to segments of strings. If a double knot is tied in the third segment of the string $sk^1sk^3sk^1s$, then the new string $sk^1sk^3sk^2sk^1s$ is defined in the algebra. A string S is part of another string S' whenever the knots and the internal segments of S correspond exactly in a 1–1 relationship to knots and internal segments of S' . Internal segments must correspond in sequence and in length. The end segments of S must also be embedded in segments of S' , where embedding is like the part relation for lines in shapes. Knots cannot be ignored in determining whether one string is a part of another string. If a knot is tied in a string, then the original string is not a part of the result. Two strings are equal if their segments and knots all correspond — each string is part of the other. Last, there are transformations in the algebra to change the length and location of strings, while preserving knots and the number of ties in them, and the relative lengths of segments. Knots are invariant under the transformations — just as, for example, labels are in shape grammars — while the lengths of segments can change. The transformations always apply to entire strings.

Interpreting the rules in a knotting grammar in terms of this algebra is fairly straightforward. Essentially, the pieces of a string(s) in the left side of a rule must be included in a string(s) that is part of the string to which the rule applies. The use of transformations may be necessary for this purpose to obtain a string of the correct length. The result of applying a rule or a sequence of rules in the order given in the grammar is a knot corresponding to the tying operation in the algebra.

Note that the algebra works for multiple strings, too, whether they are lined up, laid flat, or otherwise configured in space. Formally, these configurations would be sets of singleton strings. It is also worth considering how the algebra might be extended to material surfaces such as paper, where folds would correspond to knots. An arrangement or array of areas and folds is one enticing possibility.

A computation of a knotted string is shown in [Figure 9](#). The computation is sequential, but separated into two columns to distinguish doing from sensing,

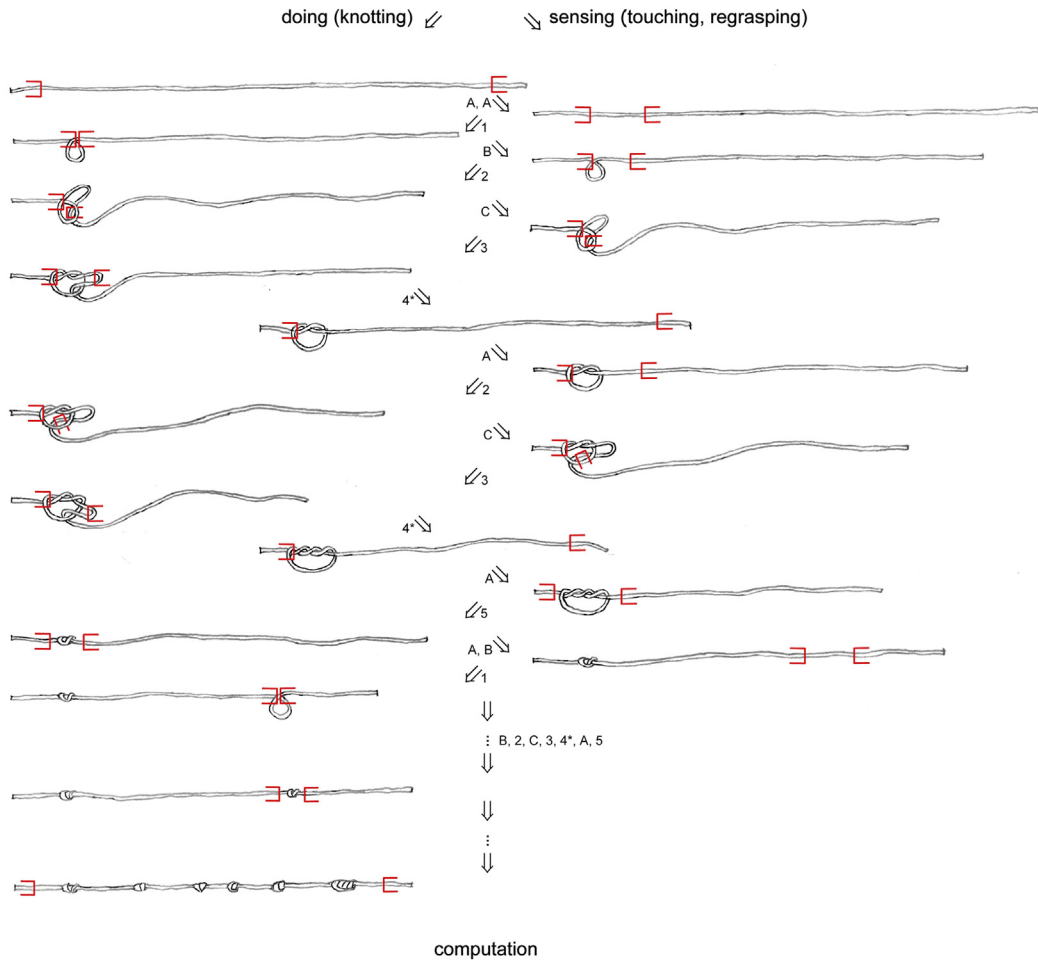


Figure 9 Knotting a string with the grammar of Figure 8

as in Figure 3. Applications of the combined knotting/grasping rule 4* are shown in the middle. Unlike the previous example in Figure 3, this computation is not always a straightforward alternation between doing and sensing.

4.3 Painting with watercolors (*à la Sargent*)

This last example is inspired by a recent exhibit at the Museum of Fine Arts Boston of the watercolor paintings of the American artist John Singer Sargent (Figure 10). This example is suggestive and speculative. We do not actually develop a grammar here, but use Sargent's work to highlight the challenges of understanding and describing computationally a rich, complex making activity such as painting. Unlike the previous two examples, this example presents a making activity involving not just one but a multiplicity of materials or stuff that interact in complex and emergent ways. Relatedly and, also

Figure 10 John Singer Sargent, American, 1856–1925
La Biancheria, 1910. Translucent watercolor, with touches of opaque watercolor and wax resist, over graphite on paper. Sheet: 40.4 × 53.1 cm (15 7/8 × 20 7/8 in.) Museum of Fine Arts, Boston. The Hayden Collection — Charles Henry Hayden Fund, 12.229. Photograph© 2015 Museum of Fine Arts, Boston



different from the previous examples, the making activity involves a multiplicity of doings, or actions, with various tools and techniques. A variety of sensings, such as seeing and touching, are also at play.

Sargent was a virtuoso of watercolor painting. His watercolors were done *en plein air* — on the spot paintings of people, things, and landscapes. He experimented with and exploited the potentials of materials in unprecedented ways, using different papers, different paints (watercolors and tempuras, dry and wet, thin washes and thick impasto), different additives to change a paint's working and visual characteristics, graphite pencil, and wax. His tools were equally various — brushes, scrapers, sponges, and knives. And he deployed his tools through ingenious techniques, both additive and subtractive, to create unexpected texture, color, and lighting effects. (See [Manick and Owen \(2013\)](#): pp. 205–223) for a discussion of Sargent's materials and techniques.) The historian Richard Ormond describes Sargent's working methods:

Sargent used every trick in the book to achieve his effects in watercolor, sponging wet washes into one another, leaving the white of the paper bare for lights, sweeping wash over wax resist to create variegated surface textures, scratching out with knife and brush end to accentuate detail, and finishing things off with thick strokes of gouache and Chinese white for highlights. His style of technical virtuosity left little room for correction. He had the skill and confidence to paint at speed, improvising like a musical performer as he went along ([Ormand, 2013](#): p. 23).

A making grammar for Sargent's watercolor painting would comprise:

Things	watercolor paintings
Stuff	various paint materials, each material represented with an appropriate algebra
Doing	painting, with various implements
Sensing	seeing (grasping, focusing attention) with the eyes, touching

The challenges in developing a grammar to characterize Sargent's highly skilled practice and expertise — even in part — are many. The stuff, the doings, and the sensings are complex, variable, and interrelated. Multiple different materials with different behaviors and properties interact with one another in different ways, in different sequences, and with different tools and techniques.

Each different material may require a different algebra, and these may need to be combined in an algebra for entities of various kinds. Interactions between different algebras (for instance, the effects of a wet wash on wax) would need to be considered. Moreover, interactions would need to include the order in which materials are used (for example, a color wash applied over wax versus wax applied over a color wash). Material properties may not follow the usual mathematical axioms for algebras such as associativity, commutativity, and so on. Indeed, there seems to be no guarantee that materials are even invariant under transformations. But these complications, and no doubt many more, add to the excitement of Sargent's methods and work.

Doing rules would be equally complex as they involve actions with different outcomes depending on the tools involved, the materials used, and the materials to which they are applied. The specific action underlying an arrow \rightarrow in a doing rule $A \rightarrow B$ might need to be characterized anew. Indeed, replacement as in shape rules may no longer be sufficient.

Another unique aspect of Sargent's painting activity, and painting in general, is time. Some materials used in painting change over time with no direct action by the maker, changing the overall results. Colors that are glossy may become matte, for example. Material changes may or may not be premeditated by the maker. Here is a description of how Sargent used the element of time in a calculated way — note the phrase in parentheses at the end:

Some of these strokes [of opaque watercolor] have an unusual speckled appearance that allows the color of the underlying paint layer to show through gaps in the pigment. He achieved this mottled effect by vigorously churning opaque watercolor with a dry brush until the paint got frothy and then dragging his brush across the paper (when the air bubbles dried, they created the gaps) (Sherry, 2013: p. 182).

Time can be incorporated into a grammar in a relatively straightforward way by defining a *timed grammar*. (See [Saeedloei and Gupta \(2010\)](#) for related work on timed automata.) In a timed grammar, a clock or timer is associated with the initial shape and rules. Time is defined through labels that function similarly to state labels in a shape grammar. A clock is set to time 0 for the initial shape and the start of a computation. When a computation begins, the clock starts. The clock time appears in each step of a computation. Any rule of the grammar that is dependent on time has a time label associated with its left and right sides. Not all rules need be time dependent or have time labels. A material element X in a painting (for example, air bubbles) that changes into a new material element Y (for example, gaps) after some interval i of time could be represented by a rule $X \rightarrow Y$, where X has a time label t associated with it and Y has a time label $t + i$ associated with it. Time rules could be included as doing rules in a making grammar, perhaps distinguished as *passive* doing rules in contrast to *active* doing rules.

5 Discussion

The approach we have taken to making involves algebras, and grammars with rules defined in terms of these algebras. The idea is to capture the salient properties of stuff and things in actual making, so that manipulating stuff and things can be described as computation. This has the advantage of clarity and precision in describing stuff and things, and in describing the making process itself. It always helps to try and understand a process, no matter how natural it may seem. Moreover, our approach may suggest new avenues — unexpected computations — for making that may not be obvious simply by the use of traditional methods of making that are tried and true.

Making shapes, making knots, making watercolors, as we describe them here, are all comparable, creative activities. But where does designing come in? We propose that shapes, knots, and watercolors are not designed, but made. Indeed, we would like to reframe the notion of designing — computational or otherwise. We view designing not as an intellectual activity in which we script in advance what we do and what can happen, but as a kind of making itself, an activity that demands perceptual, bodily engagements with materials in the world. Computation does not make designing any less a kind of making. Shape grammars have for long taken this point of view and, in so doing, stand apart from other computational design approaches. Here, we begin to show how the shape grammar perspective can be advanced in new directions for making.

Notes

1. The uses of the terms ‘stuff’ for material and ‘thing’ for an object are commonplace. However, our immediate inspiration for using the two terms in conjunction comes

from the subtitle ‘Things and Stuff’ in the introduction of Edward Adelson’s paper (2001) on the perception of materials.

References

- Adelson, E. H. (2001). On seeing stuff: the perception of materials by humans and machines. In B. E. Rogowitz, & T. N. Pappas (Eds.), *Proceedings of the SPIE 4299, human vision and electronic imaging VI* (pp. 1–12).
- Agarwal, M., Cagan, J., & Stiny, G. (2000). A micro language: generating MEMS resonators by using a coupled form – function shape grammar. *Environment and Planning B: Planning and Design*, 27(4), 615–626.
- Alberti, L. B. (1986). *The Ten Books of Architecture (1755 Leoni Edition)*. NY: Dover.
- Ashby, M., & Johnson, K. (2009). *Materials and design: The art and science of material selection in product design second edition*. Oxford: Butterworth-Heinemann.
- Beirão, J. N. (2012). *CityMaker: Designing grammars for urban design*. Delft, The Netherlands: Faculty of Architecture, Delft University of Technology. (PhD Dissertation).
- Coole, D., & Frost, S. (Eds.). (2010). *New materialisms: Ontology, agency, and politics*. Durham, NC: Duke University Press.
- Cross, N. (2011). *Design thinking*. Oxford: Berg.
- Dewey, J. (1934). *Art as experience*. New York: Capricorn Books.
- Dolphijn, R., & van der Tuin, I. (2012). *New materialism: Interviews & cartographies*. Ann Arbor, MI: Open Humanities Press.
- Duarte, J. P. (2005). Towards mass customization of housing: the grammar of Siza’s houses at Malagueira. *Environment and Planning B: Planning and Design*, 32, 347–380.
- Elkins, J. (2000). *What painting is*. NY & London: Routledge.
- Howes, D. (Ed.). (2005). *Empire of the senses: The sensual culture reader*. Oxford: Berg.
- Ingold, T. (2007a). *Lines: A brief history*. Abington, UK & New York: Routledge.
- Ingold, T. (2007b). Materials against materiality. *Archaeological Dialogues*, 14, 1–16.
- Ingold, T. (2010). The textility of making. *Cambridge Journal of Economics*, 34, 91–102.
- Ingold, T. (2011). *Being alive: Essays on movement, knowledge and description*. NY & London: Routledge.
- Ingold, T. (2013a). *Making*. NY & London: Routledge.
- Ingold, T. (2013b). Of blocks and knots. *Architectural Review*, 234(1400), 26–27.
- Jensen, M. V. (2005). An anthropological move towards tangible interaction design. *Nordic Design Research*, 1. <http://www.nordes.org/opj/index.php/n13/article/view/250/233>.
- Knight, T. W. (1989). Transformations of *De Stijl* art: the paintings of Georges Vantongerloo and Fritz Glarner. *Environment and Planning B: Planning and Design*, 16(1), 51–98.
- Knight, T. (1999). Shape grammars in education and practice: history and prospects. *International Journal of Design Computing*, 2. <http://faculty.arch.usyd.edu.au/kcdc/ijdc/vol02/papers/knightFrameset.htm>.
- Lehmann, A.-S. (2015). The matter of the medium. Some tools for an art theoretical interpretation of materials. In C. Anderson, A. Dunlop, & P. H. Smith (Eds.), *The matter of art: Materials, technologies, meanings 1200–1700* (pp. 21–41). Manchester: Manchester University Press.

- Malafouris, L. (2004). The cognitive basis of material engagement: where brain, body and culture conflate. In E. DeMarrais, C. Gosden, & C. Renfrew (Eds.), *Rethinking materiality: The engagement of mind with the material world* (pp. 53–62). Cambridge: McDonald Institute Monographs.
- Manick, A., & Owen, A. (2013). Bringing back something fine. In E. E. Hirschler, & T. A. Carbone (Eds.), *John Singer Sargent watercolors* (pp. 205–223), (Museum of Fine Arts Boston and Brooklyn Museum).
- McCormack, J. P., Cagan, J., & Vogel, C. M. (2004). Speaking the buick language: capturing, understanding and exploring brand identity with shape grammars. *Design Studies*, 25(1), 1–29.
- Muslimin, R. (2010). Interweaving grammar: reconfiguring vernacular structure through parametric shape grammar. *International Journal of Architectural Computing*, 8(2), 93–110.
- Noë, A. (2004). *Action in perception*. Cambridge, MA: The MIT Press.
- Ormand, R. (2013). Sargent and watercolor. In *John Singer Sargent watercolors* (pp. 15–25). Museum of Fine Arts Boston and Brooklyn Museum.
- Saeedloei, N., & Gupta, G. (2010). *Timed definite clause ω -grammars*. Technical Communications of the International Conference on Logic Programming (Edinburgh) 212–221.
- Sensoaesthetic Materials. Institute of Making at the University College London. Retrieved from <http://www.instituteofmaking.org.uk/research/sensoaesthetic-materials>.
- Sherry, K. A. (2013). Approaching the modern. In *John Singer Sargent watercolors* (pp. 181–183). Museum of Fine Arts Boston and Brooklyn Museum.
- Stiny, G. (1996). Useless rules. *Environment and Planning B: Planning and Design*, 23(2), 235–237.
- Stiny, G. (2006). *Shape: Talking about seeing and doing*. Cambridge, MA: The MIT Press.
- Stiny, G., & Gips, J. (1972). Shape grammars and the generative specification of painting and sculpture. *Information Processing*, 71, 1460–1465.
- Stiny, G., & Mitchell, W. J. (1980). The grammar of paradise: on the generation of Mughul gardens. *Environment and Planning B*, 7, 209–226.
- Urton, G. (2003). *Signs of the Inka Khipu: Binary coding in the Andean knotted-string records*. Austin, TX: University of Texas Press.
- Wentworth, N. (2004). *The phenomenology of painting*. Cambridge, UK: Cambridge Univ. Press.