CHAPTER 8

CONCLUSION

> We learn 10 percent of what we read, 20 percent of what we hear, 30 percent of what we see, 50 percent of what we both see and hear, 70 percent of what is discussed with others, 80 percent of what we experience personally, 95 percent of what we teach to someone else. W. Glasser [as quoted in (Griss, 1999)]

The advent of interactive, graphical workstations in the early 1980s spurred much initial excitement and enthusiasm among computer science educators. Finally, improved processing speeds and enhanced displays made it possible to produce, with relatively little effort, a clear illustration of how an algorithm works. Even more exciting, these illustrations could be produced automatically as a byproduct of an algorithm's execution, thereby guaranteeing their accuracy. Computer science educators thus had high hopes that, when used as a visual aid in lectures, or when given to students to use as a study aid, computer-generated algorithm animations not only would help students to learn algorithms better, but also would make instructors' jobs easier.

As I have illustrated, AV technology has failed to deliver on its initial promise of better learning and easier teaching. Research studies designed to substantiate the pedagogical benefits of AV have yielded mixed results, and AV technology has failed to see widespread use in computer science education. This has left AV technologists and computer science educators scratching their heads and asking, What went wrong? Why has AV technology failed to deliver?

If the wealth of published AV research in the decade after the introduction of the original interactive AV system (see Brown, 1988) is any indication, that question has been on the mind of many researchers. A review of this work suggests that at least three alternative approaches to addressing the question have been considered:

1. Create better AV technology.
2. Create better pedagogical exercises involving AV technology.
3. Revamp the methods used to evaluate the effectiveness of AV technology.

The thesis for which I have argued here maintains that the theory of effectiveness guiding these three approaches is deficient. That theory of effectiveness, which I have labeled Epistemic Fidelity Theory, holds that the value of AV technology lies in its excellent ability to capture an expert's mental model of how an algorithm works, thus leading to the robust and efficient transfer of that mental model to students who view the AV. My argument has been that any approach that merely alters surface features of AV technology design, pedagogy, and evaluation, while remaining faithful to EF Theory as an underlying theoretical foundation, is bound to fail. Rather, if we are to make progress, we must first address the problem at its roots—by adopting a more appropriate guiding theory of effectiveness.

To explore this thesis, I have set about to motivate, construct, and refine an alternative theoretical foundation for AV technology, on which further research can build. At each decision point in this theory-building venture, my main objective has been to base my choices, to the greatest extent possible, on empirical findings, rather than on intuition. The first such decision came when I selected constructivism as a more appropriate guiding theory. Through a meta-study of past studies of AV effectiveness, I found that active learner involvement appeared to be the most significant factor in successful learning outcomes. Since constructivism furnishes a well-established, coherent account of why this might be, I opted to use it as my starting point. Moreover, observing that the target skills of an undergraduate algorithms course are inherently *social*, I chose to adopt the sociocultural version of the theory, which views the locus of learning not at the level of the individual, but at the level of the community of practice, and which sees the value of AV technology not in its ability to transfer knowledge, but rather in its ability to facilitate access to more expert forms of participation in a community of practice.

The second decision came when I had to choose what to do with sociocultural constructivist theory. I decided that taking the theory seriously required a commitment to using ethnographic field techniques to carry out a naturalistic study of the community of practice being reproduced through an undergraduate algorithms course. Only through such a study would I be able to observe the effects of the sociocultural constructivist approach to using AV technology, which invites students to participate as (expert) instructors by using AV technology to construct and present their own AVs.

Traditional AV technology facilitates the construction of *high epistemic fidelity* AVs—that is, AVs that work for general input, and that have the polished look of textbook figures. My ethnographic studies found that traditional AV technology is ill-suited for this pedagogic approach, not only because the construction of high epistemic fidelity AVs requires inordinate amounts of time, but also because it requires students to engage in activities that are irrelevant to the focus of an undergraduate algorithms course. In contrast, when students used simple art supplies to construct and present their own, low epistemic fidelity AVs, they spent significantly less time overall, and the time that they did spend was dedicated to more relevant activities. An additional, unexpected benefit of low epistemic fidelity AVs was that they stimulated more relevant conversations about algorithms than did high epistemic fidelity AVs.

These findings led to a third decision: How to articulate the revised theoretical position? Remaining true to the findings of the ethnographic studies, I opted to express the alternative theoretical position in terms of four hypotheses that link cause to effect:

1. *The Activity Relevance Hypothesis*: Low input generality, low typeset fidelity, and direct graphics cause high activity relevance.

2. *The Communication Effectiveness Hypothesis*: Low epistemic fidelity causes high communication effectiveness.

3. *The Understanding and Recall Hypothesis*: Self-constructing AVs with a story line, and then presenting them to an instructor for feedback and discussion, causes high recall and understanding.

4. *The Community-Building Hypothesis*: Self-construction and high instructor communication cause high community-building.

These hypotheses are the centerpiece of the dissertation; they outline the provisional theoretical position that it proposes. However, far from being the final word on AV effectiveness, the hypotheses raise more questions than they answer. This observation brought me to a fourth decision: Which questions might I answer within the scope of this dissertation, and which ones would be best left to future research? In making this decision, I attempted to choose two research activities that would hold the most promise in demonstrating the plausibility and desirability of the theoretical position.

First, realizing that the community of computer science educators would probably be most receptive to empirical evaluations that consider traditional learning outcomes, I opted to conduct a controlled experiment in which I put to the test a key piece of the Understanding and Recall Hypothesis: On a test of procedural understanding and recall, students who construct their own AVs will outperform students who interact with an AV constructed by an expert. While the experiment failed to demonstrate a significant difference between the two treatment groups, the fact that the students who constructed their own AVs out of simple art supplies slightly outperformed the students who interacted with a predefined AV should be of interest to those instructors looking for a low-overhead way of incorporating AV technology into their courses. Indeed, the result suggests that this simple approach, which requires no technology whatsoever, will help students learn just as well as the traditional, more costly approach of having students interact with computer-based AVs.

Second, recognizing that important implications of the hypotheses for technology design, I decided to build a prototype AV system rooted in the hypotheses. While the hypotheses have implications with respect to *what* tasks system users should be able to accomplish with the system (viz., the creation and presentation of low epistemic fidelity AVs), they offer little guidance with respect to *how* users should accomplish those tasks. Thus, I found myself at final decision point: How to design the system? Once again, empirical data guided my decision; I drew from the key findings of several empirical studies in order to develop the detailed design of the system. As we have seen, the result of the design exercise was SALSA, a high-level interpreted language for programming low epistemic fidelity AVs, and ALVIS, a graphical front end for programming SALSA scripts by direct manipulation.

Having summarized the trajectory this dissertation has taken, I now elaborate on its research contributions.

8.1 Research Contributions

The primary research contribution of this dissertation is the novel theoretical position articulated by the framework of cause and effect presented in Chapter 5. The Community-Building Hypothesis can be characterized as a specialized version of sociocultural constructivism that specifically addresses the pedagogical use of AV technology in an undergraduate algorithms course. Likewise, the Understanding and Recall Hypothesis specializes sociocultural constructivism, which recognizes self-construction and presentation as crucial to learning insofar as they *vest* students in their own learning (Lave, 1997). In addition, the aspect of the Understanding and Recall Hypothesis that states that constructing an AV leads to high recall and understanding can be seen as a specialization of two other existing theories: (a) cognitive constructivism, which views learning as students' active construction of their own understandings, and (b) mnemonic theory, which underscores the value of stories in aiding memory. Finally, in contrast to the Community Building and Understanding and Recall Hypotheses, the Activity Relevance and Communication Effectiveness Hypotheses do not derive from past theoretical positions. Rather, they are quite specific to learning in an algorithms course, identifying the particular conditions under which AV construction and AV presentation will be most effective as learning activities within that context.

The framework hypotheses plainly have important implications for algorithms pedagogy. The second major contribution of this dissertation is a set of guidelines, embodied in the framework hypotheses, that outline a practical (low overhead, low cost) pedagogical approach to incorporating algorithm visualization into an algorithms course. Specifically, the pedagogical approach recommended by the hypotheses models the approach taken in Study II (see Chapter 4): Have groups of students use simple art supplies to construct AVs that illustrate a target algorithm for a few, carefully chosen input data sets, and then ask them to present their storyboards to the class for discussion and feedback.

In the process of arriving at the framework hypotheses, this dissertation makes two secondary research contributions. First, the meta-analysis of past empirical studies of AV effectiveness that I present in Chapter 2 constitutes the first attempt to synthesize and identify trends in the body of work as a whole. Second, the two ethnographic studies that I present in brief in Chapter 4, and in detail in Appendices A and B, are the first of their kind. In particular, they are the first to consider the community aspects of AV technology, and they are the first to do so in a conscientious, systematic way—that is, by using a collection of established research techniques, and by taking care to explain precisely how the techniques were used.

Finally, the two research directions I pursued in response to the framework of cause and effect make two other secondary research contributions. First, the experiment presented in Chapter 6 suggests that the self-construction factor, by itself, may not be strong enough to significantly impact procedural understanding and recall. Second, the system-building effort described in Chapter 7 constitutes the first serious attempt to ground the design of an AV system firmly in empirical data. Moreover, the resulting prototype language and system, SALSA and ALVIS, demonstrate three design features that are absent in existing AV technology:

1. support for a novel way of specifying an AV in terms of spatial logic;
2. a novel control interface that supports both forward and backward execution; and
3. a novel presentation interface that supports dynamic mark-up and modification of an AV, even while it is executing.

As is the case for any substantial research project, the most significant contribution of this work may well be the wealth of research questions it raises for future work. I consider these in the following section.

8.2 Directions for Future Research

The framework of cause and effect presented in Chapter 5 sketches out a provisional theoretical position that clearly requires further development and refinement. To that end, the two research projects presented in Chapters 6 and 7 barely scratch the surface. Indeed, each framework hypothesis invites further exploration, as discussed below.

8.2.1 The Understanding and Recall Hypothesis

Recall that the Understanding and Recall Hypothesis states that student self construction, instructor communication, and AV story content lead to improved procedural understanding and recall. In Chapter 6, I proposed a research program for systematically exploring the Understanding and Recall Hypothesis. Specifically, the research program included a series of three planned experiments, each of which manipulates one of the key causal factors of the Understanding and Recall Hypothesis while holding the others constant. The experiment presented in Chapter 6, which examined the impact of student self-construction, constitutes the initial step in this research program. The other two proposed experiments, which would examine the impact of instructor interaction and story content, remain as key pieces of future research.

In addition, the main result of the experiment of Chapter 6—that self-construction, by itself, does not appear strong enough to effect improved procedural understanding and recall—suggests that it may be important to explore interaction effects in future experiments. For example, one promising 2×2 design (see Table 14) would simultaneously explore the effects of the self-construction and instructor communication factors. This experiment would include four treatment groups: the two included in the experiment presented in Chapter 6 (Self Construction, Active Viewing), as well as two others in which students also had the opportunity to interact with an instructor (Self Construction + Instructor Communication, Active Viewing + Instructor Communication). An analysis of variance could be performed to determine not only if significant differences exist among treatment groups, but also whether there exist interaction effects. The Understanding and Recall Hypothesis would predict an interaction effect between the Self Construction and Instructor Communication factors: Students who construct their own AVs and then discuss them with an instructor will significantly outperform students who interact with a predefined AV and do not have an opportunity to interact with an instructor.

Table 14. A 2 × 2 Design that Explores Self Construction and Instructor Communication

		Self-Construction	
		No	Yes
Instructor Communication	No	12	12
	Yes	12	12

8.2.2 The Activity Relevance Hypothesis

Recall that the Activity Relevance Hypothesis states that students who build low input generality, low typeset fidelity AVs using direct graphics engage in more relevant activities than do students who use quantitative graphics to produce input-general, high typeset fidelity AVs. This hypothesis was informally verified in the ethnographic fieldwork presented in Chapter 4. Specifically, through observation, interviews, and diary analysis, I found not only that the storyboard construction exercises required far less time than did the Samba construction exercises, but that they enabled students to avoid almost completely the irrelevant low-level implementation details in which students became mired during the Samba AV construction exercises.

Plainly, this finding could be verified more rigorously through a systematic, controlled investigation. For example, one could conduct an experimental study in which a sample of algorithms students is divided into two groups: Samba and Storyboard. Pairs of students in the Samba group are required to construct an input general AV in Samba, while pairs of students in the Storyboard group are required to construct an art supply storyboard that works for just a few, carefully chosen input data sets. So that all construction activities could be videotaped for later analysis, students would be required to work on their AVs only in a closed laboratory during scheduled sessions. Post-hoc Interaction Analysis (Jordan, 1995) of the videotapes could be used to develop a classification scheme for student activities, and to determine precisely how much time each pair of students spent doing each activity type. In addition, the relative relevance of each activity type could be determined through consultation with one or more algorithms instructors. This would allow one to conduct statistical tests to determine whether significant differences exist between the two groups with respect to the relevance of the activities in which they engaged.

8.2.3 The Communication Effectiveness Hypothesis

Recall that the Communication Effectiveness Hypothesis states that low epistemic fidelity AVs stimulate more relevant conversations about algorithms than do high epistemic fidelity AVs. In the ethnographic studies presented in Chapter 4, I informally verified this hypothesis through a post-hoc analysis of videotaped footage of the student presentation sessions.

As is the case for Activity Relevance Hypothesis, this hypothesis could clearly be verified more rigorously through a systematic investigation. In fact, one could do so by extending the study of activity relevance just outlined. In particular, after they create their AVs, each pair of students could be required to present their AVs to an audience that includes the entire group of students, along with a course instructor. Once again, all presentation sessions would be videotaped in order to facilitate post-hoc analysis, as follows.

First, through post hoc Interaction Analysis (Jordan, 1995), one could develop a scheme for classifying the topical content of the monologues and conversational exchanges that take place in the presentations; the relative relevance of each topic could be determined in consultation with one or more algorithms instructors. The scheme could then be used as a basis for precisely determining the proportion of time in each presentation that is dedicated to each topic. Ultimately, statistical tests could be used to determine whether significant differences in topic relevance exist between presentations of low and high epistemic fidelity AVs.

Recall that the second measure of communication effectiveness that I proposed in Chapter 5 is *mutual intelligibility*—the extent to which an AV serves as a resource that assists conversational participants in establishing a shared understanding of the algorithms being presented. Here, a modified form of *breakdown analysis* (Doerry, 1995) could be used to determine the amount of communicative breakdown (i.e., points where mutual intelligibility is lost) in each presentation; a statistical test could be used to detect a difference between levels of mutual intelligibility in presentations of low and high epistemic fidelity presentations.

8.2.4 The Community-Building Hypothesis

Recall that the Community-Building Hypothesis causally links self-construction and instructor communication to community-building. Drawing from sociocultural constructivism, the notion of community-building used here refers to the reproduction of the Community of Schooled Algorithmaticians (COSA) through an undergraduate algorithms course. Through that process of reproduction, community newcomers—the students enrolled in the course—gradually move toward full community membership by participating in increasingly expert ways in community activities—namely, by constructing and presenting their own AVs.

While it is unlikely that any one student could gain full membership in the COSA (that is, a level of membership comparable to that of a course instructor) by the end of a single algorithms course, it should certainly be possible to observe tangible differences in a student's level of membership and identity over the course of the academic term. Therein lies the challenge in systematically validating this hypothesis: How does one measure such nebulous concepts as "level of community membership" and "identity?" While I do not claim to have a definitive answer to that question, I can suggest two promising avenues that future research might consider.

First, future research could use a carefully-designed *attitude questionnaire* to estimate the extent to which students' identity is wrapped up in the COSA. Appendix G presents an example of a questionnaire that might be used to gauge attitudes toward an undergraduate algorithms course; I developed it by adapting the "Revised Math Attitude Scale" developed by Aiken and Dreger (1961), and reprinted in (Shaw & Wright, 1967). The methodology for using such a questionnaire would be to give it to two student groups, each of which is enrolled in an alternative offering of an undergraduate algorithms course. In one of the course offerings, students would use AV technology in the traditional way—by interacting with predefined AVs. In the other of the course offerings, students would use AV technology in the way argued for in this dissertation—by constructing and presenting their own AVs. The questionnaire would be administered to both groups twice: once at the beginning of the academic term, in order to assess baseline attitudes, and again at the end of the academic term, in order to gauge students' attitudes at the end of the course. The results of the exit questionnaire could then be compared against the results of the baseline questionnaire in order to compute the relative change in attitudes for each group. Finally, a statistical test could be used to determine whether a significant difference exists between the two groups' attitudinal changes. A statistically significant difference would be grounds for concluding that one pedagogical treatment better facilitates community-building than the other.

Second, and more ambitious, future research might explore the possibility of using Cultural Consensus Theory (Romney, Weller, & Batchelder, 1986) as a means of calculating one's level of community membership. Cultural Consensus Theory recasts one's level of community membership in terms of one's level of *agreement* with community members on matters of importance to the community. To calculate agreement, the theory derives a formal statistical model for assessing the "cultural competence" of the individual members of a community of practice. The basis for such an assessment are the answers that a sample of community informants furnish to a set of questions. The questions must be carefully chosen so that they address a body of knowledge on which the community of practice is assumed to agree. However, unlike the statistical models traditionally applied to test-taking, Consensus Theory's statistical model does not assume an objective truth against which informants' answers are to be measured. Rather, the model uses informants' answers as a basis for constructing a *cultural truth*, according to which informants' cultural competence can then be assessed.[59]

Consensus Theory's statistical model constructs such a cultural truth based on patterns of agreement among informants. The assumption is that "the correspondence between the answers of any two informants is a function of the extent to which each is correlated with the [cultural] truth" (Romney, Weller, & Batchelder, 1986, p. 316). In other words, the most central members of the community, whose cultural knowledge is the most "complete," are highly likely to agree with each other by offering identical answers to the questions. On the other hand, less central members of the community are less likely to agree both with each other, and with central members of the community.

To gauge the ability of AV-based pedagogical exercises to facilitate community-building, one can employ a strategy similar to the strategy I outlined for using the attitude questionnaire. First, divide a sample of computer science students into two groups, each of which enrolls in an alternative offering of an algorithms course. One of the offerings makes use of AV technology in the traditional way, and the other offering makes use of AV technology in the sociocultural constructivist way explored in this dissertation. Next, measure each group's cultural competence twice—once at the beginning of the course to assess their baseline competence, and once at the end of the course to assess their final competence. Finally, statistically compare the two groups' mean change in competence. If a significant difference exists, it would be grounds for concluding that one of the pedagogical treatments better facilitates community-building than the other.

Of course, if Cultural Consensus theory is to be used as the foundation for such a study, one must first design an appropriate instrument for assessing COSA cultural competence. In this dissertation, I have identified three key forms of COSA participation that are mediated by AV technology:

1. *AV reading*—meaningfully interpreting a graphical representation of an algorithm;

2. *AV writing*—constructing one's own graphical representation of an algorithm; and

3. *AV presentation/discussion*—presenting an AV to community members for discussion.

Clearly, if the COSA really is a distinct community of practice, one would expect a high level of agreement among its members with respect to the way in which they perform these tasks. For example, given an AV, one would expect members of the community to "read" the AV in a similar way.[60] Likewise, given an algorithm, one would expect COSA members to "write" similar AVs that describe it—that is, AVs that capture the visualization in terms of a similar semantics.[61] The

---

[59]Hence, the use of the term *informant* (as opposed to, say, *subject*) is deliberate; it underscores the fact that participants in a Consensus Study are *informing* the researcher of their culture, rather than the researcher *subjecting* them to a test.

[60]For empirical evidence that central members of a community of practice read the representations of the community in a similar way, see (Petre & Green 1993), who studied digital circuit designers.

[61]This is, in fact, borne out by recent empirical studies that explored the human visualization of algorithms; see (Chaabouni, 1996; Douglas, Hundhausen, & McKeown, 1995, 1996).

challenge, of course, lies in coming up with a way to characterize agreement in these tasks *quantitatively*, so that results can be fed to Consensus Theory's statistical model.

I propose the following as a technique for quantitatively assessing the level of agreement among COSA members' AV reading and writing activities. The technique draws on the *semantic level analysis* technique first introduced by Douglas, Hundhausen, and McKeown (1995, 1996). Given a set of AVs that describe an identical algorithm, semantic level analysis involves mapping the lexical entities, attributes, and transformations of the AVs to the underlying semantics of the algorithm. Algorithm semantics is expressed in terms of a pseudocode-like description of the algorithm; each mapping between a lexical item of an AV and a variable or statement in the underlying pseudocode-like description is called a *semantic primitive.*

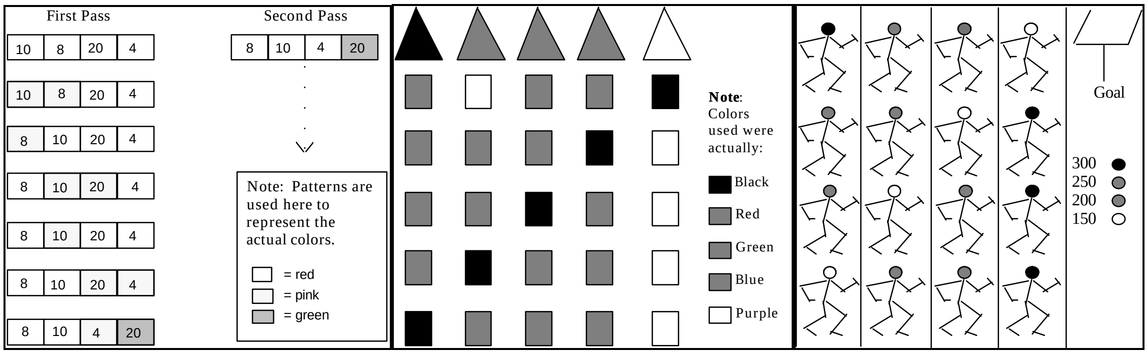
Figure 57 presents an example of the technique taken from the Douglas, Hundhausen, and McKeown studies. In this example, three alternative AVs of the bubblesort algorithm (see Figure 57(a)] observed in the study are semantically analyzed. Figure 57(b) maps the lexical entities and attributes of each AV, while Figure 57(c) maps their lexical transformations. As the example illustrates, while all three AVs differ greatly at a lexical level, they all capture the bubble sort algorithm in terms of a similar set of semantic primitives.

Notice that this semantic level analysis technique makes it possible to quantify the level of agreement between two people on AV reading and writing tasks. With respect to AV reading, two people are said to agree to the extent that they can view an AV and perform a similar semantic-level analysis on the AV. More formally, let s_1 be the set of semantic primitives (i.e., lexical-to-semantic mappings) that informant i_1 gleans from viewing an AV, and let s_2 be the set of semantic primitives that informant i_2 gleans from viewing the same AV. Then the *proportion of agreement* between informants i_1 and i_2 can be defined as the proportion of semantic primitives in s_1 and s_2 that are identical:

$$\textit{Proportion of agreement}(i_1,i_2) = \frac{|s_1 \cap s_2|}{|s_1 \cup s_2|}$$

With respect to AV writing, agreement between two informants can be calculated in an analogous fashion: Two people are said to agree to the extent that the AVs that they construct have semantic primitives in common. More formally, let s_1 be the set of semantic primitives determined to exist in the AV of informant i_1 for algorithm a, and let s_s be the set of semantic primitives determined to exist in the AV constructed by informant i_2 for algorithm a. Then the above equation expresses the proportion of agreement between the AVs of i_1 and i_2.

Clearly, additional research is needed in order to determine whether this approach to quantifying agreement on AV reading and writing tasks will ultimately prove workable. For example, a pilot study I recently conducted suggests that, while the semantic level analysis technique is learnable, people engaged in AV reading tasks tend to use diverse languages to describe the semantic mappings in a given AV. The diversity of their descriptions makes it difficult to determine the extent to which two semantic level analyses are actually in agreement. One possible way to ensure uniformity among two people's semantic level analyses is to require them to work with identical descriptions of both the underlying algorithm, and the AV being read. The former might be in high-level pseudocode, while the latter might be in a high-level AV description language like SALSA.



(a) Snapshots of three AVs of the bubblesort algorithm

SEMANTICS	NUMBER AV LEXICON	COLOR AV LEXICON	FOOTBALL AV LEXICON
Sorting element	Square	Square	Stick Figure
Magnitude of element	Number symbol	Color	Color (as weight)
Array of elements	Contiguous row of squares	Non-contiguous row of squares	Contiguous row of figures
Inner loop pass history	Rows of sorting elements	—	—
Outer loop pass history	Columns of rows	Rows of sorting elements	Rows of sorting elements
Legend explicating ordering on sort elements	—	Triangles with color spectrum	Column of color/player weight pairs

(b) Mappings between lexical entities and attributes of the bubblesort AVs and their semantics

SEMANTICS	NUMBER AV LEXICON	COLOR AV LEXICON	FOOTBALL AV LEXICON
DO outer loop	Start new column of rows	Create new row of squares	Create new row of football players
DO inner loop	Create new row of squares	—	—
a) Reference elements to be compared	Color elements pink	—	Location of football
b) Compare elements (same, <, >)	—	—	Intuitions about how player size relates to running, tackling, and fumbling
c) Exchange elements	Exchange numbers	Exchange colors	Ball carrier advances by tackling next football player in line (thereby exchanging positions with that player)
d) Don't exchange elements	—	—	Fumble football to next player in line
Terminate outer loop	Color square in correct order green	—	—
Terminate Sorting	Ordering of natural numbers, all squares green	Color squares match legend	Players ordered by weight

(c) Mappings between lexical transformations of the bubblesort AVs and their semantics

Figure 57. Example of the Semantic-Level Analysis Technique

8.2.5 The Prototype Language and System

SALSA and ALVIS, the prototype language and system derived from the framework hypotheses (see Chapter 7), clearly constitute work in progress. Much future work remains to be done in order both to refine the design features that they demonstrate, and to explore their benefits in the real world. In particular, future work should focus its efforts in three key areas. First, SALSA and ALVIS are presently implemented just fully enough to illustrate the major design implications of the hypotheses, and to demonstrate their feasibility in practice. In their present state, they fall well short of fully implementing all of the features included in the original conceptual design specification. Some key features that remain to be implemented include (a) the scissors tool, the algorithms for which prove particularly challenging; and (b) the graph and binary-tree s-structs, which would provide users with more options for laying out cutouts.

Second, future research should subject the SALSA and ALVIS prototype to iterative usability testing in order to improve the usability of its design. Ideally, the prototype would be subjected to two or more rounds of usability testing; each round would ideally include three to five target users (undergraduate algorithms students and instructors). Third, it is important to note that ALVIS was originally designed to be used with a *stylus* in order to enable users to manipulate cutouts in a way that more closely models that way in which they would do so in the real world. However, the prototype implementation is not presently set up to work with a stylus. Thus, future research would do well to incorporate a stylus into the prototype, and to explore its potential as ALVIS's input device.

Finally, and perhaps most importantly, SALSA and ALVIS, in their present states, are frail research prototypes that were constructed for the purpose of making a specific point; they were not implemented to be robust systems suitable for use in the real world. Thus, an important direction for future research is to improve the robustness of SALSA and ALVIS, so that they advance beyond frail prototypes, and toward finished systems that can be used in an actual algorithms course. As robust systems, SALSA and ALVIS can serve as the technological foundation for the algorithms class of the future that this dissertation envisions, as explained in the following section.

8.3 A Vision for the Future: The "Algorithms Studio"

Taken as a whole, the research presented in this dissertation points toward an approach to teaching algorithms that differs markedly from the conventional, lecture-based approach that presently predominates on university campuses. Inspired by the research presented in this dissertation, I would like to close by sharing my vision for the future algorithms course.

My vision draws from the instructional model presently used to teach architectural design, which constitutes a prime example of the learning-by-participation model advocated by sociocultural constructivism. Unlike algorithms students, who spend most of their time attending lectures and studying on their own, architectural students spend most of their time with their peers in an architectural studio, where they work on collaborative, directed design projects. Gathered around drafting tables, student groups work out their ideas using drafting paper, pencils, and cardboard models; other students are always around to answer questions, to bounce ideas off, or to discuss design alternatives. In addition, during scheduled review sessions, students present their work-in-progress to their instructor for feedback. During final review sessions, students present their final designs to their instructor and peers.

So, too, could it be in an algorithms course. While lectures would likely remain a necessary part of the curriculum, the length of the frequency of the lectures could be reduced in order to give students more time in the "algorithms studio." There, they would engage in collaborative algorithm design and analysis projects at their "drafting tables"—computer workstations with large displays appropriate for group collaboration. Using a conceptual algorithm design tool like SALSA and

ALVIS, students could explore alternative ways of solving assigned algorithm design problems. As they worked in the studio, peers would always be around to bounce an idea off, answer questions, or discuss possible design strategies. Moreover, students could use ALVIS to present tentative or partial solutions to their peers and instructor—both informally during studio work sessions, and more formally during scheduled review sessions. Finally, in the spirit of the animation presentation sessions that were part of the algorithms courses I studied in my ethnographic fieldwork, students could use a projection device in the studio to present their final algorithm designs to their instructor and peers on a large screen. These review sessions would not only provide instructors with an important basis for assessing students' progress, but they would also, as demonstrated in my ethnographic fieldwork, set up ideal conditions under which to discuss the conceptual foundations of algorithms.

This alternative, studio-based model of algorithms learning could go a long way not only toward getting students more actively involved in their own learning, but also toward promoting the kinds of activities and communication that prove so vital to fostering students' enculturation into the Community of Schooled Algorithmaticans. Indeed, if there is a key idea to take away from this dissertation, it is about the power of using AV technology in a radically different way in algorithms education: not as a means of producing informative displays that accurately convey how algorithms work, but rather as a means of granting students' *access* to the central practices of a community concerned with the conceptual foundations of algorithms.