CHAPTER 5

A FRAMEWORK OF CAUSE AND EFFECT

Science may be described as the art of systematic over-simplification.
K. Popper, as quoted in *Observer* (London, 1 Aug. 1982)

The overriding purpose of the ethnographic studies described in the previous chapter was to bring into sharper focus the costs and benefits of AV technology in general, and of the sociocultural constructivist pedagogical approach in particular. As we have seen, on a practical level, these studies offer several recommendations for how best to implement AV construction assignments in an undergraduate classroom. For example, the assignments should not include an input generality requirement; they should not necessarily require students to implement their AVs on a computer; and they should provide opportunities for students to present paper-and-pencil storyboards to their instructor and peers for feedback and discussion.

On a more theoretical level, these studies lay a solid empirical foundation on which to develop a specific set of hypotheses with respect to how and why algorithm visualization artifacts might be effective pedagogical aids. In an effort to formulate such a set of hypotheses, I ask three framing questions in this chapter:

1. What are the central factors that influence (cause) the effectiveness of pedagogical exercises involving AV artifacts?

2. What are the precise effects of those factors? In other words, what measures can be used to gauge the benefits of the factors?

3. What are the linkages between factors and measures? In other words, what factors or combination of factors cause what effects or combination of effects?

By answering these questions in light of the ethnographic studies presented in the previous chapter, this chapter develops a framework of cause and effect for AV effectiveness. The framework expands on and refines the repertoire of factors, measures, and hypotheses that have been explored by past empirical research. As discussed in Chapter 2, past experiments have focused squarely on causal factors suggested by various versions of EF Theory, including Strong EF, and the Individual Differences, Dual-Coding, and Learner Involvement versions of Weak EF. Likewise, past experiments have explored a narrow range of knowledge transfer measures—primarily tests of procedural and conceptual understanding. In contrast, while still hypothesizing about EF Theory's notion of understanding and recall, the framework presented here includes additional hypotheses that EF Theory simply could not predict. These additional hypotheses tailor the sociocultural constructivist position so that it applies to the particulars of algorithms learning. The result is a richer characterization of AV effectiveness—one that, in broadening its focus to the community of practice, goes beyond the confines of individual cognition to which EF Theory limits itself.

This chapter begins by describing, in greater detail, the five causal factors and four measures included in the framework. I then discuss the centerpiece of the framework: a series of four hypotheses that link cause and effect. Finally, I summarize the framework, and introduce two important research questions it raises. These research questions motivate the two alternative research directions—a series of experiments, and a prototype AV system—pursued in Chapters 6 and 7.

5.1 AV Effectiveness Factors

In the ethnographic studies reported in the previous chapter, six factors stood out as strongly influencing the effectiveness of AV-based pedagogical exercises. These factors appear as leaf nodes in the taxonomy presented in Figure 22. The subsections that follow elaborate on each of the factors in the taxonomy, drawing from both the findings of the ethnographic fieldwork, and from related research.

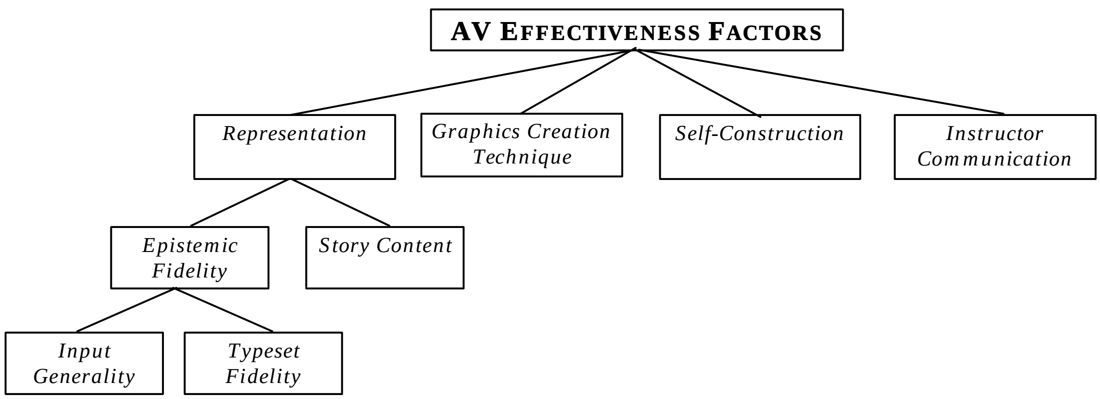

Figure 22. A Taxonomy of AV Effectiveness Factors

5.1.1 Representation Factors

The *representation* factors relate to the nature of visual representations used within AV-based pedagogical exercises; this includes their functionality, appearance, and content. The following subsections describe *input generality, typeset fidelity,* and *story content,* three factors relating to AV representation.

5.1.1.1 Input Generality and Typset Fidelity

As discussed in Chapter 2, the *epistemic fidelity* of an AV is the extent to which it provides a faithful rendition of the algorithm it is intended to illustrate in terms of an expert's mental model. Two factors influence the level of an AV's epistemic fidelity:

1. *Input generality*—Does the AV depict the algorithm for general input?

2. *Typeset fidelity*—Does the AV have the polished, computer-generated look typical of AVs that appear in textbooks?

As discussed in Chapter 2, both of these characteristics are direct consequences of *direct generation*, the AV-creation technique that nearly all AV technology embraces, including the AV technology (Samba; Stasko, 1997) used in the ethnographic studies.

Recall that a key observation from the ethnographic studies was that students not only spent inordinate amounts of time on the animation assignments, but also engaged in activities that were irrelevant and distracting with respect to the focus of the algorithms course in which they were enrolled. The reason for this, as I argued in the last chapter, was that students' AVs were required to have both *input generality* and high *typeset fidelity*. Because students had to worry about getting their animations to work for general input, and because they spent a lot of time tweaking their AVs so that they had high typeset fidelity, they ended up spending a lot of time on programming tasks that were unrelated to the actual algorithm they were trying to illustrate.

By contrast, when we eliminated input generality and typeset fidelity requirements in Study 2, we saw students spend not only significantly less time overall on the assignment, but also a higher percentage of their overall time on activities that required them to focus on the algorithms they were animating. This finding suggests that, in the interest of maximizing students' time on relevant, potentially beneficial activities, student-constructed AV assignments would do well not to require input generality and typeset fidelity.

5.1.1.2 Story Content

As reported in Chapter 4, a small minority of student groups (13%) elected to construct AVs based on stories or scenarios, in which real or fictitious human beings are engaged in some problem-solving venture. It turned out that AVs with story content had two main advantages over AVs that were not based on stories. First, students who constructed stories appeared more motivated, enthusiastic, and engaged. They appeared to take great pleasure in coming up with original stories, and in building AVs around those stories. Second, student presentations of story-based AVs clearly captured the attention of their audience, and led to more lively discussions, than did AVs that were not based on stories.

In addition to these observed benefits of story-based AVs, one can cite a more speculative advantage of story-based AVs. Several studies in the psychological literature (see, e.g., Bower & Clark, 1969; Hill, Allen, & McWhorter, 1991) establish an empirical basis for the value of stories as *mnemonic devices*—that is, techniques that help to improve one's memory of items to be learned. In these studies, the construction of personally-meaningful stories containing words to be learned consistently led to improved recall not only of the words themselves, but also of the order of the words. Researchers account for this improvement by hypothesizing that personally-meaningful stories create a highly-interconnected network of vivid images that cue sequential recall of the words: "Recall of the general story theme cues the initial sentence, and recall of the sentence cues the target words within the sentence" (Hill, Allen, & McWhorter, 1991, p. 484).

While algorithms do not contain *words* per se, they do contain *procedural steps* that must be sequentially executed in order to compute a set of outputs from a set of inputs. Thus, given the results of studies of story mnemonics, it seems reasonable to hypothesize that constructing a personally meaningful story whose storyline maps to the procedural steps of an algorithm to be learned will lead to improved recall of the algorithm's procedural behavior. I shall return to the question of what "improved recall" might mean later on in the chapter.

5.1.2 Graphics Creation Technique

Like most algorithm animation packages, the Samba package used in the ethnographic studies requires one to specify the position and appearance of objects within an animation in terms of (quantitative) Cartesian coordinates. For example, to position an object within an animation. one must specify two real-numbers—an x-coordinate and a y-coordinate. Recall that, in Chapter 2, I labeled this kind of graphics creation technique *quantitative graphics*. In contrast, in constructing their storyboards, students typically created objects by either directly cutting them out of construction paper, directly sketching them on a page, or by using a direct-manipulation drawing editor. Likewise, students typically positioned objects in their storyboard by direct placement—using a hand, pen, or mouse. Let us call this alternative graphics creation technique *direct graphics*.

Observations made in Study II indicate that, in addition to *typeset fidelity and input generality, graphics creation technique* also greatly influences the relevance of student activities. When students created animations in Samba, they spent significant amounts of time figuring out the proper dimensions and coordinates of the objects in their animations. By contrast, when students used low-tech materials to create their homemade animations, students no longer had to focus on graphics layout. Instead, they were able to concentrate more extensively on other issues, such as the procedural behavior of the algorithms they were animating.

5.1.3 Self-Construction

A primary objective the ethnographic studies was to explore the benefits of AV construction assignments. Given this, it should not come as a surprise that *self-construction*—the construction of AVs by *students*, as opposed to experts—emerged as a significant factor in these studies. The more interesting question posed and partially answered by these studies relates to the nature of the benefits of self-construction: In what ways might self-construction be beneficial? In the studies, three key benefits of self-construction emerged.

The first key benefit has to do with the *process* it promotes. AV construction assignments engage students in an exploratory, creative process of constructing and refining personally-meaningful representations. According to cognitive constructivism (see Chapter 3), this process of active learning is far superior to passively watching an AV, for it enables learners to actively construct for themselves the meaning and significance of the algorithms they are learning.

A second key benefit also stems from the process promoted by AV construction. The ethnographic studies suggest that the self-construction process gives students a personal stake in the algorithms they are learning; they become *vested* in the algorithms and the problems they are designed to solve. Articulating the sociocultural constructivist position, Lave (1997) labels this phenomenon *ownership*, and emphasizes its importance in the learning process: "It is not possible [for students] to resolve problems that are not, in some sense, their own" (p. 33).

A third key benefit relates to the *products* of self-construction: animated representations of algorithms. The ethnographic studies indicate that student-constructed representations provide students with valuable resources for making their understanding of an algorithm accessible to others. And, as I explain below, student-instructor conversations involving such animated representations appear to play an integral role in helping students to learn algorithms.

5.1.4 Instructor Communication

As just discussed, a key benefit of students' constructing their own AVs is that it helps students to make their conceptions and misconceptions of an algorithm accessible to others. The ethnographic studies demonstrate that, in subsequent discussions, student-constructed AVs appear to narrow the gap between expert and learner perspectives. An instructor can use a student's AV as a resource for understanding the student's perspective, and as a basis for clarifying concepts. A student can use her own AV as resource for explaining her perspective, asking questions, and interpreting an instructor's feedback and guidance.

Notice that, without subsequent interaction with others (most importantly with their instructor, but also with their peers), students miss out on an opportunity to obtain such valuable feedback and guidance on their emerging understanding. As a consequence, they have no way of knowing whether their view of the algorithm is on track; their misconceptions remain misconceptions, and their insights are not validated. Thus, as a follow-up to the self-construction process, student-instructor interaction would appear to figure prominently in the effectiveness of AV-based pedagogical exercises.

5.2 AV Effectiveness Measures

The previous section identified several factors that, in light of the results of the ethnographic studies, would appear to play a significant role in determining the effectiveness of pedagogical exercises involving AV artifacts. Yet, a fundamental question remains: Effective in what sense? Indeed, since the focus of the previous section was *cause*, I refrained from describing the effects in detail, or elaborating on the way in which the effects might be measured. In this section, I take a closer look at four key *effects* (which might also be called *measures*, or *benefits*) of AV-based pedagogical exercises that are brought about by the factors discussed in the previous section. These four measures are summarized in the taxonomy of Figure 23, and discussed in detail in the remainder of this section.

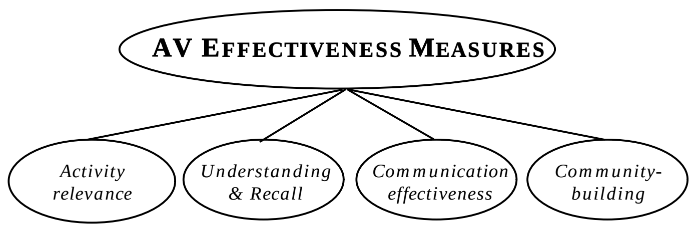

Figure 23. A Taxonomy of AV Effectiveness Measures

5.2.1 Activity Relevance

Pedagogical exercises require students to engage in activities that can be more or less relevant to the particular course in which the pedagogical exercises are enlisted. For example, in the ethnographic studies, I found that AV construction assignments that require students to construct high epistemic fidelity AVs promote activities that are irrelevant to an undergraduate algorithms course, including graphics programming and graphics touch-up. On the other hand, the studies showed that, when asked to construct low epistemic fidelity storyboards, students engage in a higher concentration of relevant activities, including group design discussions and studying.

Thus, an important effect of pedagogical exercises is the relevance of the activities they promote, or their *activity relevance*. But what constitutes a relevant activity? Clearly, there exists no absolute measure of relevance. Rather, one must first conduct a field study in order to obtain as accurate a picture as possible of what students are up to, and then evaluate each student activity individually vis-à-vis the objectives of the course. Based on such an evaluation, one can then estimate an overall percentage of time students' spend on relevant activities versus irrelevant activities. The higher the percentage of relevant activities promoted by a pedagogical exercise, the better.

5.2.2 Understanding and Recall

Thus far, the discourse on AV effectiveness has been preoccupied with a single gauge of effectiveness: *understanding*. This measure endeavors to assess how well a student learns an algorithm, upon engaging in a pedagogical exercise involving AV technology. As we saw in Chapter 2, past AV effectiveness experiments have used written post-tests to measure such understanding. The post-tests have traditionally included two kinds of questions, each designed to measure a distinct form of understanding:

Procedural understanding: Does the student understand *how* the algorithm works?

Conceptual understanding: Can the student make generalizations about the algorithm's behavior, including how efficiently it runs?

Past experiments have found that AV technology appears to have the greatest impact on procedural understanding (see, e.g., Byrne, Catrambone, & Stasko, 1996; Lawrence, 1993, ch. 9). This result is not surprising, given that AVs tend to illustrate how algorithms work, but not how efficiently or why.

Within the community of computer science educators, algorithm visualization tends to be regarded as a means to an end, with the desired end clearly being understanding: How well does the student *know* the algorithm? Recognizing the importance of this end, AV technologists have staged experiments that have focused squarely on measuring *algorithm understanding*. Indeed, for many instructors considering the adoption of AV-based pedagogical exercises, the desired proof-in-the-pudding is a well-designed experiment that demonstrates that AV-based pedagogical exercises promote better procedural understanding of an algorithm than alternative pedagogical exercises.

My fieldwork did not focus on assessing students' understanding of the algorithms for which they constructed and presented AVs. At the same time, the studies did not furnish evidence that speaks *against* the possibility that the AV construction assignments could have led students to develop an improved understanding of those algorithms. Because of the traditional importance of this measure within the computer science education community, and because the ethnographic studies do not rule the measure out, it seems an important one to include in this framework of cause and effect. The only looming questions have to do with focus and operationalization: *What* exactly should we measure, and *how* should we measure it?

The results of past research, including my own observations, suggest that we would do well to refine the traditional operationalization of understanding in terms of procedural and conceptual knowledge. In constructing AVs that are supposed to illustrate *how* an algorithm works, students tend to focus intently on the *procedural* behavior of the algorithm: the sequence of steps it executes to transform inputs into outputs. Thus, at least in situations in which students do not discuss their AVs with an instructor, it seems safe to jettison *conceptual* understanding as a potential effect of both AV construction exercises, and of any pedagogical exercises that involve viewing an AV that illustrates *how* an algorithm works.

With the focus of the measure narrowed to procedural understanding, the question of how to measure procedural understanding arises. Given that AV-based pedagogical exercises require students to engage in activities in which they explore the step-by-step behavior of an algorithm, asking students to trace through that step-by-step behavior for a novel input data set would seem to be a suitable evaluation exercise. The accuracy of their traces with respect to an error-free trace, as well as the speed with which they perform the traces, could then serve as reasonable measures of procedural understanding.

Moreover, given that AV-based pedagogical exercises require students either to construct a graphical representation of the procedural steps of an algorithm (AV construction exercises), or to view a graphical representation of the steps for a variety of input data (AV viewing and interaction exercises), it seems reasonable to expect that such exercises would help students to *recall* the procedural steps of the algorithm. In the case of AV construction exercises, this effect might be predicted by past research into the value of stories as mnemonic devices, as discussed above. In the case of AV viewing exercises, on the other hand, this effect might be predicted by past research into the value of *imagery* as a mnemonic device (see, e.g., Paivio, 1969). To measure recall, one could ask students to implement the algorithm in a programming language, and then compare their algorithm against an algorithm that is known to be correct.

5.2.3 Effective Communication

As discussed earlier, a key benefit of AV construction assignments is that they require students to *represent* their understanding, and thereby to make it accessible to others. In subsequent presentation sessions, students' self-constructed representations enable students and instructors to bridge the gap between their perspectives—to develop a shared understanding of an algorithm. Recognizing the importance of representations in mediating such learner-expert interaction, Roschelle (1990) defines *symbolic mediation* as the use of a representation "as a resource for managing the uncertainty of meaning in conversations, particularly with respect to the construction of shared knowledge" (p. 1). On this view, the extent to which a representation is able to serve as a mediational resource determines, in large part, the effectiveness of the communication about the target concepts.

A second aspect of effective communication is that the concepts discussed in such conversations should actually be *target* concepts, as opposed to concepts that are peripheral to the course. For example, in the ethnographic studies, I found that students' Samba presentation sessions primarily generated conversations *about implementation details*—how a given feature of an animation was implemented, and what implementation difficulties were encountered, and the like. I also discovered that students' storyboard presentations generated conversations that were more sharply focused on issues relevant to an algorithms course, such as the aspects of an algorithm that are important to illustrate. Arguably, the conversations about implementation details did not address the target concepts of the course, whereas the conversations about important aspects of an algorithm to illustrate did.

How does one measure these two aspects of effective communication: mutual intelligibility and topic relevance? *Conversation analytic techniques*, such as those used by Suchman (1987), scrutinize the structure and content of conversations in minute detail—utterance by utterance. The goal is to detect *communication breakdowns*—points in the communication at which a shared understanding is lost—as well as the subsequent repair of such breakdowns. Through such analysis, one can develop a fine-grained account of communication efficacy, firmly grounded in an empirical record (a transcript of the conversation). While such an account is qualitative, it nonetheless can provide insight into the extent to which representations serve as mediational resources in the establishment of shared understanding between conversational participants.

On the other hand, topic relevance can be assessed using the same general technique that I proposed to assess activity relevance. In particular, one can first obtain a fine-grained account, preferably a full transcription, of the conversations that take place during AV presentation sessions. Next, on an exchange-by-exchange basis, one can determine the relevance of the exchange with respect to the objectives of the course. Finally, one can compute topic relevance by determining the percentage of exchanges that were relevant, as opposed to irrelevant.

5.2.4 Community-Building

At a macro level, what is going on within an undergraduate algorithms course? As discussed in Chapter 3, sociocultural constructivism suggests that a distinct *community of practice* is actually in the process of reproducing itself. This reproduction takes place as newcomers become old-timers by participating, in increasingly central ways, in the practices of the community.

Recall that, in Chapter 3, I labeled the community that is in the process of reproducing itself through undergraduate algorithms courses the "Community of Schooled Algorithmaticians" (COSA). My fieldwork suggests that a well-defined participation structure was, in fact, in place in the classes I observed. Old-timers included the course instructor, and, to a lesser degree, the teaching assistant. Newcomers included the students enrolled in the course. Each of these actors participated according to established norms: The instructor participated through such activities as lecturing, holding office hours, issuing assignments, and giving exams, whereas students participated through such activities as attending lectures, studying the text, doing assignments, and taking exams.

If one accepts the sociocultural constructivist premise that, within an undergraduate algorithms course, a community of practice is in the process of reproducing itself, a new potential benefit of AV-based pedagogical exercises emerges. By providing occasions on which the community can come together to participate in meaningful activities, these exercises might contribute to the building up of the community, as well as to helping students to advance more rapidly from peripheral to central participation within the community.

But how might one measure such things as the existence of a community, and level of community membership? These are slippery notions that do not appear to lend themselves to measurement, at least not in a quantitative, operational sense. To be sure, it is not enough simply to measure individual performance in an activity that is relevant to the community, and to conclude that higher performance scores indicate a higher level of membership in the community. For one thing, such an approach assumes some sort of absolute that simply does not exist within a community, where determinations of knowledge and good performance are largely a matter of agreement (see, e.g., Boster, 1985). Moreover, such an approach overlooks the importance of *identity* (the way in which one views oneself, and is viewed by others, with respect to the community) in determining level of community membership. Indeed, as sociocultural constructivism emphasizes, "learning and a sense of identity are inseparable: They are aspects of the same phenomenon" (Lave & Wenger, 1991. p. 115; see also pp. 52–53).

In sum, a potential benefit of AV-based pedagogical exercises is the extent to which the reproduction of the COSA is fostered, as well as the extent to which students come to participate, in increasingly expert ways, within the COSA. While this effect appears difficult, if not impossible, to measure (especially in a quantitative sense), it is nonetheless important, since an implicit goal of any algorithms course is to assist students in becoming competent members of the COSA.

5.3 Hypotheses

The previous two sections outlined the central factors and measures in the AV effectiveness equation. One important question remains: What factors lead to what measures? Figure 24 puts the previous two sections together by causally linking factors to measures. If one groups these linkages by effect (the "Measures" column in Figure 24), four specific hypotheses emerge, each of which connects one or more causal factors to one of the four measures discussed in the previous section. In this section, I discuss these four hypotheses, and offer provisional explanations for them.

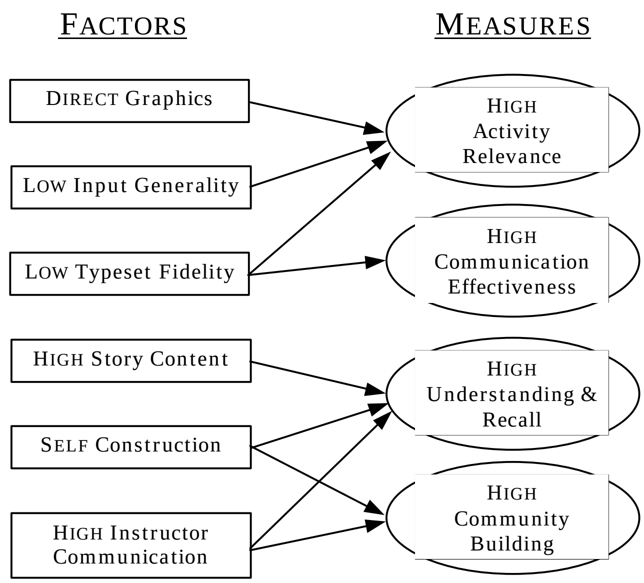

Figure 24. Graphical Summary of the Framework

5.3.1 The Activity Relevance Hypothesis

The Activity Relevance Hypothesis speculates a causal link between epistemic fidelity and activity relevance, as illustrated in Figure 25.

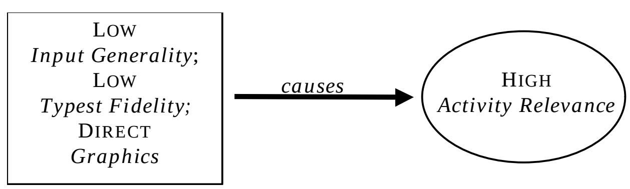

Figure 25. The Activity Relevance Hypothesis

As already discussed, when students construct *one-shot* (i.e., low input generality), low *typeset fidelity* AVs, they tend *not* to get bogged down in the (irrelevant) implementation details that they typically encounter when they construct high input generality, high typeset fidelity AVs. As a result, not only do they spend significantly less time overall on AV construction, but they also spend a higher percentage of their time on activities that are relevant to the course—that is, activities that require them to focus them on the procedural behavior of the algorithms they are animating.

5.3.2 The Communication Effectiveness Hypothesis

The Communication Effectiveness Hypothesis speculates the existence of a causal connection between typeset fidelity and communication effectiveness, as illustrated in Figure 26.

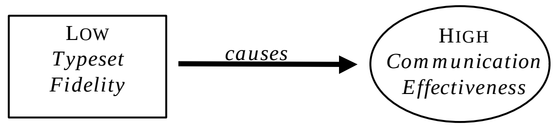

Figure 26. The Communication Effectiveness Hypothesis

In studies of architect-client interaction, rough, unpolished sketches have been shown to generate increased interaction regarding high-level design issues (see, e.g., Schumann, Strothotte, Raab, & Laser, 1996). Likewise, this hypothesis reasons that the rough, unpolished nature of low typeset fidelity AVs should encourage students and instructors to concentrate on the high-level issues behind the AVs, rather than on the aesthetics of the AVs, or on the details of how they were implemented. This shift in focus constitutes an increase in *topic relevance*, one of the two components of effective communication.

This hypothesis also reasons that the other component of effective communication, *shared understanding*, is enhanced by low epistemic fidelity AVs. In particular, as discussed in Chapter 4, two main features of AVs appear to influence their utility as mediational resources:

1. *dynamic mark-up and modification*—an ability to be annotate, mark up, and modify an AV in response to clarifying questions; and

2. *execution control*—an ability to fast-forward, rewind, step through (at a flexible pace), and jump around within an AV.

Whereas low epistemic fidelity AVs can be easily annotated, marked up and modified, the high epistemic fidelity AVs created in systems like Samba cannot; modifying high epistemic fidelity AVs typically requires changing code, an activity that is not feasible within a presentation session. Furthermore, because they are under the complete control of their presenters, low epistemic fidelity AVs support a more flexible execution model; presenters can jump ahead in an AV, go back to a previous point in an AV, and step through an AV in response to the dynamics of the interaction they are having with their audience. In contrast, high epistemic fidelity AVs do not afford nearly as much flexibility; presenters are typically locked into a “tape recorder” model that supports only starting, stopping, and stepping (see Brown, 1988, p. 65).

5.3.3 The Understanding and Recall Hypothesis

The Understanding and Recall Hypothesis posits that story content, self construction, and instructor communication are causally linked to recall and understanding, as illustrated in Figure 27.

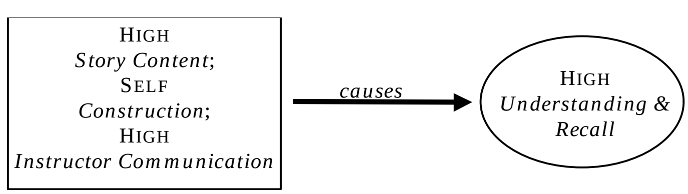

Figure 27. The Understanding and Recall Hypothesis

As discussed above, experiments have shown story construction to be an effective mnemonic device—one that helps people remember not only a list of words, but also the order of the words in the list.

This empirical result suggests that the construction of a story that models the procedural steps of an algorithm should help one to recall the procedural behavior of the algorithm. Moreover, if one defines understanding as an ability quickly and accurately to trace through the procedural steps of an algorithm for novel input data, then one would expect understanding to go up as well, since improved recall of the algorithm's procedural steps should lead to increased speed and accuracy in tracing exercises.

A variety of explanations might account for the value of self-construction in promoting understanding and recall. As briefly discussed above, the sociocultural constructivist explanation emphasizes the role of AV construction assignments in giving students a sense of *ownership* in the algorithms for which they construct AVs (see Lave, 1997). Textbooks and lectures tend to own algorithms by presenting and analyzing them in vivid detail. For example, a typical textbook might provide a pseudocode description of an algorithm, perhaps augmented with a visualization, and then proceed to analyze the efficiency and prove the correctness of the algorithm. Since the textbook appears already to have figured out everything about the algorithm, it appears to *own* the algorithm. As a consequence, students may be pushed away, and lose incentive to explore the algorithm. By allowing students to create, develop, and instantiate their own representations of an algorithm, however, AV construction assignments shift ownership of the algorithm from the textbook (and instructor) to the student. Such ownership *vests* students in the algorithm; they gain a personal stake in its meaning and significance. As a byproduct, their recall and understanding of the algorithm improves.

Also briefly discussed above, cognitive constructivism offers an alternative explanation that emphasizes the role of the AV construction process in facilitating students' active construction of their own understandings. In particular, by constructing and instantiating graphical models of an algorithm, students are required to actively articulate and put to the test their conceptualizations of the algorithm's procedural behavior. In the process, they must reorganize and adapt their understanding of the algorithm's procedural behavior so as to resolve any inconsistencies or holes that they notice in the representations they are building.

While constructing an AV may improve a student's understanding and recall of the underlying algorithm, it does not guarantee that the understanding so developed approaches that of an expert. Through a subsequent presentation session with an instructor, however, a student has the opportunity to gain crucial feedback and guidance. Such feedback and guidance can help the student to refine her understanding of the algorithm, leading to a view of the algorithm that more closely resembles that of the instructor.

Note that the causal link posited between student-instructor interaction and understanding and recall applies to *any* AV, regardless of whether it is constructed by a student or constructed by an expert. In the case of an AV constructed by an expert, student-instructor interaction plays an essential role: it enables students and instructors to *negotiate* the meaning and significance of the AV, neither of which is self-evident (see, e.g., Suchman, 1987). In the case of an AV constructed by a student, on the other hand, this hypothesis predicts a positive interaction effect involving self-construction and interaction. In particular, students who construct their own AVs place themselves in a position in which they can maximally benefit from subsequent interaction they have with an instructor regarding their AVs. Thus, the interaction they have with an instructor contributes more to improved understanding and recall than it otherwise would.

5.3.4 The Community-Building Hypothesis

The Community-Building Hypothesis suggests that the combination of self-construction and interaction within AV-based pedagogical exercises leads to the building up of a community, as illustrated in Figure 28.

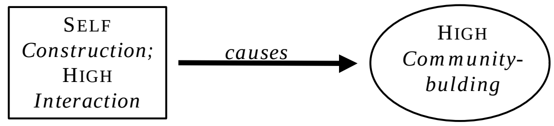

Figure 28. The Community-Building Hypothesis

This hypothesis accounts for the effectiveness of AV technology in terms of the sociocultural constructivist view of learning introduced in Chapter 3. As discussed there, AV artifacts can be seen as facilitating three key forms of participation within the community that is in the process of reproducing itself through undergraduate algorithms courses (*viz.*, the *COSA*):

1. interpreting and making sense of graphical representations of concepts and themes surrounding algorithms (*AV reading*);

2. constructing such representations (*AV construction*); and

3. using such representations as the basis for presentations and discussions (*AV-mediated communication*).

Notice that AV construction assignments provide opportunities for students to participate in all three of those activities, whereas traditional AV-based pedagogical exercises provide opportunities to participate in only one of them (AV reading). Therein, according to sociocultural constructivism, lies the primary benefit of self-construction and interaction: they provide access to increasingly central (expert) forms of participation—forms of participation to which students have not traditionally had access.

AV presentation sessions like the ones described in Chapter 4 yield a secondary benefit as well: opportunities for observation and feedback. The interaction between students and instructors that takes place provides important clues to appropriate ways of presenting, discussing, and commenting on visual representations of algorithms. Thus, in addition to participation, observation would also seem to facilitate students' centripetal movement in the community—from the outer, peripheral layers of the "community onion," to the inner, core layers of the "community onion."

It is important to emphasize that this process of advancement through the participation structure defined by a community not only enables the community to reproduce itself, but also, on the sociocultural constructivist view, *constitutes learning in itself*. Observe that this community-based notion of learning differs markedly from the conventional, cognitive-based notion of learning exemplified by the Understanding and Recall measure (see above). Indeed, coming to participate, in increasingly central ways, within a community of practice encompasses far more than acquiring procedural understanding and recall; it also entails the development of skills and expertise that are beyond those that can be evaluated by conventional written tests, including (a) an ability to effectively explain an algorithm to someone else (which, in turn, involves recognizing what aspects of the algorithm are important to illustrate, and what grain of analysis is appropriate); (b) an ability to critique explanations of algorithmic behavior at appropriate times; (c) an ability to ask appropriate questions about algorithms at appropriate times; (d) interest and motivation to explore algorithms; and (e) confidence in and comfort with the discourse of algorithms.

In sum, sociocultural constructivism provides a theoretical foundation for the hypothesis that self-construction and interaction lead to the maintenance of a community of practice in which students advance toward fuller membership. On this view, AV construction exercises succeed precisely because they provide access both to key forms of COSA participation (AV construction and presentation), and to old-timers and newcomers engaged in presenting and discussing algorithms.

5.4 Summary and Research Directions

This chapter has taken the ethnographic studies presented in the previous chapter as a point of departure for pinpointing key causes and effects in the AV effectiveness equation, and for positing a series of hypotheses that link them together. The goal has been to translate the findings of the ethnographic studies into a specific theoretical position that future research can put to the test.

Resonant with both cognitive and sociocultural constructivism, the theoretical position that emerges out of the framework stresses the value of students' constructing and refining their own personal representations of the material they are learning. At the same time, in line with sociocultural constructivist theory, the position emphasizes the importance of students' participating more centrally in algorithms practice by presenting their self-constructed AVs to their peers and instructor for feedback and discussion. Finally, in radical departure from AV technology's obsession with high epistemic fidelity visualizations, the position underscores the value of low epistemic fidelity visualizations in focusing students' on algorithms and how they work, and in promoting effective communication about algorithms.

This theoretical position can thus be seen as a specialization of sociocultural constructivist learning theory, rather than as a broadening of EF Theory. Indeed, whereas constructivism well accounts for all four framework hypotheses, EF Theory stands a chance of accounting for only one of them: the Understanding and Recall Hypotheses; the other three hypotheses lie beyond its analytical scope. Thus, if the ethnographic findings that inspired the framework can be generalized beyond the particular algorithms course that I studied , sociocultural constructivism should prove to be a more suitable theoretical foundation for guiding future AV research.

In addition to articulating the beginnings of an alternative theoretical position, the framework presented in this chapter has aimed to identify issues that are grist for further investigation. The framework indeed raises numerous research questions, which I enumerate in Chapter 8. Within the scope the scope of this dissertation, however, I limit my focus to two of the most important research issues, which I explore in the next two chapters:

1. *Empirical validation of the hypotheses*. All four of the framework's hypotheses were motivated by qualitative evidence collected in the ethnographic studies presented in the previous chapter. An obvious question to ask is, Can any of these hypotheses be validated through a more rigorous controlled experiment? Because of its similarity to the hypotheses tested in past experiments of AV technology effectiveness, the Understanding and Recall Hypothesis is a prime candidate. In Chapter 6, I propose a series of experiments for validating the Understanding and Recall Hypothesis, and I present an actual experiment that put one component of the hypothesis to the test.

2. Design implications of the hypotheses. The framework's hypotheses have profound implications for the design of AV technology. If one takes these conjectures as constraints on the design space of effective AV technology, what kind of AV artifact emerges? This question, which should be of great interest to AV technologists, is pursued in earnest in Chapter 7, in which I describe the design of a prototype AV artifact rooted in the hypotheses.