Journal of Visual Languages and Computing (2002) **13,** 259–290
doi:10.1006/S1045-926X(02)00028-9 available online at http://www.idealibrary.com on IDEAL®

A Meta-Study of Algorithm Visualization Effectiveness

Christopher D. Hundhausen*, Sarah A. Douglas† and John T. Stasko‡

*Laboratory for Interactive Learning Technologies, Information and Computer Sciences Department, University of Hawai'i, Honolulu, HI 96822, U.S.A., E-mail: hundhaus@hawaii.edu, † Human-Computer Interaction Lab, Computer and Information Science Department, University of Oregon, Eugene, OR 97403-1202 U.S.A. and ‡ College of Computing/GVU Center, Georgia Institute of Technology, Atlanta, GA 30332–0280, U.S.A.

Algorithm visualization (AV) technology graphically illustrates how algorithms work. Despite the intuitive appeal of the technology, it has failed to catch on in mainstream computer science education. Some have attributed this failure to the mixed results of experimental studies designed to substantiate AV technology's educational effectiveness. However, while several integrative reviews of AV technology have appeared, none has focused specifically on the software's effectiveness by analyzing this body of experimental studies as a whole. In order to better understand the effectiveness of AV technology, we present a systematic meta-study of 24 experimental studies. We pursue two separate analyses: an analysis of *independent variables*, in which we tie each study to a particular guiding learning theory in an attempt to determine which guiding theory has had the most predictive success; and an analysis of *dependent variables*, which enables us to determine which measurement techniques have been most sensitive to the learning benefits of AV technology. Our most significant finding is that *how* students use AV technology has a greater impact on effectiveness than *what* AV technology shows them. Based on our findings, we formulate an agenda for future research into AV effectiveness. © 2002 Elsevier Science Ltd. All rights reserved

Keywords: Algorithm visualization; Software visualization; Empirical studies of effectiveness; Literature reviews; Meta-analysis

1. Introduction

By graphically representing computer algorithms in action, algorithm visualization (AV) technology aims to help computer science students understand how algorithms work. Since its advent in the late 1970s, AV technology has evolved from batch-oriented software that enable instructors to construct animated films [1]; to highly interactive systems that enable students to explore dynamically configurable animations of algorithms on their own (e.g. [2, 3]); to interactive programming environments that enable students to quickly construct their own visualizations (e.g. [4, 5]). Over the past two decades, this evolution of software has endeavored to enhance computer science education in a variety of capacities. For example, AV software has been used:

1045-926X/02/$ - see front matter © 2002 Elsevier Science Ltd. All rights reserved

- to help instructors illustrate algorithm operations in a lecture (e.g. [2]);
- to help students as they study and learn about fundamental algorithms in a computer science course(e.g. [6]);
- to help instructors track down bugs in students' linked-list programs during office hours (e.g. [7]) and
- to help students learn about the basic operations of an abstract data type in a computer science laboratory (e.g. [8]).

Despite its intuitive appeal as a pedagogical aid, algorithm visualization technology has failed to catch on in mainstream computer science education [4, 9]. While those few educators who are also AV technology developers tend to employ their own AV technology, the majority of computer science educators tend to stick to more traditional pedagogical technologies, such as blackboards, whiteboards and overhead projectors.

Why do computer science educators tend not to use AV technology? Instructors commonly cite several reasons, including the following

- They feel they do not have the time to learn about it.
- They feel that using it would take away time needed for other class activities.
- They feel that creating visualizations for classroom use requires too much time and effort. Note that, in the AV technology literature, this reason is frequently used to motivate new technology that is easier to use, and that supports the more rapid creation of visualizations (see, e.g. [10–12]).
- They feel that it is simply not educationally effective.

All of these reasons certainly contribute, in some way, to AV technology's failure to be adopted. However, the fourth reason—that AV technology does not help students learn algorithms better than conventional teaching methods—stands out as particularly important. Indeed, the underlying purpose of AV technology is to be educationally effective, so there certainly is no reason to adopt the technology if it is not effective.

Given that the underlying purpose of AV technology is to be educationally effective, it is noteworthy that eight extant taxonomic reviews of AV and software visualization technology (see Table 1) have focused largely on system *expressiveness*. In particular, these taxonomies have focused on three main questions:

(1) What kinds of programs can be visualized with a given visualization system?
(2) What kinds of visualizations can a given visualization system produce?
(3) What methods can one use to produce and interact with visualizations?

Notably, in focusing on system expressiveness, these integrative reviews of AV technology have largely ignored a substantial body of over 20 experimental studies that have attempted to substantiate the educational value of AV technology. A casual review of these studies suggests that their results have been, as Gurka [7] aptly puts it, 'markedly mixed'. Indeed, while some of the studies have demonstrated a pedagogical advantage for students using AV technology (e.g. [13, 14]), others have found either (a) no advantage at all (e.g. [15, 16] Chapter 2), or (b) an advantage that can be only partially attributed to AV technology (e.g. [17]).

While computer science educators and technology critics tend to view the 'markedly mixed' results of these studies as a reason not to adopt AV technology, we believe that, if

Table 1. The descriptive dimensions of the eight extant taxonomies of AV and software visualization

SV taxonomy	Descriptive Dimensions
Myers [54]	*Aspect* (code, data, algorithms) × *Form* (static, animated)
Shu [55]	*What is visualized* (data or information about data, program and/or execution, software design)
Brown [56]	*Content* (direct, synthetic) × *Persistence* (current, history) × *Transformation* (incremental, discrete)
Stasko and Patterson [57]	*Aspect* × *Abstractness* × *Animation* × *Automation*
Singh and Chignell [58]	*What is visualized* (program, algorithm, data) × *Form* (static, dynamic)
Kraemer and Stasko [59]	*Visualization task* (data collection, data analysis, storage, display) × *Visualization purpose* (debugging, performance evaluation or optimization, program visualization)
Roman and Cox [60]	*Scope* × *Abstraction* × *Specification method* × *Interface* × *Presentation*
Price *et al*. [18]	*Scope* × *Content* × *Form* × *Method* × *Interaction* × *Effectiveness*

we are willing to look below the surface of 'markedly mixed,' there are deeper lessons to be learned from trends in these studies. However, as we just pointed out, existing reviews of AV technology have neglected to systematically analyze this body of AV empirical studies for the purpose of gaining further insight into AV effectiveness.

In this article, we aim to fill this gap in the research by presenting a meta-study of past experimental studies of AV effectiveness. Specific research questions to be explored by our meta-study include the following

- What factors have the studies posited to influence educational effectiveness? Have certain factors had more impact than others? Are there any notable trends?
- How have studies defined and measured 'educational effectiveness?' Have those definitions and measurement techniques influenced the results?

In light of our answers to those questions, we are ultimately interested in addressing two broader questions:

- Is AV technology educationally effective?
- What are fruitful directions for future AV technology and effectiveness research?

We begin, in Section 2, by defining the boundaries of the meta-study—that is, by specifying what research falls within and outside of its scope. Section 3 briefly overviews the 24 experimental studies that form the central data of our meta-study. In Section 4, we outline the methods we employ to analyze our corpus of studies. Section 5 presents the main findings of our analysis. Finally, in Section 6, we synthesize those findings into a set of conclusions and an agenda for future research.

2. Scope

The focus of this meta-study is on algorithm visualization effectiveness. We adopt a standard definition of algorithm visualization: a subclass of *software visualization* [18] concerned

with illustrating computer algorithms in terms of their high-level operations, usually for the purpose of enhancing computer science students' understanding of the algorithms' procedural behavior. The notion of effectiveness, as it will be used here, concerns the union of humans and technology within the context of a scenario of use. Associated with such a scenario of use is

(a) a particular objective to be fulfilled (e.g. 'learn how the target algorithm works');
(b) a particular individual or group having that objective;
(c) a particular algorithm visualization artifact that will be enlisted; and
(d) a particular target algorithm to be visualized.

Within this context, effectiveness thus responds to the question: To what extent does the algorithm visualization artifact assist the individual or group in fulfilling the objective?

2.1. AV Scenarios of Use

Given that the notion of effectiveness derives its meaning from the peculiarities of a scenario of use, as just described, what scenarios of use might be the focus of studies of AV effectiveness? One of the original scenarios of AV use envisioned by its pioneers was algorithms research. For example, Brown [2] reports that his BALSA system was used to analyze a novel, stable version of Mergesort. However, since such scenarios have not been widely reported, and since we are unaware of effectiveness studies of such scenarios, we will not focus on them in this meta-study. Rather, our focus will be on scenarios of AV use within computer science education. Indeed, AV technology has seen by far its most use within computer science education, and educational scenarios of use have been by far the most widely studied.

Figure 1 presents a taxonomy of scenarios of use within computer science education. Each of these well-established scenarios distinguishes an educational use of AV technology:

- *Lectures*. In high school classrooms and college lectures, computer science instructors use graphic representations to help them explain aspects of the algorithms under study (see, e.g. [2, Appendix A, 19]). As Gurka and Citrin [20] put it, AV in lectures is essentially 'an extension of the blackboard, but with more capabilities available' (p. 183).

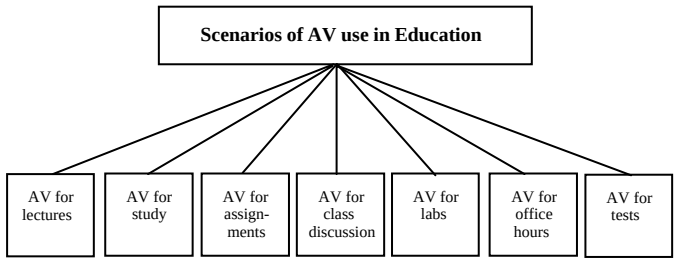

Figure 1. A taxonomy of scenarios of AV use in education

- *Assignments*. Students work on course assignments on their own time, and hand them in before established deadlines. Several computer science educators have described their use of assigments in which students construct their own visualizations of the algorithms under study [2, Appendix A; 4, 5]. Goldenson and Wang [21] describe course assignments for which students use the Pascal Genie programming environment, which has built-in design- and run-time SV tools.
- *Class discussion*. After completing assignments in which they construct their own visualizations, students might use AV technology to present their visualizations to their classmates and instructor for feedback and discussion [5]. In such scenarios, AV technology serves both to enable greater student participation in the class, and to mediate student–instructor interaction.
- *Laboratories*. In AV laboratories, students interactively explore algorithms and data structures through structured laboratory exercises [8]. Like assignments, laboratory sessions have a concrete goal, embodied in a deliverable assignment. However, unlike assignments, labs are constrained by both a location (a laboratory containing graphic workstations) and a contiguous block of time (a session or sitting).
- *Study*. Students enrolled in computer science courses have the opportunity to study for tests at any time. Depending on their individual preferences, students may elect to enlist AV in their study by drawing their own visualizations, by examining hard copies of visualizations constructed by others (professors or book authors), or by using interactive AV software to which they have access.
- *Office hours*. In college courses, professors and teaching assistants schedule weekly office hours, during which students in need of assistance may visit them. In this setting, instructors may use AV to help them diagnose bugs in students' programs [20], or to help them answer student questions. In the latter case, AV plays an explanatory role, akin to its role in lectures.
- *Tests*. In closed test-taking conditions, AV can be used to help pose questions. For example, Brown ([2, Appendix A]) reports that exams in the algorithms courses at Brown University often included stills of algorithm animations discussed in class; students would be asked to 'name-that-algorithm', just as students in an art history class might be asked to identify paintings. Alternatively, one can imagine a test question that has students to indicate the behavior of an algorithm by drawing a series of data structure snapshots.

2.2. AV research Techniques

The above scenarios of use provide a focal point for a body of research with a common interest in studying, evaluating and ultimately improving the effectiveness of AV technology. Figure 2 presents a taxonomy of the general research techniques that this body of research has employed:

- *Anecdotal* techniques aim to appeal to reader intuition by presenting instances and examples of AV system use, as recounted by the system's authors. Nearly every paper and article reporting on a novel AV system includes such an analysis.
- *Programmatic* techniques use the actual programs required to produce visualizations within a given AV system as a basis for assessment. For example, Cox and Roman [22] evaluate their Pavane system by summing the number of lines of code that are

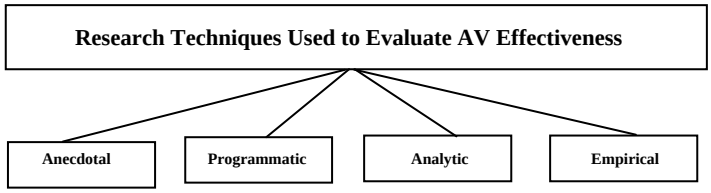

Figure 2. A taxonomy of methods for evaluating AV effectiveness

sufficient to specify a visualization within the system—the lower that number, the better.

- *Analytic* evaluation techniques (e.g. [23, 24]) aim to provide a principled assessment of an interactive system's effectiveness, while avoiding the overhead of extensive empirical data collection. Effectiveness, in the case of analytic evaluation, boils down to usability—the fewer usability problems identified, the more effective the system.
- *Empirical* evaluation, in contrast to the other techniques, involves the collection of actual data on humans involved in tasks with AV systems. An analysis process attempts to transform the data into a set of statements that respond to the research questions posed by the evaluation.

In this meta-study, we focus on effectiveness evaluations that employ *empirical* techniques. Our position is that empirical techniques are ultimately the most informative of the techniques, because they are rooted in observable, and often measurable, phenomena. Although we believe that studies that employ any of the numerous empirical techniques (see [25] for a review) have something potentially important to say about AV effectiveness, we restrict the analysis in this meta-study to those studies that use *controlled experimentation*. Controlled experiments aim to assert a causal relationship between factors (i.e. independent variables) and measures (i.e. dependent variables). While there are many variants on controlled experiments, all of the published AV experiments have been between-subjects experimental comparisons, in which two or more groups of participants use alternative means to learn about an algorithm. If statistically significant differences can be detected between the groups' performances, then experimenters may conclude that the factors significantly affect the measures.

We limit our scope to experimental evaluations for two reasons: first, because they constitute the largest, most mature subset of the empirical studies; and second, because the evolution of their design reveals an evolution in thinking about why and how AV technology is effective, implying that we can gain considerable insight from considering the legacy of these experiments as a whole.

3. Data

Our meta-study's data consist of 24 experimental studies that have considered AV effectiveness. To the best of our knowledge, this corpus comprises close to the entire

population, if not the entire population, of published AV effectiveness experiments. Table 2 provides a synopsis of these studies. For each experiment, the factors (independent variables) appear in column 2; the measures (dependent variables) appear in column 3; and a summary of the experiment's key results appears in column 4.

Twenty-two of the 24 of the experiments attempt to determine whether various factors affect learning within the 'study' scenario of use described in Section 2. In these experiments, learning is operationalized in terms of some sort of post-test, which participants take upon completing their study session. The two other experiments [26, 27] consider 'assignment' scenarios in which participants use AV to solve problems. In these experiments, problem-solving efficacy is operationalized in terms of whether the problems were solved, and how much time was needed.

Figure 3 summarizes the results of the 24 experiments. As the figure illustrates, the results have been widely mixed. Eleven of the 24 studies yielded a 'significant' result—that is, a statistically significant difference between the performance of (a) a group of students using some configuration of AV technology, and (b) another group of students using either an alternative configuration of AV technology, or no AV technology at all. For example, Lawrence [16, Chapters 6 and 9] found, in two separate experiments, that students who explored algorithm animations driven by self-constructed input data sets scored significantly higher than students who either watched the same animations driven by data supplied by the experimenter, or who had no access to such animations. Likewise, Crosby and Stelovsky [28] found that students who interacted with an algorithm animation performed significantly better than students who listened to a lecture, and that 'concrete' learners (as measured by the Myers–Briggs-Type Indicator) benefited significantly more from the algorithm animation than did 'abstract' learners. Finally, Hansen *et al*. [14] found that students who learned an algorithm using their HalVis hypermedia environment, which provides its viewers with multiple views and engages them in input data set design, interactive prediction, and queston-answering, significantly outperformed students who learned the same algorithm using (a) textual materials (Study I), (b) a lecture (Study II), or (c) a conventional, one-view algorithm animation environment with no facilities for interactive prediction or question-answering (Study V). In follow-up studies, Hansen *et al*. [14] additionally found that students who interacted with the HalVis conceptual and detailed views performed significantly better than students who did not have access to these views (Studies VII and VIII).

In contrast, a roughly equal number of studies (10) did not have a significant result. In other words, in these studies, no statistically significant differences could be found between the performance of (a) groups of students using some configuration of AV technology, and (b) groups of students using either an alternative configuration of AV technology, or no AV technology at all. For instance, Price [26] had two groups of students debug a 7500 program using a debugger with and without an animated view. He found no significant differences in debugging efficacy between the two groups. Similarly, Stasko *et al*. [15] had two groups of students learn the pairing heap data structure (a) by interacting with an algorithm animation, and (b) by studying textual materials. Although the animation group performed better on a post-test, the difference was not statistically significant. Finally, Lawrence [16, Chapters 4 and 5] compared the post-test performance of students who learned algorithms using animations with alternative representational characteristics: sticks vs. dots; 9 vs. 24 vs. 41 data elements; and labels vs. no labels. She found no significant differences among the groups.

Table 2. Summary of controlled experiments that consider AV effectiveness

Study	Indep. Var.	Depend. Var.	Key Results
Price [26]	*Debugging medium* (debugging with animated view vs. debugging without animated view)	Debugging time Whether bug found	No significant differences were found.
Stasko *et al*. [15]	*Learning medium* (text only vs. text-and-animation)	Post-test accuracy	Non-significant trend was found favoring the text-and-animation group (t=1.111, df=18, $p < 0.13$)
Lawrence [16, Chapter 4.4]	*Data set size* (9, 25, or 41 elements) *Data representation style* (horizontal/vertical sticks, dots)	Post-test accuracy	No significant differences were found
Lawrence [16, Chapter 5]	*Order of algorithm presentation* (Quick sort first vs. Selection sort first) *Data representation style* (labeled vs. unlabeled) Covariates: *Spatial and verbal abilities*	Post-test accuracy Time to take post-test	No significant differences were found Spatial and verbal abilities not correlated with performance
Lawrence [16, Chapter 6]	*Level of learner involvement* (study text/passively view animation vs. study text / actively view by constructing own input data sets)	Post-test accuracy Time to take post-test	Participants who viewed animations for which they constructed their own data sets scored significantly higher on post-test
Lawrence [16, Chapter 7]	*Representation color* (color vs. black-and-white) *Representation labeling* (algorithmic step labels vs. no labels)	Post-test accuracy Accuracy on a transfer task	Participants who viewed black-and-white animations scored significantly higher Participants who viewed labeled animations scored significantly higher
Lawrence [16, Chapter 8]	*Order of medium* (text-first vs. animation-first) *Order of algorithm presentation* (selection sort first vs. Kruskal MST first)	Post-test accuracy	No significant differences detected

Lawrence [16, Chapter 9]	*Learning medium*/*Level of learner involvement* (lecture-only vs. lecture + passively view animation vs. lecture + actively view animation by constructing own input data sets)	Free-response post-test accuracy Multiple choice/true-false post-test accuracy	On free-response post-test, participants who heard lecture and actively viewed animation significantly outperformed students who only heard lecture
Crosby and Stelovsky [28]	*Learning medium* (lecture vs. multimedia) *Cognitive style* (S/concrete vs. N/abstract) *Test question type* (text vs. graphics)	Pre- to post-test improvement	Participants who learned with multimedia significantly outperformed participants who learned through the lecture Significant interaction effect between cognitive style and learning medium: 'S' participants performed significantly better with multimedia
Byrne *et al*. [17, Study I]	*Learning medium* (animation vs. no animation) *Interactive prediction* (predict the next algorithm step vs. no prediction)	Post-test accuracy Prediction accuracy	On post-test's 'hard' questions, participants who viewed the animation and/or made predictions performed significantly better than participants who did neither.
Byrne, *et al*. [17, Study II]	Same as previous	Same as previous	Same as previous, except that difference was detected on post-test's 'procedural' questions,
Gurka [20]	*Learning medium* (animation vs. no animation) (Note: This experiment was an attempt to improve upon a portion of the Byrne *et al*. [17] experiment)	Post-test accuracy	No significant differences were found between the two groups Animation group appeared to be highly motivated by the animation
Kann *et al*. [33]	*Level of learner involvement* (program algorithm vs. program algo./construct animation vs. program algo./view anim. vs. program algo./view anim./construct anim.)	Programming accuracy Post-test accuracy	Participants who viewed animation scored significantly higher on post-test than participants who did not view animation

Table 2 (*Continued*)

Study	Indep. Var.	Depend. Var.	Key Results
Mulholland [27]	*Tracing medium* (three textual tracers, TPM)	Number of problems solved (5 min max per problem)	Participants who used the graphical tracer (TPM) solved significantly fewer problems than participants who used the textual tracers (Spy, PTP, TTT)
Hansen *et al*. [14, Study I]	*Learning medium* (HalVis hypermedia vs. text-only)	Pre- to post-test improvement (pseudocode reordering, algo. op. simulation, and prediction tasks)	Participants who learned with HalVis significantly outperformed participants who used text-only
Hansen *et al*. [14, Study II]	*Learning medium* (HalVis hypermedia vs. text-only)	Same as previous	Participants who learned with HalVis significantly outperformed participants who used text-only
Hansen *et al*. [14, Study III]	*Learning medium* (HalVis hypermedia vs. carefully selected text + problem-solving exercises)	Same as previous	No significant differences detected
Hansen *et al*. [14, Study IV]	*Learning medium* (HalVis hypermedia vs lecture)	Same as previous	Participants who learned with HalVis significantly outperformed participants who viewed lecture
Hansen *et al*. [14, Study V]	*Learning medium* (HalVis hypermedia vs. text + actively view animation by selecting own input data sets)	Same as previous	Participants who learned with HalVis significantly outperformed participants who read text and interacted with Xtango
Hansen *et al*. [14, Study VI]	*Combination of HalVis features* (full HalVis vs. HalVis without animation chunking vs. HalVis without pseudocode step highlighting vs. HalVis without interactive questions)	Same as previous	No significant differences detected

Hansen *et al*. [14, Study VII]	*Combination of HalVis views* (conceptual/detailed/populated vs. conceptual/detailed vs. conceptual/populated vs. detailed/populated)	Same as previous	Participants who interacted with conceptual view significantly outperformed participants who did not interact with the conceptual view
Hansen *et al*. [14, Study VIII]	*Combination of HalVis views* (conceptual/detailed/populated vs. conceptual only vs. detailed only vs. populated only)	Same as previous	Participants who had access to (a) all three HalVis views, or (b) the Detailed View, significantly outperformed the conceptual view and populated view groups Conceptual view group significantly outperformed the populated view group
Hundhausen and Douglas [31]	*Level of Learner Involvement* (self-construct visualizations vs. actively view predefined visualizations)	Accuracy and time on tracing and programming tasks	No significant differences were detected
Jarc *et al*. [32]	*Interactive prediction* (use animation software that enbles prediction of next algorithm step vs. use animation software with no prediction)	Post-test at end of three weekly lab sessions Learning time	No significant differences were detected on post-test The prediction group spent significantly more time using the animation software than the no-prediction group

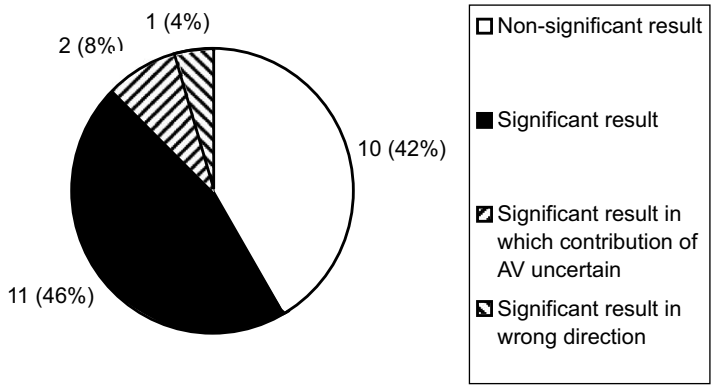

Figure 3. Summary of the results of the 24 experiments in our corpus

Two studies found a significant result in which the positive impact of visualization could not be disentangled from another factor. In these studies, Byrne *et al.* [17] found that students who made predictions regarding future animation frames while viewing algorithm animations performed significantly better than students who did not view animations or make predictions; however, the individual effects of prediction and animation could not be disentangled statistically.

Finally, one study yielded a 'negative' result in which students who used AV technology actually performed significantly worse than students who used text-based tools. In particular, Mulholland [27] found that, in Prolog tracing tasks, participants who used any of three textual tracers solved significantly more problems than participants who used TPM, a graphical tracer.

4. Method

In the field of psychology, meta-analytical techniques (see, e.g. [29]) have been developed for statistically analyzing a body of related experimental studies with similar independent and dependent variables. The goal of these meta-analytical techniques is to infer invariant cause–effect relationships based on the findings of a corpus of related studies. To do so, a statistical meta-analysis systematically combines and compares the studies in order to determine the effect size and significance level of their independent variables.

For example, in order to build a 'task-feature taxonomy' that can guide the effective use of information visualization technology, Chen and Yu [30] perform a statistical meta-analysis of a sample of 35 experimental studies of information visualization. However, the diversity of the studies in their sample makes it difficult to apply meta-analytic techniques; they are ultimately forced to reduce their sample to six sufficiently homogeneous studies focusing on information retrieval tasks. Interestingly, their analysis determines that individual differences have a larger and more consistent effect on human performance than does information visualization technology itself.

We considered using statistical meta-analytic techniques in this meta-study, but ultimately decided against them for two reasons. First, like the sample of Chen and Yu, the corpus of experimental studies on which we are focusing is quite diverse. Indeed, while

their independent and dependent variables may appear similar on the surface, the ways in which those variables are manipulated and operationalized vary considerably. For example, while some studies compare the use of AV technology against the use of conventional learning materials (e.g. [15, 17, 28]), others compare competing versions of AV technology (e.g. [16, 31, 32]). Likewise, while some studies operationalize learning in terms of a post-test consisting of exam-style questions (e.g. [15–17]), others opt to measure performance in terms of accuracy on programming [31, 33], tracing [31], and prediction [14, 17] tasks. Second, even if, like Chen and Yu, we were able to find a small number of sufficiently homogenous studies on which to perform a statistical meta-analysis, we do not believe that such a meta-analysis would be informative. This is not only because the sample that we would ultimately consider would be small, but also because large discrepancies in the results of the studies in our sample would give rise to statistically uninformative conclusions.

For these reasons, we believe that any meta-analysis of this corpus of studies must find a principled way to classify the studies into smaller groups, such that each group's results, when taken as a whole, is more uniform. What might be the basis of such a principled classification? In scrutinizing the studies in our corpus, we have observed (see also [5]) notable differences in their choices of both independent variables (i.e. the factors they posit to cause effectiveness) and their choices of dependent variables (i.e. the ways in which they measure effectiveness). For example, some studies manipulate representational features of visualizations (e.g. color, shading, geometry), while others manipulate the learner's level of activity (e.g. learner views visualization; learner designs input data; learner makes predictions regarding future visualization states). Likewise, the studies measure effectiveness in terms of differing kinds of knowledge acquisition—both conceptual and procedural.

The analysis method we employ in this meta-study, then, involves first classifying the studies in various principled ways based on their independent and dependent variables, and then scrutinizing the results of the studies *vis-a-vis* these classifications. Our analysis of study results includes both quantitative comparisons of the numbers of statistically significant and non-significant results in opposing classes of studies, and qualitative assessments of trends.

It is important to note that our analysis will *not* make judgments about the validity of each experiment—that is, the extent to which each experiment's results can be trusted. Rather, we will make the simplifying assumption that the results of each experiment in our corpus can be trusted roughly equally. While the experimental results on which we base our analyses are clearly tempered by the soundness of their experimental designs, and while there clearly are differences in the soundness of the experiment designs in our corpus, we believe that the differences are minor. Hence, critiquing experimental design would, we believe, constitute an unnecessary distraction from our focus on results.

5. Analysis

5.1. Independent Variables

We begin our analysis by scrutinizing the studies' independent variables—that is, the factors they have posited to cause effectiveness. We argue that the studies' differences in

independent variables actually reflect deeper differences in underlying theories of effectiveness: the studies' assumptions about how and why AV might be effective. In the studies in our corpus, AV technology is enlisted as a pedagogical tool; its goal is to enhance learning. It follows that the theories of effectiveness underlying the studies in our corpus correspond with alternative theories of learning.

This section begins by presenting the four alternative learning theories adopted by the studies in our corpus. Next, we link each study to a particular theory based on its choice of independent variables. Following that, we quantitatively assess the robustness of each theory by considering the number of significant vs. non-significant results of the studies associated with it. Finally, we perform a reality check on our findings by assessing the extent to which the empirical results of competing theories actually lend support to the theory that our analysis found to have the greatest support. This finer-grained analysis enables us to draw more definitive conclusions about the robustness of that theory.

5.1.1. Four Theories of Effectiveness

The varied designs of the AV experimental studies reflect a variety of underlying theories of effectiveness. While finer distinctions may certainly be made among these studies' theoretical underpinnings, the broad-brush analysis presented here places them into four broad theoretical camps, which we label Epistemic Fidelity, Dual-coding, Individual Differences, and Cognitive Constructivism. Below, we briefly describe these theories; for a fuller treatment, see [5].

Epistemic Fidelity. Epistemic Fidelity theory [5, 34, 35] has its roots in a representationalist epistemological framework (see, e.g. [36]), which assumes that humans carry around in their heads symbolic models of the physical world, and that such symbolic models are the basis for all of their reasoning and action. The key assumption of Epistemic Fidelity theory, then, is that graphics have an excellent ability to encode an expert's mental model of an algorithm, leading to the robust, efficient transfer of that mental model to the viewer (see Figure 4). Thus, Epistemic Fidelity theory emphasizes the value of a good denotational match between the graphical representation and the expert's mental model. The higher the 'fidelity' of the match, the more robust and efficient is the transfer of that mental model to the viewer of the visualization, who non-problematically decodes and internalizes the target knowledge.

Dual-coding. Based on Mayer and Anderson's [37] integrated dual-code hypothesis, Dual-coding theory proceeds from Paivio's [38] assumption that 'cognition consists largely of the activity of two partly interconnected but functionally independent and distinct symbolic systems' (p. 308). One encodes verbal events (words); the other encodes non-verbal events (pictures). According to Mayer and Anderson's hypothesis, visualizations that encode knowledge in both verbal and non-verbal modes allow viewers to build dual *representations* in the brain, and referential connections between those representations. As a consequence, such visualizations facilitate the transfer of target knowledge more efficiently and robustly than do visualizations that do not employ dual-encoding.

Individual Differences. A legacy of psychological experiments have attempted to operationalize, and to better understand, individual differences in human cognitive abilities and learning styles (see, e.g. [39]). The key contribution of this research has been not only a battery of instruments for rating and classifying individuals along several dimensions, but also empirical results that make important statements about human performance

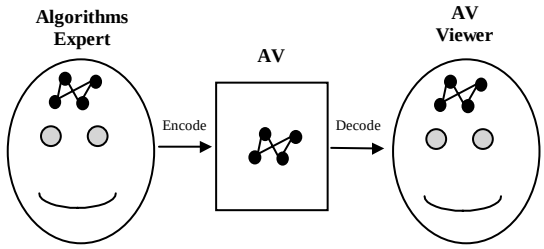

Figure 4. A schematic of *epistemic fidelity* theory's view of knowledge transfer

relative to individual differences so measured. Thus, in the context of this analysis, Individual Differences theory asserts that measurable differences in human abilities and styles will lead to measurable performance differences in scenarios of AV use. For example, within the scope of Epistemic Fidelity theory's knowledge transfer model (see Figure 4), Individual Differences with respect to learning style (see, e.g. [40]) might enable some individuals to decode visualizations more efficiently and robustly than other individuals.

Cognitive Constructivism. Rather than regarding knowledge as representations of an objective reality that people carry around in their heads, Cognitive Constructivism (see, e.g. [41]) asserts that there is no absolute knowledge. Instead, it holds that individuals construct their own individual knowledge out of their subjective experiences in the world. By becoming actively engaged with their environment, individuals actively construct new understandings by interpreting new experiences within the context of what they already know.

Note that Cognitive Constructivism's emphasis on active learning has important implications for the effective use of AV technology. In particular, it suggests that individuals do not stand to benefit from the technology by merely passively viewing visualizations, no matter how high the level of their epistemic fidelity. Instead, AV technology users must become more actively engaged with the technology in order to benefit most from it. The technology, on this view, is seen not as a conveyer of knowledge, but as a tool for knowledge construction.

In order to highlight the key distinctions among the four theories just discussed, Table 3 synopsizes the theories and the role they ascribe to an AV.

5.1.2. Linking Studies to Theories

The independent variables chosen by each of the 24 experimental studies summarized earlier, as well as the ways in which those independent variables are manipulated, provide insight into each study's theoretical underpinnings. By examining each study's experimental manipulation, we can link it to one of the theories of effectiveness just discussed.

Table 4 links each of the 24 experimental studies to an underlying theory of effectiveness. In four cases [16, Chapter 5 and 7, 28, 14, Study VI], a study is linked to two underlying theories, because the study defined independent variables, or performed experimental manipulations, that were judged to support multiple underlying theories.

As this table indicates, we judged ten studies to be aligned with Epistemic Fidelity theory. These studies manipulated either (a) representational features of visualizations, or (b) the order in which visualizations are presented. The hypothesis that these studies

Table 3. Juxtaposition of the four theories of effectiveness

Theory	Synopsis	Role of AV
Epistemic Fidelity	Graphics have an excellent ability to encode an expert's mental model, leading to the robust, efficient transfer of that mental model to viewers	Encoding of knowledge
Dual-coding	Representations with dually coded information (e.g. graphics and text) promote the most robust and efficient knowledge transfer	Encoding of knowledge
Individual Differences	Differences in human cognitive abilities and learning styles enable some to benefit from SV more than others	Unspecified, but 'encoding of knowledge' fits in well with the theory
Cognitive Constructivism	Active engagement with SV enables one to construct one's own understanding	Resource for knowledge construction

are positing is that certain representational features, or certain orderings of features, will promote the knowledge decoding process more robustly and efficiently than others.

In two separate studies, Lawrence [16, Chapters 5 and 7] compared the efficacy of representations in which information is singly (graphics only) and doubly encoded (graphics and textual labels). These studies are clearly inspired by Dual-coding theory, which holds that the dually coded representations will promote more robust and efficient knowledge transfer.

Lawrence [16] and Crosby and Stelovsky [28] considered the impact of spatial abilities, cognitive abilities, and learning styles on one's ability to learn from a visualization. These studies were clearly influenced by Individual Differences theory.

Finally, 14 studies manipulated the way in which humans are engaged with their environment to accomplish some task for which AV is assumed to be beneficial. By varying primarily the kinds of activities and forms of engagement, and not the representation, these studies demonstrate a loyalty to Cognitive Constructivism, which views the benefits of AV technology not in its ability to transfer knowledge, but in its ability to promote the construction of knowledge through active engagement.

5.1.3. Coarse-grained Analysis of Theory Robustness

Given that each study shows a loyalty to one of the theories of effectiveness, an important question arises: What level of experimental support has been demonstrated for each of the four theories? To start to answer that question, Figure 5 presents a bar graph of the experimental results *vis-a-vis* each of the four theories. In this bar graph, each experimental study is assumed to have equal weight. The length of each bar corresponds to the number of studies that have been guided by the corresponding theory. The filled-in portion of each bar indicates the number of statistically significant results that were obtained in support of the theory[a]. The amount each bar is filled in thus graphically indicates the proportion of the results that were statistically significant.

[a] Note that the one study that obtained *negative* significant result [27] is counted as a non-significant result in the Epistemic Fidelity bar.

Table 4. The experimental studies *vis-a-vis* the effectiveness theories they were designed to support

Epistemic Fidelity	Dual Coding	Individual Differences	Cognitive Constructivism
Price [26]	Lawrence [16, Chapter 5]	Lawrence [16, Chapter 5]	Stasko *et al*. [15]
Lawrence [16, Chapter 4.4]	Lawrence [16,Chapter 7]	Crosby and Stelovsky [28]	Lawrence [16, Chapter 6
Lawrence [16, Chapter 7]			Lawrence [16, Chapter 9]
Lawrence [16, Chapter 8]			Crosby and Stelovsky [28]
Gurka [20]			Byrne *et al*. [17, Study I]
Mulholland [27]			Byrne *et al*. [17, Study II]
Hansen *et al*. [14, Study III]			Kann *et al*. [33]
Hansen *et al* [14, Study VI]			Hansen *et al*. [14, Study I]
Hansen *et al*. [14 Study VII]			Hansen *et al*. [14, Study II]
Hansen *et al* [14, Study VIII]			Hansen *et al*. [14, Study IV]
			Hansen *et al*. [14, Study V]
			Hansen *et al*. [14, Study VI]
			Hundhausen and Douglas [31]
			Jarc *et al*. [32]

As Figure 5 illustrates, Cognitive Constructivism has been the most tested of the four theories (14 studies), with Epistemic Fidelity theory trailing close behind (10 studies), and Dual-coding theory and Individual Differences theory lagging far behind. Moreover, notice that studies in support of Cognitive Constructivism have obtained the greatest number (10), and highest percentage (71%), of statistically significant differences. Based on this broad-brush analysis, one can conclude that Cognitive Constructivism has garnered the most consistent empirical support.

5.1.4. Finer-grained Analysis of Cognitive Constructivism

Clearly, the conclusion reached by the above analysis must be considered tentative, because the analysis fails to consider the global case for a given theory—that is, whether study results outside of a given theoretical camp lend support to, or weaken, a given theory. Focusing on the case for Cognitive Constructivism, the analysis that follows scrutinizes its support across the entire corpus of studies. As we shall see, this analysis lends further credence to the theory by illustrating its broader predictive success.

An important assumption of Cognitive Constructivist theory is that learner activity matters; active learning is assumed to be superior to passive learning. Rather than passively viewing algorithm visualizations, active learners augment their viewing with such activities as

- constructing their own input data sets (e.g. [16, Chapter 9]);
- making predictions regarding future visualization states (e.g. [17]);
- programming the target algorithm (e.g. [33]);

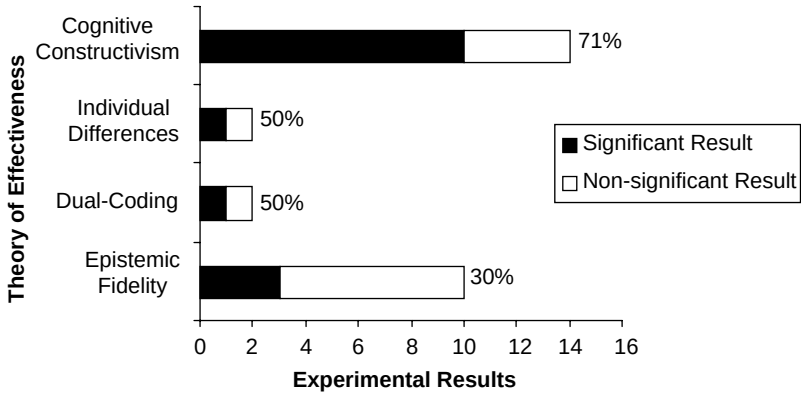

Figure 5. Experimental results *vis-a-vis* theory of effectiveness. Total number of experiments (height of hollow bar), number of experiments yielding significant results (height of filled-in bar), and percentage of experiments yielding significant results (proportion of hollow bar that is filled-in) in each of four general categories of experiments, classified by their underlying theory of effectiveness

- answering strategic questions about the visualization (e.g. [14]); and
- constructing their own visualizations (e.g. [31]).

On the Cognitive Constructivist view, then, AV technology is seen as educationally effective to the extent that it actively engages learners in such activities.

Given this, an obvious question arises: Does the precise form of the activity matter? In other words, are some activities better at promoting 'active learning' than others? The Cognitive Constructivist position is that, as long as the learner's activity is salient to the target knowledge or skill, the effort required to engage in the activity is more important than its actual form. Thus, assuming equal salience, Cognitive Constructivist theory would predict that the more effort required to engage in the activity, the more robust the learning.

Applying this reasoning to controlled experimental comparisons, we find that Cognitive Constructivism predicts an educational advantage only in cases in which the activity of one treatment group is more effortful than that of the other treatment group(s). Conversely, if all treatment groups' activity is roughly equivalent in terms of necessary effort, then the theory does not predict any learning differences.

To further scrutinize the robustness of Cognitive Constructivism, we can examine the results of the 24 experiments in our corpus *vis-a-vis* the equivalence of the effort required to perform the activities in their competing treatment groups. As a first step, we must judge the equivalence of the effort required to perform the activities in competing treatment groups (see Table 5). If, in a given experiment, the competing learning activities require roughly equivalent cognitive effort, we classify them as 'Effort equivalent.' For example, we judge that viewing two alternative visualizations [16, Chapter 4.4] requires roughly equivalent effort. Likewise, we estimate that paper-and-pencil problem-solving exercises require effort roughly equivalent to that of interacting with the HalVis hypermedia system [14, Study III], which engages users in equivalent problem-solving

Table 5. Classification of experiments based on the equivalence of the effort required to perform the activities in competing treatment groups

Effort Equivalent	Effort not Equivalent
Price [26]	Stasko *et al.* [15]
Lawrence [16, Chapter 4.4]	Lawrence [16, Chapter 6]
Lawrence [16, Chapter 5]	Lawrence [16, Chapter 9]
Lawrence [16, Chapter 7]	Crosby and Stelovsky [28,]
Lawrence [16, Chapter 8]	Byrne *et al.* [17, Study I]
Gurka [20]	Byrne *et al.* [17, Study II]
Mulholland [27]	Kann *et al.* [33]
Hansen *et al.* [14, Study III]	Hansen *et al.* [14, Study I]
Hansen *et al.* [14, Study VII]	Hansen *et al.* [14, Study II]
Hansen *et al* [14, Study VIII]	Hansen *et al.* [14, Study IV]
	Hansen *et al.* [14, Study V]
	Hansen *et al.* [14, Study VII]
	Hundhausen and Douglas [31]
	Jarc *et al.* [32]

activities. If, on the other hand, a notable difference exists in the effort levels required to engage in the competing treatment groups of a given experiment, we classify the experiment as 'Effort not equivalent.' For example, in our judgment, reading textual materials, listening to a lecture, and passively viewing a visualization all require markedly less effort than actively using a visualization in conjunction with some sort of learning exercise—for example, prediction [17, 32], input data selection [16, Chapter 9], or programming [33].

An important implication of Cognitive Constructivism is that experiments rated as 'Effort not equivalent' will show a significant learning advantage for the more effortful group . Conversely, the theory suggests that experiments rated as 'Effort equivalent' will show no significant differences between the treatment groups[b]. The next step in the analysis is to see whether these predictions are actually borne out. Under the assumption that each experimental study has equal weight, Figure 6 carries out this step by plotting the number of results and significant results *vis-a-vis* the 'Effort not equivalent/Effort equivalent' classification presented in Table 5. In six of the nine studies in which competing treatment group activities required roughly equivalent effort, no significant results were found. In contrast, in ten of the 14 studies in which one of the treatment groups required more effort, the group expending the higher level of effort significantly outperformed the other group. These results accord reasonably well with the theory's predictions. In the case of 'Effort equivalent' studies, Cognitive Constructivism predicts 67% of the results. In the case of 'Effort not equivalent' studies, Cognitive Constructivism predicts 71% of the results.

[b] Of course, these predictions assume that an experiment's learning materials are well designed and contain enough information to enable participants to perform well on the post-test. Indeed, if the experimental materials are deficient—for example, if the algorithm animation is poorly designed or lacks necessary information—then no amount of effort is likely to lead to a measurable learning difference.

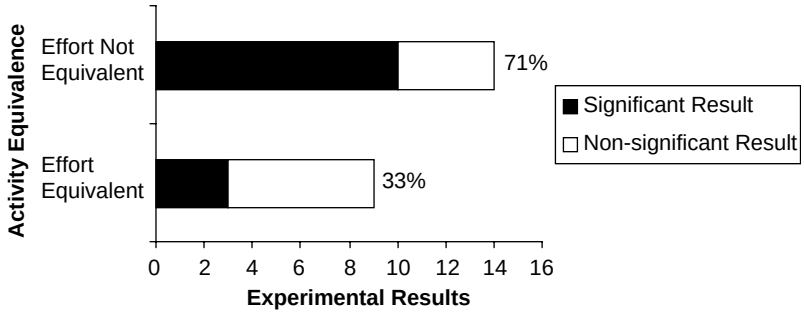

Figure 6. Results of experimental studies *vis-a-vis* activity effort classification of Table 5

This quantitative finding is further reinforced by contrasting two studies classified as 'Effort equivalent' with similarly designed studies classified as 'Effort not equivalent'. Gurka [20] attempted to replicate one of the experiments of Byrne *et al.* [17], which obtained a significant result in favor of participants who made predictions and viewed an animation. However, Gurka eliminated the prediction conditions from her experiment, opting instead to compare a group of participants who used an animation against a group of participants who did not, with no substantial differences in the groups' levels of engagement. Both groups of participants, for example, engaged in group study as part of their learning session. As would be predicted by Cognitive Constructivist theory, Gurka's de facto equalization of participants' activities led to a non-significant result.

Likewise, in a series of studies, Hansen *et al.* [14] found that participants who used their HalVis multimedia environment, which actively engages its users by having them construct their own input data sets, answer questions and make predictions, significantly outperformed participants who learned from textual materials (Studies I and II), and viewed a lecture (Study IV). In one of their studies (Study III), however, the researchers attempted to equalize the activities of the HalVis and text-only groups by having students in the text-only group not only read articles on an algorithm, but also complete a series of paper-and-pencil exercises. As would be predicted by Cognitive Constructivist theory, the researchers' decision to engage the text-only participants as actively as participants in the HalVis group led to their failure to find a significant difference between the two groups' learning outcomes.

In sum, our finer-grained analysis shows that Cognitive Constructivism predicts not only the greatest percentage of significant results (77%), but also a majority (60%) of the non-significant results. This analysis lends further credence to our coarse-grained conclusion that Cognitive Constructivist theory is the most robust. In contrast, Epistemic Fidelity theory has had the least predictive success, having predicted just 30% of the significant results, with another significant result running completely counter to the theory's predictions. Finally, due to low numbers of supporting studies, it is fair to say that the jury is still out on Dual-coding and Individual Differences theories.

5.2. Dependent Variables

We now move to an analysis of the studies' dependent variables—that is, the ways in which they have measured effectiveness. The two studies in our corpus that examined

'assignment' scenarios measured effectiveness in terms of participants' success at solving debugging and tracing problems with the help of AV technology [26, 27]. Neither of these experiments yielded a significant result. A survey of the remaining 22 studies' dependent variables suggests that they measured effectiveness in remarkably similar ways. Indeed, all of these experiments elected to measure effectiveness in terms of knowledge acquisition. The underlying hypothesis was that the experimental manipulation will lead some experimental treatments to acquire target knowledge more robustly than others.

On closer inspection of the 22 studies that measure effectiveness in terms of knowledge acquisition, we find notable differences both in terms of *what* they measure, and *how* they measure it. Below, we further investigate these two key differences, in an attempt to determine whether certain choices of dependent variables have led to greater success in detecting learning differences.

5.2.1. Differences in What Knowledge is Measured

The first notable difference among the studies' dependent variables lies in the precise types of knowledge they attempt to measure. Consistent with typical college exams in computer science, the studies have aimed to measure two distinct types of knowledge:

- *conceptual* or *declarative*—an understanding of the abstract properties of an algorithm, e.g. its Big-O efficiency, its range of output, or limits on the input data that it can process. A sample test question might be 'What is the worst-case efficiency of the algorithm?'
- *procedural*—an understanding of the procedural, step-by-step behavior of an algorithm, that is, how it operates on a set of input data. A sample test question might involve tracing an algorithm's key variables and data structures for a given set of input data.

It is important to note that these two forms of knowledge are not necessarily distinct. Frequently, a high level of conceptual knowledge is needed to be able to understand an algorithm's procedural behavior. For example, in order to understand how the Quicksort algorithm works, one needs a conceptual understanding of recursion. Conversely, understanding procedural operations can help with conceptual questions. For example, grasping the pattern of Quicksort's divide-and-conquer strategy can give one insight into the algorithm's efficiency.

Table 6 classifies the 22 knowledge-measuring studies according to whether their evaluation instruments test (a) *conceptual and procedural knowledge*, (b) *conceptual knowledge only*, or (c) *procedural knowledge only*. (Note that some studies include multiple dependent measures that test more than one of these combinations. These studies are listed multiple times—once for each respective combination.) Since most visualizations focus on the procedural behavior of an algorithm, it is no surprise that a majority of the studies have attempted to measure procedural knowledge alone. For the same reason, only a small number (3) of the studies have measured conceptual knowledge alone. Nine studies have forged a middle ground by measuring both forms of knowledge.

Under the assumption that each experimental study has equal weight, Figure 7 carries the analysis further by graphically presenting the number of significant and non-significant results *vis-a-vis* the three classes of experiments categorized in Table 6. As the figure illustrates, all three knowledge measures garnered comparable levels of experimental

Table 6. Classification of experimental studies based on the type of knowledge required to answer the questions on their evaluation instruments

Conceptual and procedural	Conceptual only	Procedural only
Stasko *et al.* [15]	Lawrence [16, Chapter 7]	Lawrence [16, Chapter 7]
Lawrence [16, Chapter 4.4]	Lawrence [16, Chapter 9]	Lawrence [16, Chapter 8]
Lawrence [16, Chapter 5]	Byrne *et al.* [17, Study II]	Crosby & Stelovsky [28]
Lawrence [16, Chapter 6]		Kann *et al.* [33]
Lawrence [16, Chapter 9]		Byrne *et al.* [17, Study I]
Gurka [20]		Byrne *et al.* [17, Study II]
Kann *et al.* [33]		Hundhausen and Douglas [31]
Jarc *et al.* [32]		Hansen *et al.* [14, Study I]
		Hansen *et al.* [14, Study II]
		Hansen *et al.* [14, Study III]
		Hansen *et al.* [14, Study IV]
		Hansen *et al.* [14, Study V]
		Hansen *et al.* [14, Study VI]
		Hansen *et al.* [14, Study VII]
		Hansen *et al.* [14, Study VIII]

Note: Some studies are listed more than once, because they employed multiple measurement instruments.

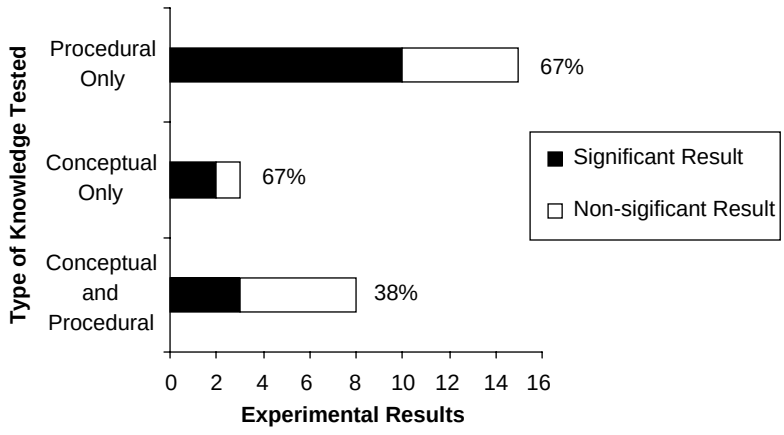

Figure 7. Experimental results *vis-a-vis* type of knowledge tested

support; no one measure appears to be more sensitive to learning effects than any other. Given that algorithm visualizations illustrate the procedural behavior of algorithms, it is perhaps somewhat surprising that two out of the three studies that measured conceptual knowledge exclusively obtained significant results. However, the sample size (3) is clearly too small for general conclusions to be drawn.

Of the other two knowledge measurements, *procedural knowledge only* appears to have been more sensitive to learning differences than *conceptual and procedural knowledge*, although this

result is difficult to interpret. One might speculate that the conceptual portion of the conceptual-and-procedural tests is what diminished their sensitivity. If that were the case, we would expect the conceptual-only tests to be even less sensitive than procedural-and-conceptual tests. That is not the case, however. Indeed, we find that combining the conceptual-only results with the conceptual-and-procedural results actually *raises* the overall sensitivity of conceptual-and-procedural tests to 45% (five of the 11 significant results). In spite of these inconsistent results, procedural test questions still appear to be somewhat more sensitive to AV technology's educational benefits than conceptual questions.

5.2.2. Differences in How Knowledge Acquisition is Measured

The methodology employed to measure knowledge acquisition constitutes the second notable difference in the studies' dependent variables. In thirteen of the 22 studies (see Table 7), a post-test designed to measure knowledge acquisition formed the sole basis of measurement. Post-test scores constituted the data that are statistically analyzed. In contrast, the other nine studies (see Table 7) gave participants both a pre-test and a post-test. In these studies, pre-test to post-test improvement formed the basis of the statistical analyses.

As Figure 8 illustrates, the results of experiments employing these two alternative measurement techniques differ noticeably. Seven of 13 (54%) of the studies that measured learning using a single post-test found a statistically significant difference between treatment groups. In contrast, seven of the nine (78%) studies that measured learning based on pre- to post-test improvement found statistically significant differences. In interpreting the 78% success rate of pre- to post-test improvement, one should keep in mind that it is based almost entirely on the work of a single line of studies. Nonetheless, given the difference in success rates, one can cautiously conclude pre- to post-test improvement may be more sensitive to learning differences than simple post-test performance.

Table 7. Classification of experimental studies based on their method of measuring learning

Post-test only	Pre- to post-test improvement
Stasko *et al.* [15]	Crosby and Stelovsky [28]
Lawrence [16, Chapter 4.4]	Lawrence [16, Chapter 5]
Lawrence [16, Chapter 5]	Hansen *et al.* [14, Study II]
Lawrence [16, Chapter 6]	Hansen *et al.* [14, Study III]
Lawrence [16, Chapter 7]	Hansen *et al.* [14, Study IV]
Lawrence [16, Chapter 8]	Hansen *et al.* [14, Study V]
Lawrence [16, Chapter 9]	Hansen *et al.* [14, Study VI]
Gurka [20]	Hansen *et al.* [14, Study VII]
Kann *et al.* [33]	Hansen *et al.* [14, Study VIII]
Byrne *et al.* [17, Study I]	
Byrne *et al.* [17, Study II]	
Hundhausen and Douglas [31]	
Jarc *et al.* [32]	

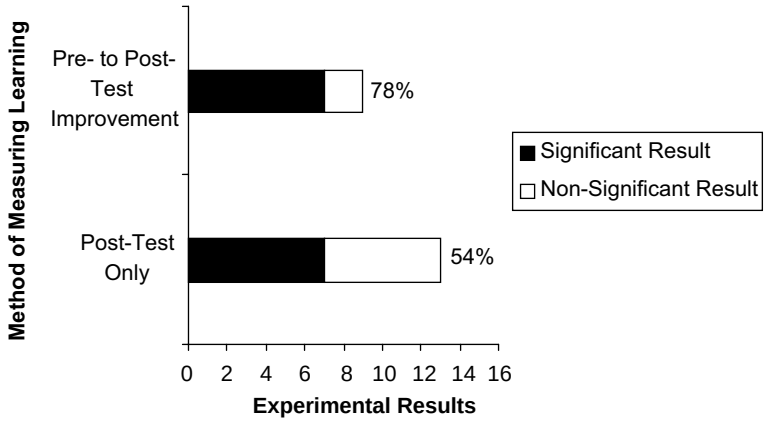

Figure 8. Experimental results *vis-a-vis* two alternative methods for evaluating learning

Viewing these results in light of Cognitive Constructivism's predictions lends further credence to this finding. Just two experimental studies [14, Studies III and VI] that measured pre- to post-test improvement actually failed to obtain a significant difference. Of those two studies, Cognitive Constructivism would have predicted only one of the studies to find a significant difference between treatment groups. Thus, according to Cognitive Constructivism, the pre- to post-test improvement measure failed just once. In contrast, the post-test only measure failed three times. Indeed, of the six 'post-test only' studies that failed to find a significant difference, Cognitive Constructivism would have predicted three [15, 31, 32] to have found a significant difference.

6. Conclusions

In this article, we have presented an integrative review of empirical studies of algorithm visualization effectiveness, in order to uncover trends in the research that might help us better understand how and why AV technology is effective. The focus of our review has been on one particular class of empirical studies: controlled experiments of AV technology in educational scenarios of use. We have pursued two separate analyses of 24 such experimental studies. Our first analysis examined their results *vis-a-vis* their independent variables in an attempt to better understand what factors have most consistently caused effectiveness. Conversely, our second analysis looked at the studies' results *vis-a-vis* their dependent variables, in order to understand both how effectiveness has been defined, and what measures are best able to detect it. What exactly have we learned from our meta-study's review and analysis? We conclude by revisiting our four original research questions in light of our results.

6.1. What Factors have had the Most Impact on Effectiveness?

Four general theories of effectiveness, encompassing differing epistemological assumptions about knowledge and its relationship to an AV, have been embraced by our sample of experimental studies: Epistemic Fidelity, Dual-coding, Individual Differences, and

Cognitive Constructivism. By far the greatest number of studies have been guided by either Epistemic Fidelity or Cognitive Constructivism. The other two theories, Dual-coding and Individual Differences, have not been explored thoroughly enough for definitive conclusions to be drawn. As our coarse-grained analysis indicates, experimental manipulations of AV learner activities, indicative of adherence to Cognitive Constructivist theory, have had more significant and consistent impact on experiments' dependent variables than have experimental manipulations of AV representation, indicative of adherence to Epistemic Fidelity theory. Moreover, an analysis of participant groups' engagement effort equivalence with respect to experimental results across all experiments further substantiates the pedagogical impact of using AV technology to engage students actively in their own learning. Thus, according to our analysis, *how* students use AV technology, rather than *what* students see, appears to have the greatest impact on educational effectiveness.

6.2. What Measures have been Most Sensitive to Effectiveness?

Twenty-two of the 24 studies in our corpus operationalize effectiveness in terms of conceptual knowledge, procedural knowledge, or both. An analysis of experimental results based on type of knowledge measured suggests that procedural knowledge may serve as a more sensitive measure of AV technology's benefits than conceptual knowledge. We take comfort in the fact that this finding is borne out in other non-experimental studies of AV technology not included in our corpus (e.g. [42]).

Closer inspection of the studies' dependent variables suggests that they have measured knowledge using two different techniques: post-test performance only, and pre- to post-test improvement Assuming that a learning difference truly exists between treatment groups, our analysis suggests that pre- to post-test improvement will be more likely to find such a difference than post-test performance alone. It is important to qualify this finding in two ways. First, it is biased in the sense that it is based largely on the results of one particular line of studies [14]. Second, we believe that measuring pre- to post-test improvement introduces a methodological problem that has not been addressed by the studies that employed it: The pre-test may give participants knowledge of the target algorithm outside of the treatment condition. Thus, we believe that studies that use this method to measure effectiveness need to address this methodological issue.

Finally, we would be remiss not to point out that our conclusions are limited in that they are based on a corpus of studies that considered an extremely limited range of dependent variables. A few studies employed alternative measures—for example, time spent learning [32], the time spent taking the post-test [31] and success on graphical vs. textual questions [28]. By and large, however, the studies in our corpus operationalized learning in terms of conceptual and procedural knowledge acquisition.

6.3. Is av Technology Effective?

Clearly, answers to the previous two questions provide a backdrop for addressing the all-important question of whether AV technology is educationally effective. In light of our meta-study findings, our answer to this question would have to be a qualified 'yes'. In what ways is AV technology educationally effective, and in what ways is it not educationally effective?

Let us begin with the ways in which it is *not* effective. With few exceptions, we found that studies in which students merely viewed visualizations did not demonstrate significant learning advantages over students who used conventional learning materials. Perhaps contrary to conventional wisdom, this observation suggests that the mere presence of AV technology—however well-designed and informative the visual representations it presents may appear to be—does not guarantee that students will learn an algorithm. To state this in terms of the learning theory (Epistemic Fidelity) underlying this form of AV technology use, our findings suggest that algorithm visualizations do not merely transfer an expert mental model of an algorithm to a student's brain. In successful applications of AV technology, something else appears to be going on.

In particular, our meta-study suggests that the most successful educational uses of AV technology are those in which the technology is used as a vehicle for actively engaging students in the process of learning algorithms. For example, as we have seen, AV technology has been successfully used to actively engage students in such activities as

- what-if analyses of algorithmic behavior (e.g. [16, Chapter 9]);
- prediction exercises (e.g. [17]) and
- programming exercises (e.g. [33]).

Notice that, in such cases, rather than being an instrument for the transfer of knowledge, AV technology serves as catalyst for learning. To state this in terms of the learning theory (Cognitive Constructivism) underlying this form of AV technology use, our results suggest that algorithm visualizations are educationally effective insofar as they enable students to construct their own understandings of algorithms through a process of active learning.

In sum, our meta-study suggests that AV technology is educationally effective, but not in the conventional way suggested by the old proverb 'a picture is worth 1000 words'. Rather, according to our findings, the form of the learning exercise in which AV technology is used is actually more important than the quality of the visualizations produced by AV technology. This is not to say that visualization quality does not matter—in a successful learning activity, a well-designed visualization certainly contributes—but rather to say that the form of activity is more important than the form of the visualization.

6.4. What Directions for Future Research Appear Most Fruitful?

Finally, our meta-study findings suggest several directions for future research into AV effectiveness. We conclude by highlighting what we see as the most important of these, organizing our discussion around the topics of independent variables, dependent variables, and scope.

6.4.1. Independent Variables

As we have seen, the majority of AV effectiveness research over the past decade has been guided by just two theories: Epistemic Fidelity and Cognitive Constructivism. In the interest of gaining deeper insight into AV effectiveness, we believe that future research would do well to explore other alternatives. One obvious place to start is with Dual-coding and Individual Differences theories, both of which have already garnered support by studies in our corpus. Dual-coding theory has, in fact, had predictive success in a legacy

Table 8. Two alternative theories of effectiveness that appear to be promising avenues for future research

Theory	Synopsis	Role of AV
Situated Action	AV is a communicative resource for building a shared understanding of algorithms	Communicative resource akin to speech, gesture, and gaze
Sociocultural Constructivism	AV enable people to participate in a community in increasingly central ways	Community artifacts that grant access to central community activities

of experimental studies investigating the pedagogical benefits of instructional hypermedia (e.g. [37]); we would be interested to see it pursued further within the scope of pedagogical AV.

We believe another promising source of theoretical inspiration is recent anthropological research into the social and situated nature of learning and communication. In the style of Table 3 (p. 33), Table 8 outlines two other general theories of effectiveness that come from that line of work. Building on Situated Action Theory (see, e.g. [43]), Roschelle [35] argues that visualizations are one of a multitude of *mediational resources* that help groups of people to negotiate a shared understanding of the topics under study. Alternatively, Sociocultural Constructivism (e.g. [44]) views AV effectiveness at the level of the community of practice, rather than the individual. In particular, AV technology is seen as effective insofar as it provides students with access to the kinds of expert activities normally carried out by algorithms teachers. Hundhausen [5] uses Sociocultural theory to guide his study of the use of AV technology in an undergraduate algorithms course.

6.4.2. Dependent Variables

Measuring procedural and conceptual knowledge in terms of test performance has a solid foundation in practice; indeed, it is the same measure used in traditional computer science courses to evaluate student performance. However, in order to obtain a deeper understanding of AV technology's educational benefits, future empirical studies should, in our opinion, have the courage to explore alternative measures—preferably in concert with traditional measures so as not to disenfranchise themselves from the mainstream.

Since a study's dependent measures are a logical consequence of the learning theory that informs it, alternative measures can be explored within the context of alternative theories. In stark contrast to the learning theories that influenced the studies in our corpus, the two alternative learning theories introduced above (see Table 8) situate knowledge not in the head, but in the broader realms of social interaction (Situated Action) and community reproduction (Sociocultural Constructivism). According to these theories, individual knowledge acquisition does not serve as an adequate measure of learning, which must be measured within the broader context of social interaction within communities of practice.

For example, Situated Action theory would recommend evaluating an algorithm visualization by creating a social situation in which two learners use the visualization to

establish a shared understanding of the underlying algorithm. In such an interaction, *conversation analysis* would be used to determine the extent to which the visualization serves as a mediational resource for the learners. For example, Douglas *et al.* [45] use this evaluation measure in their studies of the human visualization of sorting algorithms.

Likewise, Sociocultural Constructivism would recommend evaluating an algorithm visualization within the scope of its use as a cultural artifact in a community of practice such as the one being reproduced in an undergraduate algorithms course. In such a setting, the effectiveness of the visualization would be judged according to its ability to enable learners gradually to participate more fully in the community—that is, to increasingly take on the identity and roles of the course instructor. For example, within the scope of a junior-level algorithms course, Hundhausen [5] uses ethnographic field techniques to qualitatively evaluate effectiveness in this way.

6.4.3. Scope

We intentionally limited the scope of our meta-study to experimental studies of AV in educational scenarios of use. In so doing, we clearly neglected important lines of related empirical work that are grist for future analysis. Below, we describe three of the most important of these.

Other research techniques. Our meta-study focuses exclusively on experimental studies of AV effectiveness. However, as we pointed out in Section 2.2, controlled experimentation is just one of several empirical evaluation techniques that one might use to study AV effectiveness. Four other relevant empirical techniques that have been employed by past AV effectiveness research include:

- *Usability tests*—these endeavor to identify, diagnose, and ultimately remedy problems with an interactive system's user interface by videotaping a small number of participants as they complete representative tasks with the system (see, e.g. [46]).
- *Ethnographic field techniques*—these include any of the qualitative techniques one might use to conduct a field study [47] in a naturalistic setting—e.g. participant observation, interviews, and artifact collection.
- *Questionnaires and surveys*—these are often used as a complementary source of data in empirical studies. They elicit written responses to a set of questions in which the researcher is interested (see, e.g. [48]).
- *Observational studies*—these investigate some activity of interest in an exploratory, qualitative fashion, often through analysis of videotaped footage of humans interacting with AV technology (see, e.g. [25]).

We believe that each of these alternative empirical methods can play a valuable role in helping us to gain insight into AV effectiveness. For example, usability tests can tell us whether an AV system's user interface is preventing its users from reaping the benefits of a visualization (see, e.g. [49]). Likewise, ethnographic field techniques and observational studies can help us understand how and why AV technology might be effective in a real classroom (e.g. [5]), or in a realistic study session (e.g. [42]). Questionnaires and surveys can help us understand AV technology users' preferences, opinions, and advice regarding AV technology design and use (see, e.g. [50]). Thus, an important area for future research is to perform a meta-study that includes the published empirical studies that

employ these alternative techniques to study AV effectiveness. According to our estimates, close to 30 such studies have been published. .

Other aspects of scenarios. The experimental studies in our corpus have maintained a narrow focus on specific parts of a given scenario of use. For example, in the 'study' scenario considered by most of the experiments, the focus has been on students' use of the technology. The other tasks that must be done 'behind the scenes', such as preparing the visualizations for student use and integrating the technology into a course curriculum, are left unexplored.

While the actual use of AV technology for learning is plainly the crux of any scenario of AV use, the studies' narrow focus on that use has prevented them from obtaining a broader perspective of effectiveness. Even if AV technology is found to be effective as a learning aid, a whole host of other considerations could figure equally prominently in an overall assessment of effectiveness. For example, how long did it take to prepare and set up the AV technology? Given the choice between using conventional materials and AV technology, which will instructors choose, and what considerations do they deem important in making that choice? In his ethnographic fieldwork, Hundhausen [5] began to address these questions within the broader scope of an undergraduate algorithms course.

Other scenarios. Although AV has applications outside of educational scenarios—for example, in algorithms analysis and software engineering—our meta-study has neglected those scenarios. We are, in fact, unaware of a body of research that has empirically evaluated AV technology in non-educational scenarios. We suspect that this is because such research does not, in fact, exist. The empirical evaluation of AV effectiveness in non-educational scenarios is thus an important open area of research.

Notice also that AV is a subarea of *software visualization*, which encompasses not only the visualization of algorithms, but also the visualization of entire software systems. Visualizations of software systems are designed to help members of programming teams do such things as improve system performance [51], comprehend the structure and evolution of software systems [52], and track down bugs [53]. A future meta-study would do well to consider the effectiveness of software visualization in such industrial scenarios. Pertinent guiding questions include the following:

- To what extent has software visualization been effectively applied in industry?
- Has software visualization increased productivity?
- Has software visualization benefited the large software teams typical in industry?
- Has software visualization benefited the kinds of *distributed* programming teams that are common today?

Our suspicion is that few empirical studies that address such questions have been published. If this is indeed true, then an important direction for future research is to subject industrial software visualization systems to the same kind of systematic effectiveness evaluation that educational AV technology has undergone.

Acknowledgements

The first author wrote an early version of this paper as part of his Ph.D. comprehensive exam in the Computer and Information Science Department at the University of Oregon,

and gratefully acknowledges the inspiration and astute guidance of his then-advisor, Sarah Douglas.

References

1. R. Baecker (1975) Two systems which produce animated representations of the execution of computer programs. *SIGCSE Bulletin* **7,** 158–167.
2. M. H. Brown (1988) *Algorithm Animation*. The MIT Press, Cambridge, MA.
3. J. T. Stasko (1990) TANGO: a framework and system for algorithm animation. *IEEE Computer* **23,** 27–39.
4. J. T. Stasko (1997) Using student-built animations as learning aids. In: *Proceedings of the ACM Technical Symposium on Computer Science Education*. ACM Press, New York, pp. 25–29.
5. C. D. Hundhausen (1999) Toward effective algorithm visualization artifacts: designing for participation and communication in an undergraduate algorithms course. Unpublished Ph.D. dissertation, Department of Computer and Information Science, University of Oregon.
6. P. Gloor (1998) Animated algorithms. In: *Software Visualization*: *Programming as a Multimedia Experience* (M. Brown, J. Domingue, B. Price & J. Stasko, eds) The MIT Press, Cambridge, MA, pp. 409–416.
7. J. S. Gurka, & W. Citrin (1996) Testing effectiveness of algorithm animation. In: *Proceedings of the 1996 IEEE Symposium on Visual Languages*. IEEE Computer Society Press, Los Alamitos, CA, pp. 182–189.
8. T. Naps (1990) Algorithm visualization in computer science laboratories. In: *Proceedings of the 21st SIGCSE Technical Symposium on Computer Science Education*. ACM Press, New York, pp. 105–110.
9. R. Baecker (1998) Sorting out sorting: a case study of software visualization for teaching computer science. In: *Software Visualization: Programming as a Multimedia Experience* (M. Brown, J. Domingue, B. Price & J. Stasko, eds) The MIT Press, Cambridge, MA, pp. 369–381.
10. R. Duisberg (1987) Visual programming of program visualizations. In: *Proceedings of the IEEE 1987 Visual Language Workshop*. IEEE Computer Society Press, Los Alamitos, CA.
11. E. Helttula, A. Hyrskykari & K. J. Raiha (1989) Graphical specification of algorithm animations with ALLADDIN. In: *Proceedings of the 22nd Annual Conference on Systems Sciences*. IEEE Computer Society Press, Los Alanitos, CA, pp. 892–901.
12. J. T. Stasko (1991) Using Direct Manipulation to Build Algorithm Animations by Demonstration. In: *Proceedings of ACM CHI '91 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 307–314.
13. A. W. Lawrence, A. N. Badre & J. T. Stasko (1994) Empirically evaluating the use of animations to teach algorithms. In: *Proceedings of the 1994 IEEE Symposium on Visual Languages*. IEEE Computer Society Press, Los Alamitos, CA, pp. 48–54.
14. S. R. Hansen, N. H. Narayanan & D. Schrimpsher (2000) Helping learners visualize and comprehend algorithms. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning* **1**.
15. J. Stasko, A. Badre & C. Lewis (1993) Do algorithm animations assist learning? An empirical study and analysis. In: *Proceedings of ACM INTERCHI'93 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 61–66.
16. A. W. Lawrence (1993) Empirical studies of the value of algorithm animation in algorithm understanding. Unpublished Ph.D. dissertation, Department of Computer Science, Georgia Institute of Technology.
17. M. D. Byrne, R. Catrambone & J. T. Stasko (1999) Evaluating animations as student aids in learning computer algorithms. *Computers & Education* **33,** 253–278.
18. B. A. Price, R. M. Baecker & I. S. Small (1993) A principled taxonomy of software visualization. *Journal of Visual Languages and Computing* **4,** 211–266.
19. J. Bazik, R. Tamassia, S. Reiss & A. van Dam (1998) Software visualization in teaching at Brown University. In: *Software Visualization: Programming as a Multimedia Experience* (M. Brown, J. Domingue, B. Price & J. Stasko, eds). The MIT Press, Cambridge, MA, pp. 383–398.

20. J. S. Gurka (1996) Pedagogic Aspects of Algorithm Animation. Unpublished Ph.D. dissertation, Computer Science, University of Colorado.
21. D. R. Goldenson & B. J. Wang (1991) Use of structure editing tools by novice programmers. In: *Empirical Studies of Programmers: Fourth Workshop*. Aldex Publishing, Norwood, NJ, pp. 99–120.
22. K. C. Cox & G. C. Roman (1994) An evaluation of the Pavane visualization system. Technical Report WUCS-94-09, Department of Computer Science, Washington University of St. Louis, St. Louis, MO, April 1994.
23. J. Nielsen (1992) Finding usability problems through heuristic evaluation. In: *Proceedings of ACM CHI '92 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 373–380.
24. P. G. Polson, C. Lewis, J. Rieman & C. Wharton (1992) Cognitive walkthroughs: a method for theory-based evaluation of user interfaces. *International Journal of Man-Machine Studies* **36,** 741–773.
25. D. J. Gilmore (1990) Methodological issues in the study of programming. In: *Psychology of Programming* (J.-M. Hoc, T. R. G. Green, R. Samurcay & D. J. Gilmore, eds). Academic Press, San Diego, pp. 83–98.
26. B. Price (1990) A framework for the automatic animation of concurrent programs. Unpublished M.S. thesis, Department of Computer Science, University of Toronto.
27. P. Mulholland (1998) A principled approach to the evaluation of SV: a case study in Prolog. In: *Software visualization: Programming as a Multimedia Experience* (M. Brown, J. Domingue, B. Price & J. Stasko, eds). The MIT Press, Cambridge, MA, pp. 439–452.
28. M. E. Crosby & J. Stelovsky (1995) From multimedia instruction to multimedia evaluation. *Journal of Educational Multimedia and Hypermedia* **4,** 147–162.
29. L. V. Hedges & I. Olkin (1985) *Statistical Methods for Meta-Analysis*. Academic Press, Orlando.
30. C. Chen & Y. Yu (2000) Empirical studies of information visualization: a meta-analysis. *International Journal of Human–Computer Studies* **53,** 851–866.
31. C. D. Hundhausen, & S. A. Douglas (2000) Using visualizations to learn algorithms: should students construct their own, or view an expert's? In: *Proceedings 2000 IEEE International Symposium on Visual Languages*. IEEE Computer Society Press, Los Alamitos, pp. 21–28.
32. D. J. Jarc, M. B. Feldman & R. S. Heller (2000) Assessing the benefits of interactive prediction using web-based algorithm animation courseware. In: *Proceedings SIGCSE 2000.* ACM Press, New York, pp. 377–381.
33. C. Kann, R.W. Lindeman, & R. Heller (1997) Integrating algorithm animation into a learning environment. *Computers & Education* **28,** 223–228.
34. E. Wenger (1987) *Artificial Intelligence and Tutoring Systems*. Morgan Kaufmann, Los Altos, CA.
35. J. Roschelle (1990) Designing for conversations. Presented at AAAI Symposium on Knowledge-Based Environments for Learning and Teaching, Stanford, CA, 1990.
36. A. Newell & H. A. Simon (1972) *Human Problem Solving.* Prentice-Hall, Englewood Cliffs, NJ.
37. R. E. Mayer & R. B. Anderson (1991) Animations need narrations: an experimental test of a dual-coding hypothesis. *Journal of Educational Psychology* **83,** 484–490.
38. A. Paivio (1983) The empirical case for dual coding. In: *Imagery, Memory, and Cognition: Essays in Honor of Allan Paivio* (J. C. Yuille, ed.). Lawrence Erlbaum Associates, Hillsdale, NJ.
39. C. Cooper (1997) *Individual Differences*. Oxford Illustrated Press, Oxford.
40. R. Riding & S. Rayner (1998) *Cognitvie Styles and Learning Strategies*. David Fulton Publishers, London.
41. L. B. Resnick (1989) Introduction. In: Knowing, *Learning, and Instruction: Essays in Honor of Robert Glaser* (L. B. Resnick, ed.). Erlbaum, Hillsdale, NJ, pp. 1–24.
42. C. Kehoe, J. T. Stasko & A. Taylor (2001) Rethinking the evaluation of algorithm animations as learning aids: an observational study. *International Journal of Human–Computer Studies* **54,** 265–284.
43. L. A. Suchman (1987) *Plans and Situated Actions: The Problem of Human–Machine Communication*. Cambridge University Press, New York.
44. J. Lave & E. Wenger (1991) *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press, New York, 138pp.

45. S. A. Douglas, C. D. Hundhausen & D. McKeown (1996) Exploring human visualization of computer algorithms. In: *Proceedings 1996 Graphics Interface Conference*. Canadian Graphics Society, Toronto, CA, pp. 9–16.
46. J. Rubin (1994) *Handbook of Usability Testing*. John Wiley and Sons, New York.
47. R. Sanjek (1995) Ethnography. In: *Encyclopedic Dictionary of Social and Cultural anthropology* (A. Barnard & J. Spencer, eds.). Routledge, London.
48. W. Foddy (1994) *Constructing Questions for Interviews and Questionnaires: Theory and Practice in Social Research*. Cambridge University Press, Cambridge.
49. D. Lavery & G. Cockton (1995) A pilot study of early usability evaluation methods for software visualisations. FIDE Technical Report FIDE/95/141[, University of Glasgow, Glasgow, Scotland.
50. A. Badre, M. Baranek, J. M. Morris & J. T. Stasko (1992) Assessing program visualization systems as instructional aids. In: *Computer Assisted Learning*, ICCAL '92 (I. Tomek, ed.). Springer-Verlag, New York, pp. 87–99.
51. D. Kimelman, B. Rosenburg & T. Roth (1994) Strata-Various: Multi-layer visualization of dynamics in software system behavior. In: *Proceedings Visualization* '94 IEEE Computer Society Press, Los Alamitos, CA, pp. 172–178.
52. M. J. Baker & S. G. Eick (1995) Space-filling software visualization. *Journal of Visual Languages and Computing* **6**, 119–133.
53. T. Shimomura & S. Isoda (1991) Linked-list visualization for debugging. *IEEE Software* **8,** 44–51.
54. B. A. Myers (1990) Taxonomies of visual programming and program visualization. *Journal of Visual Languages and Computing* **1,** 97–123.
55. N. C. Shu (1988) *Visual programming*. Van Nostrand Reinhold, New York.
56. M. H. Brown (1988) Perspectives on algorithm animation. In: *Proceedings of the ACM SIGCHI '88 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 33–38.
57. J. T. Stasko & C. Patterson (1992) Understanding and characterizing software visualization systems. In: *Proceedings of the 1992 IEEE Symposium on Visual Languages*. IEEE Computer Society Press, Los Alamitos, CA, pp. 3–10.
58. G. Singh & M. H. Chignell (1992) Components of the visual computer: a review of relevant technologies. *Visual Computer* **9,** 115–142.
59. E. Kraemer & J. T. Stasko (1993) The visualization of parallel systems: an overview. *Journal of Parallel and Distributed Computing* **18,** 105–117.
60. G. C. Roman & K. C. Cox (1993) A taxonomy of program visualization systems. *IEEE Computer* **26,** 11–24.