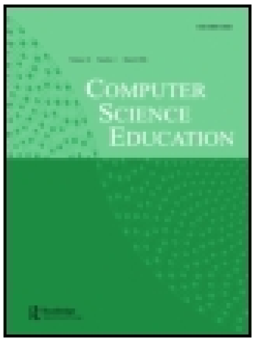

Computer Science Education

ISSN: 0899-3408 (Print) 1744-5175 (Online) Journal homepage: https://www.tandfonline.com/loi/ncse20

A systematic literature review of student engagement in software visualization: a theoretical perspective

Abdullah Al-Sakkaf, Mazni Omar & Mazida Ahmad

COMPUTER SCIENCE EDUCATION
https://doi.org/10.1080/08993408.2018.1564611

ARTICLE

A systematic literature review of student engagement in software visualization: a theoretical perspective

Abdullah Al-Sakkaf, Mazni Omar and Mazida Ahmad

Human-Centered Computing Research Lab, School of Computing, Universiti Utara Malaysia, Sintok, Malaysia

ABSTRACT

Background and Context: In spite of the decades spent developing software visualization (SV), doubts still remain regarding their effectiveness. Furthermore, student engagement plays an important role in improving SV effectiveness as it is correlated with many positive academic outcomes. It has been shown that the existing SV has failed to engage students effectively.

Objective: Therefore, there is a need to understand the theories behind SV design from the engagement perspective to produce a roadmap for future tool construction. The aim of this study was to identify the theories have been used in literature to explain or construct student engagement with SV in computer science courses for novices.

Method: We performed a systematic literature review that identified a total of 58 articles published between 2011 and 2017, which were then selected for the study. However, among them, only 18 articles had discussed their theoretical framework.

Findings: The results of this study show a richness in the theoretical framework obtained from different disciplines, however, constructivism is still dominant in the computing education research (CER) domain. It is evidently clear from the findings that the theories generated from the CER domain are needed to enhance the effectiveness of SV.

Implications: As a result of this review, we suggest several design principles and engagement attributes to be considered while creating an engaging SV.

ARTICLE HISTORY
Received 22 March 2018
Accepted 28 December 2018

KEYWORDS
Systematic literature review; software visualization; program visualization; learning theory; introductory programming education; engagement

1. Introduction

The teaching of programming is one of the widely studied areas in computing education research (CER). It has long been viewed that programming is a challenging and difficult course for novices. Despite decades of research to decrease the failure and drop-out rates in introductory to programming courses (CS1), where the failure rates are higher as compared to other courses, problems remain in existence (Vihavainen, Airaksinen, & Watson, 2014). Therefore, software visualization (SV) is recognized as a promising solution to

CONTACT Abdullah Al-Sakkaf abbdullah1@hotmail.com

enhance programming education for novices. It can be defined as "the use of the crafts of typography, graphic design, animation, and cinematography with modern human-computer interaction technology to facilitate both the human understanding and effective use of computer software" (Price, Baecker, & Small, 1993). Algorithm visualization, and programming visualization are two main types of SV (Price et al., 1993). Many SV has been developed since 1980; however, there are still some critical issues regarding these tools, which include: low adoption rate, short-lived research prototype, and high concern about their effectiveness and their impact on the learning outcome (Fouh et al., 2014; Isohanni & Järvinen, 2014; Shaffer, Akbar, Alon, Stewart, & Edwards, 2011; Sorva, Karavirta, & Malmi, 2013). Surprisingly, the pedagogical effectiveness of SV has shown mixed results (Ben-Ari et al., 2011; Cooper, Shaffer, Edwards, & Ponce, 2014; Hundhausen, Douglas, & Stasko, 2002; Shaffer et al., 2010; Sorva et al., 2013).

In recent years, researchers have shown an increased interest in student engagement, as it plays a significant role in improving the effectiveness of SV and educational technologies in general (Isohanni & Järvinen, 2014; Naps et al., 2003; Sorva et al., 2013). The term student engagement has been used to refer to the learner involvement with the teaching system that includes the interaction with the system (tools), instructor, or other students within the system. The role of engagement in SV has received increased attention over the last decade owing to the influential work by Hundhausen et al. (2002) and Naps et al. (2003). SV can promote better learning by increasing student engagement with the tool. However, it has been shown that existing SV has failed to engage students effectively. Despite the importance of student engagement in the successful design of SV tools, the focus on how to improve engagement when constructing SV tools is very limited from a theoretical viewpoint. This has negatively impacted the students' programming skills because poor engagement with SV leads to poor educational benefits. Thus, to remedy the situation, studies on engagement will have to consider the possible theories and techniques for student engagement with SV within the learning process, thereby allowing instructors and designers of these tools can improve and increase student engagement.

Researchers have acknowledged that a solid theoretical foundation is fundamental for an effective educational technology design (Malmi et al., 2014). As computer science (CS) education is an interdisciplinary field, Fincher and Petre (2004) identified that there is a need to link the research to a relevant theory for CS educational research. Likewise, Hidalgo-Céspedes, Marín-Raventós, and Lara-Villagrán (2016) observed in the visualization studies that there is a lack of consideration for learning theories on building the theoretical framework. In the same vein, Shaffer et al. (2010) advocated that the theoretical foundations for creating effective algorithm visualization appear to be steadily improving. However, there is no agreement on a clear definition of the term "theory" (Bikner-Ahsbahs & Prediger, 2010). This paper will use Malmi et al. (2014)'s definition, in which theory is defined as “a broad class of concepts that aims to provide a structure for conceptual explanation or established practice and use such terms as theories, models and frameworks (TMF) to describe particular manifestations of the general concept of theory”. The role of engagement with SV is important, yet there is still a need to expand our knowledge in that matter. This article is the first step toward this by exploring the current theoretical perspective to build a successful SV that engages the student.

In this paper, we present and discuss the results of a study where we identified and analysed the theoretical foundation reported in 58 papers published between 2011 and 2017. The overall objective of this review was to address the theories, frameworks and models used in the design of the engaging SV by conducting a systematic literature review (SLR). The motivation for this paper is to help SV systems’ designers to improve the effectiveness of learning and enhance learning outcomes in CS1 using SV by promoting student engagement. Also, it will help to explore the good practices and theoretical bases founded in the current literature. In addition, it will help to provide useful guidelines for upcoming research to build their study foundation – whether a study focus for analysis, prediction, or designing a tool. In this study, a focus was placed on the design and evaluation of the software visualization system in the introductory programming course for novices from the perspective of the students’ engagement and their impact on the learning outcomes and the effectiveness of these systems.

2. Background

2.1. Student engagement

In education, student engagement plays an important role in learning process (Ericson, Guzdial, & Morrison, 2015; Kahu, 2013; Schindler, Burkholder, Morad, & Marsh, 2017). It initially emerged as an academic concept during the 1970s (Schindler et al., 2017). Engagement has received significant attention due to the movement towards student-centered education, constructivist instructional methods and promising studies suggesting relationships between student engagement and positive academic outcomes (Schindler et al., 2017). As Shulman (2002) notes: “Learning begins with student engagement, which in turn leads to knowledge and understanding”. The roles of engagement in education have been studied extensively in order to handle educational problems such as the high student drop-out rate, low academic achievement and high rates of student boredom and alienation (Fredricks, Filsecker, & Lawson, 2016; Kahu, 2013; Schindler et al., 2017). Furthermore, the increased engagement benefits include improving the student motivation (Urquiza-Fuentes & Velázquez-Iturbide, 2013), enhancing student retention (Pechenkina, Laurence, Oates, Eldridge, & Hunter, 2017), increasing study time (Sorva et al., 2013), and improving the student performance in the course (Urquiza-Fuentes & Velázquez-Iturbide, 2013).

Despite its common usage, engagement is used in different disciplines to mean different things (Fredricks et al., 2016; Schindler et al., 2017). According to the National Survey of Student Engagement (NSSE), student engagement refers to the time that students allocate to educational activities to contribute to the desired outcomes and as the quality of their related efforts (Kuh, 2009). In addition, engagement is defined as "the extent to which students are engaging in activities that higher education research has shown to be linked with high-quality learning outcomes' (Krause & Coates, 2008). However, engagement has evolved to be viewed as more of a multidimensional construct. Fredricks, Blumenfeld, and Paris (2004) defined engagement based on each engagement dimension, which is stated as follows: (1) cognitive; (2) behavioral; and (3) emotional. Behavioral engagement has been defined in terms of participation, effort, attention, persistence, positive conduct and the absence of disruptive behavior. Emotional engagement has been defined as the extent of positive emotional reactions to teachers, peers and classroom activities, as well as valuing learning and having interest in the learning content. Cognitive engagement is defined in terms of self-regulated learning, using deep learning strategies and exerting the necessary effort to comprehend complex ideas or master difficult skills. Furthermore, the Fredricks et al. (2004)"s definition received increased attention in recent studies, due to its broad scope, and incorporating different types of engagement that help to capture a wider range of student experiences (Reschly & Christenson, 2012; Schindler et al., 2017). In addition, this definition is student-centered, where its focus is on direct student indicators rather than combining student indicators with confounding variables, such as faculty behavior, curriculum design and campus environment (Schindler et al., 2017).

2.2. Technology and student engagement

"Successful technologies are not just usable; but they engage the user" (O'Brien & Toms, 2008). Recently, there has been a growing interest in determining how technology influences student engagement (Katuk, Omar, & Halim, 2014; Schindler et al., 2017). Along with this growth, many researchers have developed new forms of student engagement with learning. Student engagement with systems varies based on the purpose and context of a system, where some systems are designed for one-time user engagement, while others are designed to engage the user more frequently (i.e. daily, or during a course lifetime). Educational technologies can promote better learning by increasing the learner interaction with the system. However, in some cases, the role of quality of engagement has become more important than the amount of time a learner spends on the system. Several models or frameworks were proposed to explain engagement with technology (e.g. Gunuc & Kuzu, 2015; O'Brien & Toms, 2008; Schindler et al., 2017). However, the existing frameworks in the field are based on a specific domain.

2.3. The role of engagement with software visualization

The evidence suggests that student engagement is an important factor for designing and building an effective and successful SV tool (Isohanni & Järvinen, 2014; Naps et al., 2003; Sorva et al., 2013; Urquiza-Fuentes & Velázquez-Iturbide, 2013; Velázquez- Iturbide, Hernán-Losada, & Paredes-Velasco, 2017). However, there is still a need for exploring the role of engagement with SV (Sirkiä & Sorva, 2015; Sorva et al., 2013). Hundhausen et al. (2002) concluded that how AV technology used is actually more important than the quality of the visualization produced by AV. Similarly, Naps et al. (2003) argued that these visualizations have little educational value unless it supports the active engagement of a learner. In other words, how students interact with visualization has a significant impact on their learning from visualization. As a consequence, over the past ten years, researchers have focused more on investigating the educational effectiveness of the tools and the questions of how to engage students with visualizations in a better way (Fouh, Akbar, & Shaffer, 2012; Isohanni & Knobelsdorf, 2011; Sorva et al., 2013). Recently, through the newly-developed tools, the major focus has shifted toward increasing the interaction and engagement of learner. It has been debated how research on software visualizations in programming education should be developed (Isohanni & Järvinen, 2014). The current SV systems have a problem with the lack of engaging the student to use them; as a result, it has negatively impacted their learning outcome (Hosseini, Sirkiä, Guerra, Brusilovsky, & Malmi, 2016).

Several taxonomies were proposed to explain the student engagement level of interaction with the visualization tools, which are: Engagement Taxonomy (ET) by Naps et al. (2003), Extended Engagement Taxonomy (EET) by Myller, Bednarik, Sutinen, and Ben-Ari (2009), and 2 Dimensional Engagement Taxonomy (2DET) by Sorva et al. (2013). These taxonomies hypothesised that using a higher level of engagement would improve learning outcome. Naps et al. (2003) established an ET for a student engagement with visualization technologies, which states the following levels: no viewing, viewing, responding, changing, construction, or presenting. According to Naps et al. (2003), these taxonomies neither treat the level as a hierarchy and nor do they have an ordinal scale. He advocates that more engagement is better and that mixing different engagement levels could lead to better learning outcomes. ET was developed to describe the possible engagement levels that can be tested in an experiment (Kno- belsdorf, Isohanni, & Tenenberg, 2012). Myller et al. (2009) refined ET by introducing four more levels to EET, which comprise the following: no viewing, viewing, controller viewing, entering input, responding, changing, modifying, constructing, presenting, and reviewing. He hypothesised that increasing the level of engagement will improve the collaboration process. Finally, Sorva et al. (2013) constructed the 2DET, which has two dimensions: (1) direct engagement with visualization; and (2) content ownership. The direct engagement consists of seven levels, which are: no viewing, viewing, controller viewing, responding, applying, presenting, or creating. The second dimension -content ownership- is concerned with an indirect form of engagement that results from the student's relationship with the target software; that is, the content of the visualization.

ET has historically been one of the most commonly used evaluation frameworks (Banerjee, Murthy, & Iyer, 2015; Velázquez-Iturbide et al., 2017). Even though previous studies measured the effect of engagement level on learning outcomes, their results were not consistent (Banerjee et al., 2015). For instance, Banerjee, Murthy, and Iyer (2013)'s study failed to find any significant differences in the test scores when using different engagement levels. This taxonomy did not provide any further explanation as to why each higher engagement level provides a better result compared with the previous one (Malmi et al., 2014). In other words, the taxonomy did not underpin how each engagement contributes to different engagement constructs. However, there is still a need for further discussion on the level. For example, Sorva et al. (2013) argued that the level of engagement in ET was unable to fully explain the direct engagement level. He also stated that ordering of level failed to match with Bloom's revised taxonomy. Likewise, it is reported that ET failed to provide a comprehensive explanation of other forms of student engagement outside the scope (Knobelsdorf et al., 2012). In general, the aforementioned taxonomies do not fully explain the roles and engagement and how to increase student engagement with educational technologies. Along similar lines, Cetin and Andrews-Larson (2016) argues that the students' engagement level is not the only determinant of student success in visualization. Isohanni and Knobelsdorf (2011) pointed out that rather than only focusing on how the students engage, there is also a need to understand how the students interact with the system and how to keep them in an engaged mode.

Thus, little is known about the relationships between engagement levels and learning outcome and further studies are needed to understand engagement with visualization in depth. So far, several unanswered questions still remain unanswered, such as: (1) what are the factors that enhance student engagement with SV? (2) How to keep students engaged with SV? (3) and how to enhance the quality of student engagement with SV?

Several studies have investigated several designs principles, which borrowed from various theories from different disciplines, to improve student engagement with SV. Such approaches, however, have failed to address the main problem, which could be attributed to several reasons. First, It has been established that engagement is a multi-dimension construct, these study main focus on behavioral engagement and neglected the remain dimensions, despite the importance of all dimensions to build an engaging experience for a student. Considering different design principles to cover these different dimensions could improve student engagement in a holistic way. In this study, we hypothesized that by considering all aspect of engagement using different design principles we could improve student engagement with SV.

3. Related works

Literature reviews in CER domain are common where the focus is usually on the areas that lack precision. To date, there are few related reviews to this study, which could be categories based in the following themes. First, literature reviews that focused on the theoretical aspects of the CER literature in general (e.g. Lishinski, Good, Sands, & Yadav, 2016; Malmi et al., 2014). For instance, the SLR done by Malmi (2014) focus on analyzing TMFs used by CER from 2005 to 2011. He claimed that neither educational theories nor other theories from the different fields can be used to explain the process and challenges faced in the teaching and learning in the CER area. Furthermore, there is a lack of fundamental theories to be used in explaining how students learn computer science (Malmi et al., 2014). As SV is part of CER, the same goes for it. A further important observation by Malmi et al. (2014) stated that half of the papers, included in his review, were not build based on previous theoretical foundations.

Second, reviews that focused on software visualization (e.g. Hidalgo-Céspedes et al., 2016; Hundhausen et al., 2002; Sorva et al., 2013; Urquiza-Fuentes & Velázquez- Iturbide, 2009). Urquiza-Fuentes and Velázquez-Iturbide (2009) study surveyed successful evaluations of SV and related them to the ET. Sorva et al. (2013) review, for example, explore the existing PV systems and summarize the evaluation studies for these tools. In addition, the review acknowledged that the results were unclear with respect to student engagement. The study demonstrated the need for a clear framework to be used in future research regarding student engagement and PV. In the same vein, a recent study by Hidalgo-Céspedes et al. (2016) concluded that very few systems used learning theory as a theoretical foundation. The study identified the principles that could contribute to the effectiveness of tools based on Vygotsky's learning theory.

Table 1 list the related SLRs and how they compared to the current study. Previous studies have not dealt with the role of engagement and SV from theoretical perspective in depth. Malmi et al. (2014) study have generally been concerned with TMFs in the CER domain. It does not give a conclusion on how to utilize these theories to build and design an effective SV in term of engagement. To date, no review has yet been conducted to underpin the theoretical background for the role of engagement in SV. In conclusion,

Table 1. Comparison of related SLR.

Author	Period	Papers	Focus
Urquiza- Fuentes Velázquez-Iturbide (2009)	<2009	33	Reviews successful educational experiences in using SV
Sorva et al. (2013)	1979–2012	n/a	Survey PV systems in the last 3 decades
Malmi et al. (2014)	2008–2011	308	Identifies theories, models, and frameworks that have been used in CER literature
Hidalgo- Céspedes et al. (2016)	2013–2016	36	Examine learning principles on recent PV systems
Our study	2011–2017	58	Identifies TMFs that have been used to explain or construct learner engagement in SV

previous studies showed that the challenge in the theoretical foundation in CER and SV still exists.

4. Method

This researcher chose a systemic literature review method in order to answer the research question. The protocol was built following the guidelines provided by (Kitchenham, Budgen, & Brereton, 2015). The protocol involves three main phases, all of which are shown in Table 2. The main research question is "Which theories have been used to explain or construct student engagement in SV?". Beyond looking for evidence that theories are being used, we are interested in understanding how do we apply these theories to build engaging SVs in CS1 courses.

The search strategy was formulated where the main keywords were engagement and software visualization. Initially, the terms relating to SV were identified, such as program and algorithm visualizations. In the literature, animation was sometimes used instead of visualization (i.e. algorithm animation). The final search string was:

(engagement) AND ("software visualization" OR "software visualisation" OR
"program visualization" OR "program visualisation" OR
"algorithm visualization" OR "algorithm visualisation"
OR "software animation" OR "program animation" OR
"algorithm animation")

The search covered articles published between 2011 and March 2017. This researcher used the following databases in the search to identify and collect relevant

Table 2. Phases of the SLR by Kitchenham et al. (2015).

(1) Planning the review
  (a) Identification of the need for a review
  (b) Development of a review protocol
(2) Conducting the review
  (a) Identification of research
  (b) Selection of primary studies
  (c) Study quality assessment
  (d) Data extraction
  (e) Data synthesis
(3) Reporting the review

Table 3. Inclusion and exclusion criteria.

Inclusion criteria	
1	Paper must be on the use of SV in teaching CS1 course that involved higher education students
2	Paper must be present SV tool(s) or discussed program visualization in general
3	Paper must report empirical results
Exclusion criteria	
1	Paper will be excluded if it is duplicated
2	Books, Letters, editorials and position papers will all be excluded
3	Paper will be excluded if it is not written in English
4	Paper will be excluded if full text is not available
5	Paper will be excluded if it is not a full paper (abstract only, poster, or workshop)
6	Paper will be excluded if it is not related to programming courses
7	Paper will be excluded if it is not a primary study
8	Paper is not a SV as defined in this paper
9	Paper is not involving higher education students

manuscripts: (1) IEEE Xplore; (2) ACM Digital Library; (3) Springer Link; (4) Science Direct; (5) Wiley Online; and (6) Scopus. The search strategy was validated by checking the five key papers. The selection process was separated into three stages after removing duplicated titles. In the first stage, the exclusion criteria applied to articles' titles only. Then, we screened the paper based on its abstract and conclusion. Finally, the paper was selected based on the full text screening. Table 3 presents the inclusion and exclusion criteria applied to all the retrieved studies.

4.1. Extracting of theories

To investigate the theoretical foundation in SV, the data extraction form was designed to collect all the information needed to address the review goals. The data extraction form was made up of six items, as shown in Table 4. The main outcome for this review was to identify the fundamental theories used in the area of SV, which is not an easy task. TMF provides a useful account of how papers utilise previous theoretical backgrounds when analyzing, explaining, or designing their work. Further, theoretical backgrounds also guide research to propose a hypothesis. In addition, it helps to build a common terminology to be used, which will help to improve the communication. Theory refers to well-established theories, such as constructivism or flow theory. On the other hand, the terms model and framework refer to established conceptual constructs; for example, Naps' taxonomy or Bloom's taxonomy. Given that each paper could present different TMFs when discussing the related works, it is important to ensure that

Table 4. Items and descriptions of the data extraction form.

Item	Description
Name	The name given for the theory by the author if available
Theory Overview	Identify the theory(s) used in this study
Terminology	Determine whether it is theory, model, or framework
Types of Theory	Determine the type of theory
Theory Disciplines	From where this theory was originated
Purpose of Theory	To explain in more details the purpose of the theory in the study

the selected TMFs were clearly used or developed in the paper. To explore the theoretical background of this paper, we adapted the TMMCER classification system as proposed by Malmi et al. (2010). Thus, the theory selection focused on the following part of each paper:

(1) If there is a dedicated theoretical section.
(2) If it is stated in the abstract, introduction, or in the design of tool/study description that this work is based on a specific theoretical background.
(3) In the discussion section, if the gained result was interpreted and explained using a theory or theoretical background.

The extraction process focused on an explicit theoretical background mentioned in the paper, and it was the focus of the search. However, we shared a concern about finding a large portion of papers that did not use or mention the theoretical background for their studies, as revealed by Malmi et al. (2014). To identify the theoretical foundation of each paper, the researchers utilised the following process proposed by Hannay, Sjoberg, and Dyba (2007):

(1) Candidacy for theory:

- The mention of the terms theory, model, framework, or grammatical derivatives thereof, together with at least one reference, or, alternatively,
- The identification of constructs and relationships in a body of conceptual argumentation delineated by diagrams, words, etc.; and

(2) Explanation of the cause-effect relationship:

- Its use in the roles of design, post hoc explanation, testing, modification, proposal, or basis.

After a theory is identified, its form determines its type. Each of these theories has a different use or goal in a research study. For example, it could be used to demonstrate how something should be used or developed in practice, or how to explain a phenomenon. Besides, theories are used to explain the relationships among constructs. Gregor (2006) proposed a taxonomy to classify the structural nature of theories in information systems based on their goals, as shown in Table 5. Next, the theories will be categorised based on from where this theory was originated (e.g. CER, Computing, Education, Psychology, etc.). This is important to understand how the area of SV is developed. This item will be identified based on the source of theory disciplines. Finally, the purpose of theory and how it is used in the study context will be explained.

Table 5. A taxonomy of theory types.

Type	Description
1 Analysis	The theory does not extend beyond analysis and description. No causal relationships among phenomena are specified and no predictions are made (Says what is)
2 Explanation	The theory provides explanations but does not aim to predict with any precision. There are no testable propositions (Says what is, how, why, when, and where)
3 Prediction	The theory provides predictions and has testable propositions but does not have well-developed justificatory causal explanations (Says what is and what will be)
4 Explanation and prediction	The theory provides predictions and has both testable propositions and causal explanations (Says what is, how, why, when, where, and what will be)
5 Design and action	The theory gives explicit prescriptions (e.g. methods, techniques, principles of form and function) for constructing an artifact (Says how to do something)

Source: (Gregor, 2006)

5. Results and discussion

5.1. Analysis of the selected articles

The query string retrieved 432 results from six databases. After we discarded 66 duplicated articles, 366 articles were included in the selection process. In stage one, the exclusion criteria were applied to the title of each paper which resulted in 117 excluded articles. At stage two, 140 articles were excluded based on the abstract and conclusion section of the articles. At the final stage, the remaining 109 articles were screened based on a full-text criterion. In this stage, another 51 articles were excluded. By the end of the selection process, a total of 58 articles matched the selection criteria for reviewing. The full selection phases and number of papers identified at each stage can be seen in Figure 1.

Of the 58 selected articles, just over one-third (n = 18) of articles mentioned the theoretical framework of the study. Among these articles, when articles' TMFs were used for purposes other than explaining the role of engagement, they were excluded from the study. Hereinafter, the paper will discuss the TMFs extracted from these 18 articles. These articles were distributed based on the number of TMFs and used as follows: 12 articles use a single TMF, five use two TMFs and only one use three TMFs. Table 6 lists the extracted TMFs from these papers. In total, the result revealed 17 distinct TMFs that have been used in program visualization research. As shown in Table 7, more than half of the used TMFs derived from Education disciplines, while around a third of the theories derive from the Psychology discipline. On the other hand, Naps taxonomy, which was used in five different studies, was the only extracted TMF driven from the CER discipline.

In terms of the type of TMF, 60% of researchers found TMF identified as a theory, while 36%, identified it as a framework and only 4% identified it as a model. It is important to note that Naps taxonomy is treated as a taxonomy in the existing literature; however, some of the literature also defines it as a framework. For the sake of simplicity, we treated Naps taxonomy as a framework on account of the fact that it has been heavily tested in the domain of visualization research. Moreover, Naps taxonomy goes beyond a simple

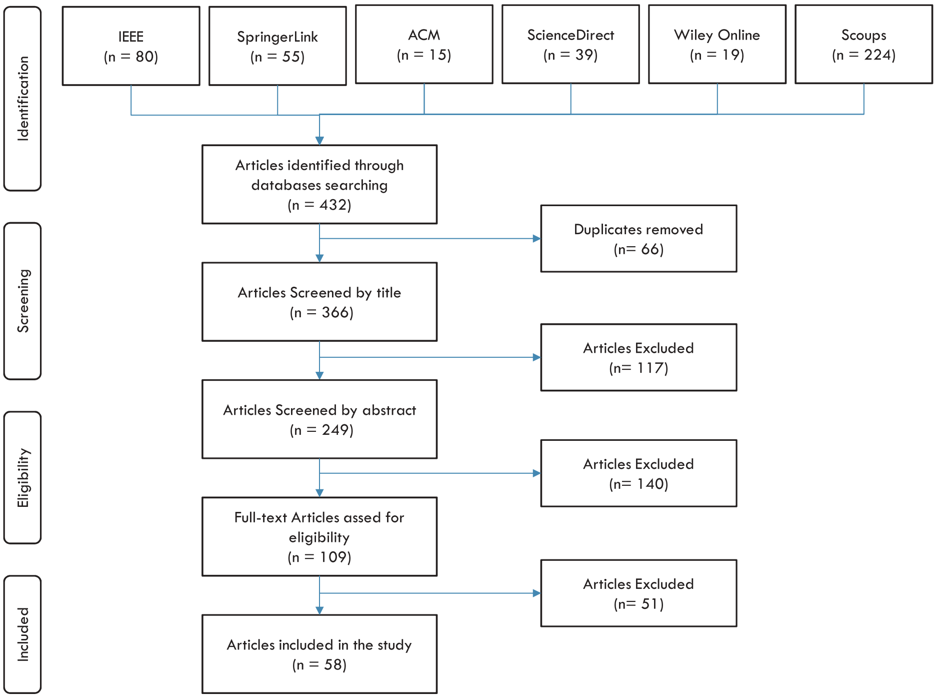

Figure 1. Study selection process.

taxonomy as it provides a guide to design the experiment for the evaluation of a system. However, Table 8 indicates the type of theory based on Gregor (2006)'s taxonomy of theory types in information systems. More than half of these TMFs were used for design and action, as most of the existing research discusses how to develop an effective artifact. This result may be explained by the fact that engaging design principles were challenging in developing effective SV, as a result, designers used different theories to overcome this problem. However, just over third were used for the purposes of analysis. Naps taxonomy (n = 5), and Bloom taxonomy (n = 2) are widely used to analyze the experimental result of using SVs in the classroom.

5.2. *Extracted theories*

5.2.1. *Education theories*

In the selected papers, education theories are dominant. The data extraction revealed 9 different educational theories used a total of 13 times. This result makes sense because learning theories and paradigms are always a good starting point to design effective education technology. The learning theories are important to understand the process of learning and how knowledge and skill are acquired. Behaviorism, constructivism, cognitivism, and humanism are predominant learning theories that have guided education and instructional

Table 6. List of extracted TMF.

	Author	Theory	TMF	Type	Disciplines
1	Velázquez-Iturbide et al. (2017)	Self-determination theory	F	Prediction	Psychology
2	Shi, Min, and Zhang (2017)	SOLO taxonomy	F	Prediction	Education
3	Végh and Takáč (2017)	Constructivism	T	Design and action	Education
4	Velázquez-Iturbide, Hernán-Losada, and Pérez-Carrasco (2016)	Cognitive Load Theory	T	Design and action	Education
5		Variation Theory	T	Design and action	Education
6	Cetin and Andrews-Larson (2016)	Constructionism	T	Analysis	Education
7	Odisho, Aziz, and Giacaman (2016)	Active learning	T	Design and action	Education
8		Constructive Alignment	M	Design and action	Education
9	Yohannis and Prabowo (2015)	Theory of motivation	T	Design and action	Psychology
10		Flow Theory	T	Design and action	Psychology
11	Gordon and Guo (2015)	Cognitive Load Theory	T	Design and action	Education
12	Velázquez-Iturbide, Pérez-Carrasco, and Debdi (2015)	Bloom Taxonomy	F	Analysis	Education
13	Banerjee et al. (2015)	Naps Taxonomy	F	Analysis	CER
14	Moreno, Sutinen, and Joy (2014)	Constructivism	T	Explanation	Education
15	Affandy et al. (2014)	Naps Taxonomy	F	Analysis	CER
16	Buchanan and Laviola Jr. (2014)	Affordance Theory	T	Design and action	Psychology
17	Mishra et al. (2014	Flow Theory	T	Design and action	Psychology
18	Urquiza-Fuentes and Velázquez-Iturbide (2013)	Naps Taxonomy	F	Analysis	CER
19		Cognitive Load Theory	T	Design and action	Education
20		Dual-coding theory	T	Design and action	Psychology
21	Hall, Fouh, Breakiron, Elshehaly, and Shaffer (2013)	Theory of change	T	Design and action	Psychology
22		Social cognitive theory	T	Design and action	Education
23	Banerjee et al. (2013)	Naps Taxonomy	F	Analysis	CER
24	Urquiza-Fuentes and Velázquez-Iturbide (2012)	Naps Taxonomy	F	Analysis	CER
25		Bloom Taxonomy	F	Analysis	Education

Table 7. Theoretical foundation type.

Discipline	Number of theories	Frequency (*n*)	Percentage (%)
CER	1	5	20.00
Education	9	13	52.00
Psychology	7	7	28.00

Table 8. Type of selected theories.

	Discipline	Frequency (*n*)	Percentage (%)
1	Analysis	8	32%
2	Explanation	1	4%
3	Prediction	2	8%
4	Explanation and prediction	0	0%
5	Design and action	14	56%

developments. Each orientation is characterised by different learning theories and representative principles, and these orientations may be related to the surge of interest in computer-based learning. Despite the variety of learning theories, constructivist theories are the most dominant theories in the SV field (Hundhausen et al., 2002; Lee & Roßling, 2011; Malmi et al., 2014) and in computer science education in general (Machanick, 2007). For instance, Hundhausen et al. (2002) observed that the success of AV experiments is predicted by the theory of personal constructivism.

Constructivist theories state that people actively construct knowledge rather than passively receive and store ready-made knowledge. The pedagogy based on constructivism is student-centered, which places a greater emphasis on the learner's prior experience rather than the teacher's, and on the active construction of knowledge rather than the passive receipt of information. Constructivism has many interpretations; among them, Piaget's cognitive constructivism, Paper's constructionism and Vygotsky's social constructivism. The results obtained from different interpretations of constructivism are summarised in Figure 2.

Social constructivists instead emphasise the importance of the social and cultural nature of individuals' knowledge construction and tend to see knowledge as something that is defined through social collaboration and language use. In spite of Vygotsky's social constructivism, program visualizations are seen as sociocultural tools, as (Hidalgo-Céspedes et al., 2016) pointed out. Bandura (1999)'s social cognitive theory state that learning is affected by cognitive, behavioral and environmental factors. Furthermore, it identified several critical concepts that contribute to learning which are: human agency, self-regulation and self-efficacy. Self-efficacy is an important factor that refers to an individual's belief in their capabilities to successfully control actions or events. In Hall et al. (2013)'s which is based on social cognitive theory, the instruction that allows students to check their own progress at a designated level of proficiency has a positive impact on motivation.

On the other hand, Hundhausen et al. (2002) observed that the most effective algorithm visualization was built and guided by the cognitive constructivism theories. A cognitive constructivism theory predicted that the more effort put to engage students in an activity, the more robust learning could be achieved (Hundhausen et al., 2002; Malmi et al., 2014). In light of cognitive constructivism, the role of SV is not that of an artifact to transfer knowledge to the student, but rather that it should enable the student to construct the knowledge through active engagement (Hundhausen et al., 2002). Active learning is one of the

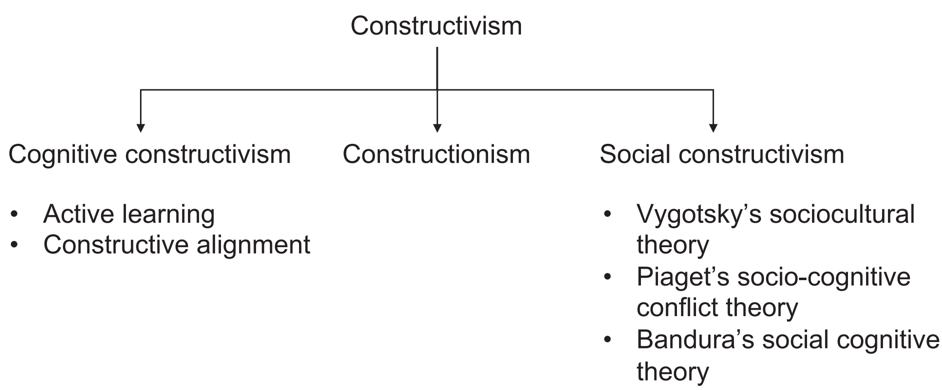

Figure 2. Summary of constructivism theories.

principles of cognitive constructivism where the learner actively constructs new understandings by becoming actively engaged with their activity (Urquiza-Fuentes & Velázquez-Iturbide, 2013). Therefore, the PV designer should consider different types of activities and engagement features to increase the learners' active engagement and enable them to construct knowledge (Hundhausen et al., 2002; Moreno et al., 2014). In Végh and Takáč (2017)'s study, the constructive learning theory was used to make a design decision to select interactive activities. On the other hand, several studies have used cognitivism to support their design decision for constructing an artifact or for the purposes of analysis (Hall et al., 2013; Moreno et al., 2014; Odisho et al., 2016). Instead, to get the largest benefit of SV for the students, they should be more actively engaged with the technology. In this respect, visualization is seen as a tool for constructing knowledge rather than a conveyor of knowledge.

The constructionism built based upon Piaget's constructivism theory shares the same view of learning as building knowledge through the progressive internalisation of actions. However, in constructionism, the focus shifts toward enabling the student to construct a public artifact to improve their learning involvement. In addition, the meaning of construct differs in both theories. In Piaget's theory, it refers to the cognitive construct of knowledge, while Paper used the term to refer to physical construct (e.g. building a sand castle) Paper's constructionism is implied in Cetin and Andrews-Larson (2016)'s study. The study follows the principles of this study to design a CV that leverages student engagement and improves learning outcomes.

The Cognitive Load Theory (CLT) is also important due to the complexity nature of learning programming languages for novices. The instructional design and educational technology design should both consider CLT during the design phase. Gordon and Guo (2015) noted that requiring novice learners to solve problems on their own too early places a strain on their working memory. In the same vein, Velázquez-Iturbide et al. (2016) argued that the design of instructional technology should avoid unnecessary cognitive load by using alternative instructional material, such as worked examples. By considering CLT, several design decisions have been taken to avoid placing an excessive cognitive load on students.

Variation theory takes a different view of learning compared with cognitive theories (Guo & Pang, 2011). However, the theory draws important principles that could guild the design of SV, which are: "critical aspects/features" and "patterns of variation and invariance". According to last principles, the student must experience a variation of phenomena or object. Four patterns of variation and invariance have been defined as following: contrast, separation, fusion, and generalization (Guo & Pang, 2011). The contrast pattern could be applied in SV by enabling the student to manipulate the impute values, and see how that affect the execution. In another hand, separation pattern could be achieved by providing the student with a different view of the execution process, for example, line-by-line view, memory changes, or object-oriented view. The remaining educational theories are SOLO taxonomy and Bloom's taxonomy. They are mainly used to analyze the student understanding levels. In sum, educational theories are important to design an effective artifact because how knowledge is absorbed, processed and retained during learning will be helpful to take important design decisions.

5.2.2. Psychology theories

We obtained seven theories from the psychological domain. Most of these theories are used to suggest a design technique or principles when constructing the artifact. Besides, in Velázquez-Iturbide et al. (2017) employed the theory in his study to explain the relationships between the constructs. The study focused on the impact of PV on student motivation based on the self-determination theory. This study explores the motivation as another factor that could contribute to the effectiveness of program visualization besides ET. Self-determination theory is a theory of motivation founded in 1985 by Deci and Ryan (1985). The theory constructs motivation into following dimensions: intrinsic motivation, extrinsic motivation and amotivation. In education, it is used to analyse student motivation in an educational context (Deci, Vallerand, Pelletier, & Ryan, 1991). Its main focus is to promote students' interest in learning, valuing of education and confidence in their own capacities and attributes (Deci et al., 1991). Several studies link self-determination motivation with positive learning outcomes, such as academic performance and likelihood to stay at school (Deci et al., 1991). It should be noted that there are different views on the relationship between engagement and motivation (Reschly & Christenson, 2012). Some scholars have used both terms interchangeably, as in Yohannis and Prabowo (2015) study, while others distinguish between the definition of the two terms, as in Velázquez-Iturbide et al. (2017) study. Yohannis and Prabowo (2015) study conceptualises the design framework for gamified algorithm learning by considering the Intrinsically Motivating Instruction by Malone (1980). Games were used to motivate the learner to learn algorithms by providing encouraging feedback and delivering challenges through establishing well-defined goals with proper difficulties. Further, to improve engagement, it is important to first increase the learners' motivation. According to Malone (1980), the main characteristics for computer-based learning environments, especially computer games, can be organised into three categories: challenge, fantasy and curiosity. This theory aims to increase learner motivation and engagement with games to learn more.

The Flow Theory has also attracted greater attention over the last few years with computer-based learning and e-learning systems to improve student engagement. In general, the theory describes how cognitive engagement happens when a person involves an activity, such as working, doing sports, or other leisure activities. According to (Csikszentmihalyi, 2014), a flow experience is a state in which people are fully involved in an activity affected by intrinsic motivation. It could begin if there is a fit between the skills of the self and the challenges afforded by the environment (Csikszentmihalyi, 2014). In performing a particular learning activity, the flow theory suggests that engagement is achieved when the given level of a challenge is equal to the individuals' levels of skill (Katuk et al., 2014). In educational technology research, this theory is used to design flow activities (or experience) and to understand the users' reactions and motivations towards the application (Csikszentmihalyi, 2014; O'Brien & Toms, 2008). For instance, in Mishra et al. (2014)'s study, the theory was used to design an artifact to maintain different levels of student engagement by supporting varying levels of activity. In the same vein, in Yohannis and Prabowo (2015)'s study, which was affected by the flow principles, the different levels of the tools should be carefully designed to ensure the flow of student. The system should avoid overly difficult or too easy materials to keep it challenging to students. Also, as the learners' skills grow, the challenge level in the system should also be grown.

In Hall et al. (2013)'s study, they drive their design of tutorials and activities from the theory of change in order to increase student engagement with the system. Furthermore, artifacts will help to keep students as active participants in the assessment and monitor their progress; the same goes for the instructors. In contrast, Buchanan and Laviola (2014) took a different approach to increase student engagement. His study focused on finding an appropriate interface for the PV system based on the affordance theory. Affordance theory is essential to designing an interface that makes it more evident to foster exploration and learning for the user (Buchanan & Laviola, 2014). The interface should match the task domain so they can increase the physical or communication activity pattern that facilitates the creation of a new mental model. It has been acknowledged that an inappropriate interface will increase the cognitive load on the student (Buchanan & Laviola, 2014). The remaining theories (i.e. Dual-coding theory) gives explicit prescriptions for constructing artifacts. One application of dual-coding theory in the domain, the system designer adding additional verbal channel (like a textual, or audio explanation) beside the animation (non-verbal) to improve the student knowledge acquisition.

5.2.3. CER theories

Naps taxonomy was the only TMF that originated from the CER domain in this study. This is consistent with the result in the Malmi et al. (2014)'s, which reported that Naps' taxonomy was among the most used TMFs. Naps et al. (2003) argued that visualizations have little educational value unless it supports the active engagement of a learner. This taxonomy hypothesised that using a higher level of engagement will improve learning outcome. The review found five studies that adopted Naps' taxonomy.

First, Banerjee et al. (2015) compared the learning outcome by using program visualization in response to different engagement levels (Naps Taxonomy); namely: "responding" and "viewing". It also showed a significant relationship between cognitive achievement as a dependent variable and the perception of learning when using a higher engagement level (Responding over Viewing) as an independent variable. In his study, he also set behavioral engagement as another dependent variable, which also showed the significant result with a higher level of engagement.

In Banerjee et al. (2013)'s article, we found a significant difference in the relative rate of correct solution of the procedural questions on the post-test when using a higher engagement level. Also, the study reported a difference in classroom behavioral engagement between the two groups. However, there was no significant difference in the post-test scores. The author reported these results due to factors other than engagement level, such as learner characteristics or challenge level.

Finally, Urquiza-Fuentes and Velázquez-Iturbide (2013) used engagement level as their independent variable (IV), where the dependent variable (DV) comprised knowledge acquisitions, drop-out rate and learner satisfaction. The study focused on no-viewing, viewing and the construction levels of ET. The study reported that there was an increase in learning acquisition with a viewing and construction level over a no-viewing approach. Over a long-term analysis, the study found a significant decrease in the drop-out rate for students.

Furthermore, in the following studies Suryana and Husin (2014); Urquiza-Fuentes and Velázquez-Iturbide (2012) conducted their experiment based on Naps' framework. Overall, ET is an important framework used to evaluate the effectiveness of software visualization system. However, the aforementioned studies reported several limitations to this framework. Urquiza-Fuentes and Velázquez-Iturbide (2012) emphasized that there are other factors that need to be considered when evaluating engagement more than engagement level. In the same vein, Banerjee et al. (2013); Urquiza-Fuentes and Velázquez-Iturbide (2013) identified other constructs that could correlate to effectiveness besides engagement level, such as learner characteristics, topic complexity and challenge level of questions.

ET plays an important role for future studies in studying and understanding the effectiveness of program visualization. Along with Hundhausen et al. (2002)'s study, the development of SV aims toward focusing on engagement and interactive visualization. In addition, many studies used Naps' taxonomy for taking decisions regarding tools and designs, and for analysing the effectiveness of these tools. However, the studies using ET showed mixed results. Some empirical studies found a significant relationship between engagement levels and learning outcome; however, these taxonomies are still yet to describe this phenomenon (Malmi et al., 2014). That could be attributed to this theory since the only focus is on one factor that influences student engagement, which then reflects on the pedagogical effectiveness of the system. Hence, this finding raises the following question as "to which extent could ET give insight about student engagement with SV?".

6. Design principles for engaging SV

The findings of this review provide insights into how to design an engaging system. Based on the reviewed theories, we can summarise design guidelines for developers to enhance student engagement when building SV or its activity. Below, we suggested several principles to enhance student engagement based on our results discussion:

(1) Using a higher engagement level or a mix of them. The engagement taxonomy is still essential in the domain; however, more factors should be considered. (Constructivism)
(2) Keep the cognitive load of activity as low as possible, especially when targeting a novice student. Also, consider the additional load on a student when using the SV because the student needs to learn how to use the tool. Worked-examples and the Parson problem are good examples of low cognitive load activities that can be implemented in SV. (CLT)
(3) Hide irrelevant information from the visualization process. Many processes and events can occur during code execution; however, a good design should only present the most relevant information to the student. Further, the information must match the objective or goals of the task or activity. In contrast, software visualization that targets professional and experienced developers could present a wider view and more information to the their user. (CLT)
(4) Keep the activity challenge level consistent with the students' current skills to keep them in the flow with the system. A very difficult or very easy activity will have a negative impact on student engagement. The SV system needs to maintain the challenge level carefully during the progress of a student. (Flow theory)
(5) The student becomes intrinsically motivated with an activity when it gives the student a sense of competence, autonomy, relatedness, and purpose. (Self-determination theory)
(6) Also, extrinsic motivation can help by linking the system with receiving rewards, such as grades or some gamification elements (e.g. leaderboards, badges, etc.). (Self-determination theory)

The effectiveness of SV is a debatable issue in the domain. It does not exclusively depend on the aesthetic of artifacts and related elements, such as color or animation. Instead, student engagement with clear pedagogical goals is very significant in term of effectiveness. Considering learning theories could also greatly contribute to that. The aforementioned principles extracted from the common theories used in the domain are a good starting point to improve the effectiveness of SV. In addition, these principles enhance student engagement, as previously stated when we discussed how the original theories were correlated with student engagement in the literature. Enhancing student engagement includes increasing the time the student spends using the tools and improving the quality of that time.

6.1. The engagement attribute

Different theories were used to put a lens on how visualization influences student engagement. Each has its advantages and/or limitations from different perspectives. In this section, we try to align between these theories and practices by extracting the engagement attribute from the existing TMFs. The engagement attribute could be defined as a characteristic of user-computer interaction that influences or is a component of engagement (O'Brien & Toms, 2008). This list of attributes will help the system designer to build or construct a new activity or feature in SV. O'Brien and Toms (2008) reported several attributes based on his extensive study. The study considers the following theories to build the attribute list: flow theory, aesthetic theory, play theory and information theory. In the light of O'Brien and Toms (2008)'s engagement attributes and aforementioned theories, we have created a list of engagement attributes, as provided below:

Aesthetics: refers to visual appearance or the study of natural and pleasing aspects of the system. There are many elements relating to aesthetic that could be taken into consideration with SV, such as text, graphics and animation appearance. The visual appearance of these elements includes font, colors, mixing graphics with text, the design of the diagram and so on. A good design emphasises aesthetics over the amount of information displayed on the screen to increase the readability and aesthetic sense of the end user. Furthermore, giving the user the ability to customise the interface element is a good practice in the domain.

Affective appeal: is the emotional reaction to a system that encourages a love of learning, such as: attitudes, interests, relationships and values. Affect refers to a user's emotional investment, how immersed in the system they are and how they sustain their involvement in the environment.

Challenge: The challenge level of a task has to be adapted to the student's level to keep them engaged. An optimal engagement could be achieved when the challenge level of an activity matches the individual's level of skills. The failure to match those two levels could lead to either boredom or anxiety. Student skills should be mounted during the time as the student's skills will increase within the course.

Immediate feedback: Feedback is critical to maintaining student engagement as it is a base for future action. Further, it is the immediate response or reaction from the system or system user (i.e. teacher, or peer) to a current action in terms of evaluating a solution and showing how to demonstrate the progress, etc. The feedback should be constructive and informative in solving students' difficulties.

Intrinsic motivation: The student becomes intrinsically motivated when learning tools create a scenario that provides competence, autonomy, relatedness and purpose. Intrinsic motivation refers to the performance of an activity for no apparent reinforcement other than the process of performing the activity. It is the highest form of motivation that is characterised by interest, enjoyment, inherent satisfaction and self-regulated conception with the activity.

Extrinsic motivation: Extrinsic motivation is defined as the performance of an activity because it is perceived to be instrumental in achieving valued outcomes that are distinct from the activity itself. Extrinsic motivation is induced by external stimuli, such as additional points for an exam.

Perceived control: is to which extent the student had the opportunity to decide how to proceed through the system. This sense of control keeps student engaged and decides their learning path in the system. The students' ability to control over their learning by choosing the topic, activity, assessment and even the time increases the students' sense of control.

Low cognitive load: Ensuring an activity's cognitive load is at a lower state and avoiding overload student working memory early, which will cause students to leave the system. A reduced load in working memory is important as working memory is limited in both capacity and its decay period. According to CLT, asking novice students to solve programming problems is too early; instead, they can learn better from lower load activities.

Social interaction: As in social constructivism, social interaction by allowing students to interact with each other will facilitate their learning. In addition, social interaction will increase the student's sense of belonging and connection, which reflects on their engagement. In addition, it is associated with behavior and emotional engagement. Social interaction can occur in term of collaboration, negotiation and discussing their ideas with other peers and teachers. A system should facilitate such interaction between the students and provide additional social channels between different peers.

7. Conclusions

In this paper, the researchers presented a systematic literature review on student engagement in software visualization from a theoretical perspective. The study relied on queries from six databases and search engines which have revealed a total of 432 articles. Of them, 58 were selected for the final extraction to answer the research questions. However, this study found only 18 articles that explicitly mentioned the theoretical framework used in their study. These TMFs were recorded and discussed in more detail in the article. Around half of these TMFs originated from educational research, while only 20% of these TMFs originated from the CER discipline, which indicates that the field needs more research and enhancement from a theoretical perspective. Namely, the Naps Taxonomy for engagement was the only TMF to originated from the CER discipline, which used to design and evaluate the software visualization.

The results of this study show a richness in the theoretical framework obtained from different disciplines. However, it can be concluded that most of these TMFs were conceived as user-centered designs. In educational technologies, constructivism is a dominant part of the instructional and tools design. Despite the need for contemporary frameworks in CER on how and why to use technology in education practice, a new framework or theory failed to emerge. The evidence from this study implies the need for new TMFs to guide the educators and tools designers as to how to align the educational goals with academic and pedagogical output when using technology. Despite Malmi et al. (2014) advocates the need for more TMFs to emerge from CER, however, it is disappointing that the CER research did not reveal additional theories to describe an important topic like engagement and how it influence users. As engagement is a complex topic and it needs to insight to understand how technology can influence the engagement of users. There is a demand in the CER field to have it is own TMF to design, analyse and predict engagement.

Despite its exploratory nature, this study offers some guidelines on how to design an engaging software visualization. From this study, the researchers suggest several engagement attributes in order to maintain engagement while designing SV. This paper was the researchers' first step towards enhancing our understanding of how to increase student engagement in SV tools and how technology influences student engagement. As the role of engagement with SV is still questionable, further studies will need to be undertaken to give us insight into engagement with technology while considering different aspects and perspectives of engagement also is important. Lastly, additional work is required to originate design frameworks and models to give us greater insight into how technology influences student engagement.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the Universiti Utara Malaysia and Ministry of Higher Education of Malaysia under Fundamental Research Grant Scheme (FRGS) [S/O code:13581].

Notes on contributors

Abdullah Al-Sakkaf is a doctoral student at the School of Computing, College of Arts and Sciences, Universiti Utara Malaysia. He has an MSc in Information Technology from UUM in 2017, and a Bachelor of Computer Information System from Al-Ahgaf University in Yemen. His research focuses on computer science education and teaching programming.

Mazni Omar is a senior lecturer at the School of Computing, College of Arts and Sciences, Universiti Utara Malaysia. She received the BSc. degree (with honors) in information technology from Universiti Utara Malaysia, in 2000, the MSc. degree in software engineering from Universiti Teknologi Malaysia, in 2002, and the Ph.D. degree in information technology and quantitative sciences from Universiti Teknologi MARA, Malaysia, in 2012. Her current research interests include empirical software engineering, data mining and knowledge management.

Mazida Ahmad is an Associate Professor at the School of Computing, College of Arts and Sciences, Universiti Utara Malaysia. She received the BMIS degree from International Islamic University of Malaysia, in 2001, the MSc. degree in Software Engineering from Universiti Teknologi Malaysia, in 2003, and the Ph.D. degree in Knowledge Management from Universiti Sains Malaysia, in 2010. Her current research interests include knowledge management, information system development and software engineering education.

ORCID

Abdullah Al-Sakkaf http://orcid.org/0000-0002-9586-9491
Mazni Omar http://orcid.org/0000-0003-1816-2940
Mazida Ahmad http://orcid.org/0000-0001-9536-5042

References

Suryana, N., & Husin, B. (2014). Effectiveness of integrated algorithm-program visualization: A case study with the 3De-AlProV. *Advanced Science Letters*, *20*(1), 304–308.

Bandura, A. (1999). Social cognitive theory: An agentic perspective. *Asian Journal of Social Psychology*, *2*(1), 21–41.

Banerjee, G., Murthy, S., & Iyer, S. (2013). Program visualization: Effect of viewing vs. responding on student learning. In L.-H. Wong (Ed.), *Proceedings of the 21st international conference on computers in education*.

Banerjee, G., Murthy, S., & Iyer, S. (2015). Effect of active learning using program visualization in technology-constrained college classrooms. *Research and Practice in Technology Enhanced Learning*, *10*(1), 15.

Ben-Ari, M., Bednarik, R., Levy, R.-B.-B., Ebel, G., Moreno, A., Myller, N., & Sutinen, E. (2011). A decade of research and development on program animation: The jeliot experience. *Journal of Visual Languages & Computing*, *22*(5), 375–384.

Bikner-Ahsbahs, A., & Prediger, S. (2010). Networking of theories—An approach for exploiting the diversity of theoretical approaches. In B. Sriraman & L. English (Eds.), *Theories of mathematics education: Seeking new frontiers* (pp. 483–506). Berlin: Springer.

Buchanan, S., & Laviola, J. J., Jr. (2014). CSTutor: A sketch-based tool for visualizing data structures. *Transaction on Computer Education*, *14*(1), 3:1–3:28.

Cetin, I., & Andrews-Larson, C. (2016). Learning sorting algorithms through visualization construction. *Computer Science Education*, *26*(1), 27–43.

Cooper, M. L., Shaffer, C. A., Edwards, S. H., & Ponce, S. P. (2014). Open source software and the algorithm visualization community. *Science of Computer Programming*, *88*, 82–91.

Csikszentmihalyi, M. (2014). *Applications of flow in human development and education*. Dordrecht: Springer.

Deci, E., & Ryan, R. (1985). *Intrinsic motivation and self-determination in human behavior*. New York, NY: Plenum Press..

Deci, E., Vallerand, R., Pelletier, L., & Ryan, R. (1991). Motivation and education: The self-determination perspective. *Educational Psychologist*, *26*(3–4), 325–346.

Ericson, B. J., Guzdial, M., & Morrison, B. B. (2015). Analysis of interactive features designed to enhance learning in an ebook. *Proceedings of the 11th annual international conference on international computing education research* (pp. 169–178). New York, NY: ACM.

Fincher, S., & Petre, M. (2004). *Computer science education research*. London, UK: Taylor & Francis.

Fouh, E., Akbar, M., & Shaffer, C. (2012). The role of visualization in computer science education. *Computers in the Schools*, *29*(1–2), 95–117.

Fouh, E., Karavirta, V., Breakiron, D. A., Hamouda, S., Hall, S., Naps, T., & Shaffer, C. (2014). Design and architecture of an interactive eTextbook - The OpenDSA system. *Science of Computer Programming*, *88*, 22–40.

Fredricks, J., Blumenfeld, P. C., & Paris, A. H. (2004). School engagement: Potential of the concept, state of the evidence. *Review of Educational Research*, *74*(1), 59–109.

Fredricks, J., Filsecker, M., & Lawson, M. A. (2016). Student engagement, context, and adjustment: Addressing definitional, measurement, and methodological issues. *Learning and Instruction*, *43*, 1–4.

Gordon, M., & Guo, P. J. (2015, December). Codepourri: Creating visual coding tutorials using a volunteer crowd of learners. *2015 IEEE symposium on visual languages and human-centric computing (VL/HCC)* (Vol. 2015, pp. 13–21). Atlanta, GA: IEEE.

Gregor, S. (2006). The nature of theory in information systems. *MIS Quarterly*, *30*(3), 611–642.

Gunuc, S., & Kuzu, A. (2015). Confirmation of campus-class-technology model in student engagement: A path analysis. *Computers in Human Behavior*, *48*, 114–125.

Guo, J.-P., & Pang, M. F. (2011, December 1). Learning a mathematical concept from comparing examples: The importance of variation and prior knowledge. *European Journal of Psychology of Education*, *26*(4), 495–525.

Hall, S., Fouh, E., Breakiron, D., Elshehaly, M., & Shaffer, C. (2013). Evaluating online tutorials for data structures and algorithms courses. *Proceedings of the 2013 ASEE annual conference & exposition*. Atlanta, GA: American Society for Engineering Education.

Hannay, J. E., Sjoberg, D. I. K., & Dyba, T. (2007). A systematic review of theory use in software engineering experiments. *IEEE Transactions on Software Engineering*, *33*(2), 87–107.

Hidalgo-Céspedes, J., Marín-Raventós, G., & Lara-Villagrán, V. (2016). Learning principles in program visualizations: A systematic literature review. *2016 IEEE frontiers in education conference (FIE)* (pp. 1–9). Erie, PA: IEEE.

Hosseini, R., Sirkiä, T., Guerra, J., Brusilovsky, P., & Malmi, L. (2016). Animated examples as practice content in a java programming course. *Proceedings of the 47th ACM technical symposium on computing science education* (pp. 540–545). New York, NY: ACM.

Hundhausen, C. D., Douglas, S. A., & Stasko, J. T. (2002). A meta-study of algorithm visualization effectiveness. *Journal of Visual Languages & Computing*, *13*(3), 259–290.

Isohanni, E., & Järvinen, H.-M. (2014). Are visualization tools used in programming education?: By whom, how, why, and why not? *Proceedings of the 14th koli calling international conference on computing education research* (pp. 35–40). New York, NY: ACM.

Isohanni, E., & Knobelsdorf, M. (2011). Students' long-term engagement with the visualization tool VIP. *Proceedings of the 11th koli calling international conference on computing education research - koli calling '11* (pp. 33). New York, NY: ACM Press.

Kahu, E. R. (2013). Framing student engagement in higher education. *Studies in Higher Education*, *38*(5), 758–773.

Katuk, N., Omar, M., & Halim, N. A. (2014). Improving engagement in hypermedia learning: A design for navigation support. *Proceedings of the 16th international conference on information integration and web-based applications & services* (pp. 40–43). New York, NY: ACM.

Kitchenham, B., Budgen, D., & Brereton, P. (2015). *Evidence-based software engineering and systematic reviews*. New York, NY: Taylor & Francis Group.

Knobelsdorf, M., Isohanni, E., & Tenenberg, J. (2012). The reasons might be different: Why students and teachers do not use visualization tools. *Proceedings of the 12th koli calling international conference on computing education research* (pp. 1–10). New York, NY: ACM.

Krause, K., & Coates, H. (2008). Students' engagement in first-year university. *Assessment & Evaluation in Higher Education*, *33*(5), 493–505.

Kuh, G. D. (2009). The national survey of student engagement: Conceptual and empirical foundations. *New Directions for Institutional Research*, *2009*(141), 5–20.

Lee, M.-H., & Roßling, G. (2011). Constructivist and constructionist approaches to constructing algorithm visualizations: A proposal. *2011 IEEE 11th international conference on advanced learning technologies* (pp. 171–173). Athens, GA: IEEE.

Lishinski, A., Good, J., Sands, P., & Yadav, A. (2016). Methodological rigor and theoret- ical foundations of cs education research. *Proceedings of the 2016 ACM conference on international computing education research* (pp. 161–169). New York, NY: ACM.

Machanick, P. (2007). A social construction approach to computer science education. *Computer Science Education*, *17*(1), 1–20.

Malmi, L. (2014). Theory — What is it for? *ACM Inroads*, *5*(4), 34–35.

Malmi, L., Sheard, J., Simon, B. R., Helminen, J., Kinnunen, P., & Taherkhani, A. (2014). Theoretical underpinnings of computing education research: What is the evi- dence? *Proceedings of the 10th annual conference on international computing education research* (pp. 27–34). New York, NY: ACM.

Malmi, L., Sheard, J., Simon, B. R., Helminen, J., Korhonen, A., & Taherkhani, A. (2010). Characterizing research in computing education: A preliminary analysis of the literature. *Proceedings of the 6th international workshop on computing education research* (pp. 3–12). New York, NY: ACM.

Malone, T. W. (1980). What makes things fun to learn? Heuristics for designing instructional computer games. *Proceedings of the 3rd ACM sigsmall symposium and the first SIGPC symposium on small systems* (pp. 162–169). New York, NY: ACM.

Mishra, S., Balan, S., Iyer, S., & Murthy, S. (2014). Effect of a 2-week scratch intervention in CS1 on learners with varying prior knowledge. *Proceedings of the 2014 conference on innovation & technology in computer science education - iticse '14* (pp. 45–50). New York, NY: ACM Press.

Moreno, A., Sutinen, E., & Joy, M. (2014). Defining and evaluating conflictive animations for programming education: The case of Jeliot ConAn. *Proceedings of the 45th ACM technical symposium on computer science education* (pp. 629–634). New York, NY: ACM.

Myller, N., Bednarik, R., Sutinen, E., & Ben-Ari, M. (2009). Extending the engagement taxonomy: Software visualization and collaborative learning. *Transaction on Computer Education*, *9*(1), 7:1–7:27.

Naps, T., Rodger, S., Velázquez-Iturbide, Á., Rößling, G., Almstrum, V., Dann, W., … McNally, M. (2003, June). Exploring the role of visualization and engagement in computer science education. *ACM SIGCSE Bulletin*, *35*(2), 131–152.

O'Brien, H. L., & Toms, E. G. (2008). What is user engagement? A conceptual framework for defining user engagement with technology. *Journal of the American Society for Information Science and Technology*, *59*(6), 938–955.

Odisho, O., Aziz, M., & Giacaman, N. (2016). Teaching and learning data structure concepts via visual kinesthetic pseudocode with the aid of a constructively aligned app. *Computer Applications in Engineering Education*, *24*(6), 926–933.

Pechenkina, E., Laurence, D., Oates, G., Eldridge, D., & Hunter, D. (2017, April). Using a gamified mobile app to increase student engagement, retention and academic achievement. *International Journal of Educational Technology in Higher Education*, *14*(1), 31.

Price, B. A., Baecker, R. M., & Small, I. S. (1993). A principled taxonomy of software visualization. *Journal of Visual Languages & Computing*, *4*(3), 211–266.

Reschly, A. L., & Christenson, S. L. (2012). Jingle, jangle, and conceptual haziness: Evolution and future directions of the engagement construct. In S. L. Christenson, A. L. Reschly, & C. Wylie (Eds.), *Handbook of research on student engagement* (pp. 3–19). Boston, MA: Springer US.

Schindler, L. A., Burkholder, G. J., Morad, O. A., & Marsh, C. (2017). Computer-based technology and student engagement: A critical review of the literature. *International Journal of Educational Technology in Higher Education*, *14*(1), 1–28.

Shaffer, C., Akbar, M., Alon, A. J., Stewart, M., & Edwards, S. (2011). Getting algorithm visualizations into the classroom. *Proceedings of the 42nd ACM technical symposium on computer science education - SIGCSE '11* (pp. 129–134). New York, NY: ACM.

Shaffer, C., Cooper, M. L., Alon, A. J. D., Akbar, M., Stewart, M., Ponce, S., & Edwards, H. (2010). Algorithm visualization: The state of the field. *ACM Transactions on Computing Education*, *10*(3), 1–22.

Shi, N., Min, Z., & Zhang, P. (2017). Effects of visualizing roles of variables with animation and IDE in novice program construction. *Telematics and Informatics*, *34*(5), 743–754.

Shulman, L. S. (2002). Making differences: A table of learning. *Change: The Magazine of Higher Learning*, *34*(6), 36–44.

Sirkiä, T., & Sorva, J. (2015). How do students use program visualizations within an interactive Ebook? *Proceedings of the 11th annual international conference on international computing education research* (pp. 179–188). New York, NY: ACM.

Sorva, J., Karavirta, V., & Malmi, L. (2013). ACM transactions on computing education. *A Review of Generic Program Visualization Systems for Introductory Programming Education*, *13*(4), 15.1–15.64.

Urquiza-Fuentes, J., & Velázquez-Iturbide, Á. (2009). A survey of successful evaluations of program visualization and algorithm animation systems. *Transaction on Computer Education*, *9*(2), 9:1–9:21.

Urquiza-Fuentes, J., & Velázquez-Iturbide, Á. (2012). Comparing the effectiveness of different educational uses of program animations. *Proceedings of the 17th ACM annual conference on innovation and technology in computer science education - ITICSE '12* (pp. 174). New York, NY: ACM Press.

Urquiza-Fuentes, J., & Velázquez-Iturbide, Á. (2013). Toward the effective use of educational program animations: The roles of student's engagement and topic complexity. *Computers & Education*, *67*, 178–192.

Végh, L., & Takáč, O. (2017). Using interactive card animations for understanding of the essential aspects of non-recursive sorting algorithms. In J. Janech, J. Kostolny, & Gratkowski (Eds.), *Proceedings of the 2015 federated conference on software development and object technologies* (pp. 336–347). Cham, Switzerland: Springer International Publishing.

Velázquez-Iturbide, Á., Hernán-Losada, I., & Paredes-Velasco, M. (2017). Evaluating the effect of program visualization on student motivation. *IEEE Transactions on Education*, *60*(3), 238–245.

Velázquez-Iturbide, Á., Hernán-Losada, I., & Pérez-Carrasco, A. E. I. I. (2016). A “multiple executions” technique of visualization. *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE ’16* (pp. 59–64). New York, NY: ACM.

Velázquez-Iturbide, Á., Pérez-Carrasco, A., & Debdi, O. (2015). Experiences in usability evaluation of educational programming tools. In *Stem education: Concepts, methodologies, tools, and applications* (pp. 461–480). Hershey, PE: IGI Global.

Vihavainen, A., Airaksinen, J., & Watson, C. (2014). A systematic review of approaches for teaching introductory programming and their influence on success. *Proceedings of the 10th annual conference on international computing education research* (pp. 19–26). New York, NY: ACM.

Yohannis, A., & Prabowo, Y. (2015). Sort attack: Visualization and gamification of sorting algorithm learning. *2015 7th international conference on games and virtual worlds for serious applications (vs-games),* Skovde, Sweden (pp. 1–8). IEEE.