EUROGRAPHICS Symposium on Sketch-Based Interfaces and Modeling (2012)
L. B. Kara and K. Singh (Editors)

SetPad: A Sketch-Based Tool For Exploring Discrete Math Set Problems

Travis J. Cossairt and Joseph J. LaViola Jr.[1]

[1]University of Central Florida, Department of EECS, Orlando, FL, USA

Abstract

We present SetPad, a new application prototype that lets computer science students explore discrete math problems by sketching set expressions using pen-based input. Students can manipulate the expressions interactively with the tool via pen or multi-touch interface. Likewise, discrete mathematics instructors can use SetPad to display and work through set problems via a projector to better demonstrate the solutions to the students. We discuss the implementation and feature set of the application, as well as results from a formal user study measuring the effectiveness of the tool for students solving set proof problems. The results indicate that SetPad allows for efficient solutions to proof problems, and has the potential to have a positive impact when used as an individual student application or as an instructional tool.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [User Interfaces]: Interaction styles-User Interfaces—Graphical user interfaces

1. Introduction

As part of a computer science curriculum, it is important that students gain an understanding of core concepts such as discrete mathematics and set theory. Traditionally, an instructor uses a whiteboard to work out set theory problems while students solve similar problems on their own using traditional pen and paper. However, these methods can be limiting because they cannot provide instant feedback while the student or instructor explores a problem and can lead to errors in logic and misunderstanding of the laws of set algebra. Furthermore, traditional methods are limited in workspace and suffer from handwriting legibility issues.

To overcome these limitations, we have developed SetPad. By limiting manipulations to follow the laws of set algebra, we posit that students will be more likely to explore set proof problems and thus learn why and how expressions can be manipulated to solve the proofs. Additionally, we reinforce the names of the algebraic laws by displaying the applicable laws as the user moves the terms around in an expression. As such, SetPad could replace traditional pen and paper when used by students for exploring and solving set problems.

Additionally, SetPad has strong potential as an instructional tool when combined with a projector. A discrete mathematics instructor can use SetPad instead of the typical dry-erase whiteboard or overhead to show students how to solve set problems. By doing so, instructors gain nearly unlimited workspace, have guaranteed legible expressions, and have each step displaying the applied set algebra law automatically. SetPad also provides a simple interface for instructors to easily create new set proof problems. By starting out with any basic expression, they can use the tool to transform it to

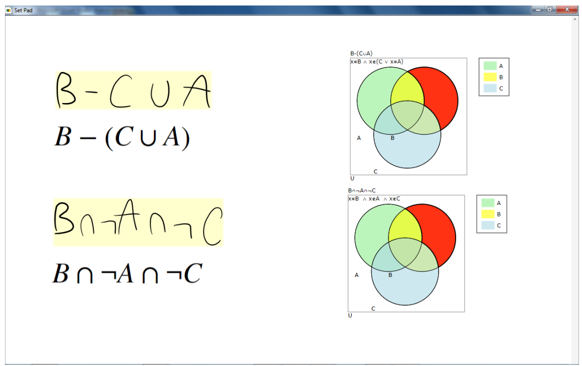

Figure 1: *Using SetPad to compare two equivalent set expressions.*

© The Eurographics Association 2012.

find equivalent mathematical relationships with derived expressions for the purposes of generating a step-by-step proof problem.

Lastly, SetPad also assists instructors when their students use the tool in the classroom. Student solutions to a set problem can be stored, aggregated, and distributed to instructors in real time. This aggregation is presented to instructors with a special tree visualization for use in observation and teaching analysis. We believe this ability holds great potential for giving instructors information about where students may be struggling with a particular set problem or in applying a particular set algebra law.

2. Related Work

There are several areas of work using pen and touch interfaces to create a digital workspace for exploring mathematics concepts. Hands-On Math [ZBAK10] is a recent effort that uses a digital workspace (in this case a Microsoft Surface) to allow freeform sketching and visualizing of simple algebraic expressions. Thimbleby and Thimbleby developed a simple calculator that combines stylus and touch input for solving simple algebra problems [TT05]. Like SetPad, these systems allow direct touch manipulation to transform expressions in an effort to explore the solution. SetPad differs in its gestures to manipulate the expressions, as well as its focus on set visualization through Venn diagrams and set algebra.

Stapleton, Delaney, Rodgers and Plimmer have explored recognizing user sketched Euler and Venn diagrams [DPSR10] [SDRP11] and have built SketchSet [WPS*11] using those findings. Their work is similar in that it can visualize Venn diagram problems, but differs in that SetPad focuses on sketching set theory algebra with our diagrams used solely for the visualization of that algebra.

MathPad^2 [LZ04] also presents a pen and gesture-based UI for sketching mathematical expressions. It features the ability to visualize and manipulate the graphical representation of the mathematics. VectorPad [BL10] lets users write down vector mathematics and presents animations illustrating different vector operations. Although these systems are similar to SetPad, they do not support set expression manipulation or visualization or try to provide an understanding of the application of mathematical laws (i.e., set algebra laws in SetPad). Other systems have been developed for entering mathematics using a pen-based interface but they also are not focused on set expression entry and manipulation or designed to assist in discrete mathematics instruction [LLM*08, Mic11, ZMLL08].

Commercially, AlgebraTouch [Sof11] is a touch-based math application available on the App Store for the iPhone. Instead of starting with an open workspace, AlgebraTouch attempts to teach basic algebra by working problem by problem with instructions on how to manipulate the expression using touch gestures. Although it does not use sets in any of its problems and only works with basic algebraic laws like commutativity, the animation and gestures it presents are visually appealing and intuitive.

SetPad also differs from these works in that in combines the use of a pen-based sketch system together with multi-touch capabilities. Combining these two approaches into one system was explored by Hinckley, Pahud, and Buxton [HPB10] where they advocated a "division of labor between pen and touch where the pen writes, touch manipulates, and the combination of pen+touch yields new tools." SetPad follows these principles by using the pen mostly for writing new expressions, and focusing on using the multi-touch interface to manipulate the expressions or for workspace maneuvering. SetPad did not however combine both systems together for any gestures for the sake of simplicity, but it certainly could be pursued in future work.

With regard to other high-level logic software, both Tarski's World [BE93] and Fitch [BE02] were developed to let students work with first order logic and proofs. Tarski's World provides an environment where students create a 3D graphical representation of a chess-like world, and then must describe that world using first order logic sentences. Fitch is a tool used to enter and build formal proofs that, like SetPad, gives instant feedback to the user when working through each step of the proof and marks steps that do not check out properly. While similar, neither of these applications focuses on solving and manipulating set proof problems like SetPad.

In terms of software tools specifically for assisting students and instructors with discrete mathematics, there has been little reported work in the literature [PWB11]. One of the few examples was a software system coupled with Hall and O'Donnell's discrete math textbook [HO00]. This software was text-based and similar to a computer algebra system, which is in direct contrast to SetPad's pen-and-paper style interface. To the best of our knowledge, SetPad is the first application that supports discrete mathematics instruction, in particular set theory, using pen and multi-touch input.

3. User Interface

SetPad is built off the starPad SDK 1.3 framework [Mil09], built by the Microsoft Center for Research on Pen-Centric Computing at Brown University. StarPad is a general-purpose framework for Windows written in the C# Language and .NET Platform 3.5, using the Windows Presentation Foundation (WPF) 3.5 framework. This framework was chosen for its robustness in recognizing and typesetting handwritten mathematical expressions and symbols, as well as its modification flexibility to allow further recognitions and manipulation. SetPad itself is a C# WPF application that runs on any Windows XP or Windows 7-based device that also has pen and touch support.

In SetPad, users start with a blank open workspace where

© The Eurographics Association 2012.

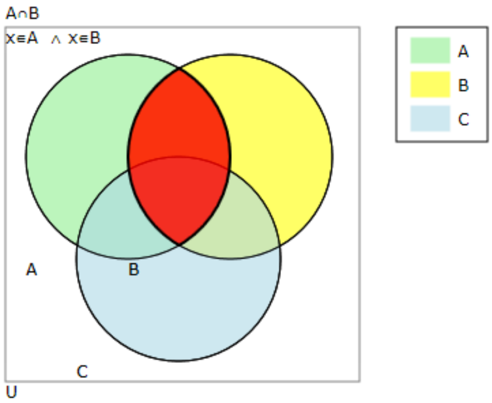

Figure 2: *Venn diagram visualization for a selected set expression.*

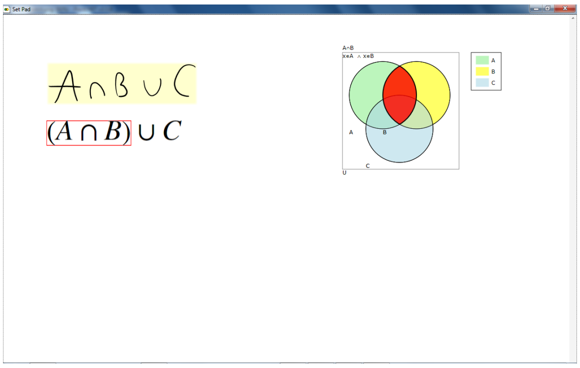

Figure 3: *Sketching an expression and selecting a sub-expression that SetPad visualizes in red.*

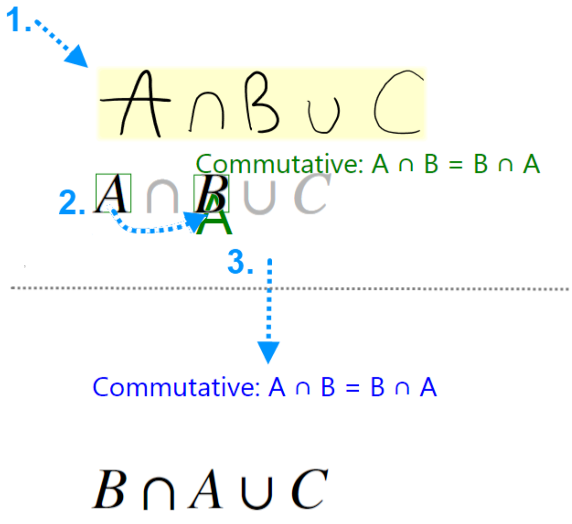

Figure 4: *Interactively manipulating an expression using multi-touch and pen input. (1. A set expression is sketched. 2. A user touches set A, and drags it without releasing to set B, causing a potential Commutative transformation to appear. 3. A user releases the touch, causing the Commutative transformation to be applied and the expression to be changed and redrawn below.)*

they can begin to sketch an expression in either set notation (e.g., "$A \cap B$") or element of notation (e.g., "$x \in A$"). As they sketch out expressions, the tool renders a Venn diagram graphic in real-time displaying the relationship between the sets described by the expression (See Figure 1).

SetPad also displays information in the rendered graphic area using the original expression notation they have sketched as well as alternate notation (i.e., set notation as well as "x element of" notation) (See Figure 2). This alternate notation provides additional information to aid in understanding what the expression is describing.

Additionally, after the expression is recognized by the starPad framework, users can choose to tap on any part of the expression to see just that sub-expression displayed in the rendering area (See Figure 3).

Once a student or instructor has completed sketching the initial expression, they can sketch below that recognized expression what they think the next step would be in a proof. By using the Venn diagram visualization, they can verify equivalence between the two expressions (as shown in Figure 1). Alternatively, they can use the interactive algebraic law manipulation feature described in Section 3.1.

3.1. Interactive Algebraic Law Manipulation

One of the key features of SetPad is that students and instructors can explore set expressions and set algebra, without fearing that they have violated any set algebra laws. By dragging expressions around with the pen or finger, the tool displays what valid operations are allowed, alongside the law that applies to make that operation valid. By displaying the law that would apply, along with the input expression and resulting expression, we believe SetPad reinforces to the student when certain transformations can take place and why, even if the operation is not the correct next step or the one they eventually decide to use.

If the user lifts the pen or touch for a valid move, SetPad applies that law to the expression and moves on to the next step below in the workspace while also showing what law was applied to get the expression to the new form (See Figure 4). The user can then proceed to manipulate the new active expression, or choose to sketch the next step using the pen.

Based on feedback given during initial pilot testing, the interactive gestures were designed to be very simple using

© The Eurographics Association 2012.

just one touch or pen down, and then typically dragging one operand into another. Additionally, pilot test participants requested that gestures work bi-directionally, that is, that they could either drag operand 1 into operand 2 or operand 2 into operand 1 with the same results.

Currently, SetPad recognizes and implements the following laws of set algebra:

3.1.1. Commutative Laws

A user can initiate the law of commutativity by dragging one expression or set into another within the same parent expression. If the law is applied, the operands switch positions with each other.

3.1.2. Distributive Laws

A user can initiate the distributive law by dragging an expression over the operator of the expression it is to be distributed over. If the law is applied, the operand is distributed in the resulting two sub expressions.

A reverse distribution can be performed by the user dragging a common factor out from either sub expression back into the distributing operator. When applied, the expression is transformed back to the original simplified expression. Originally, we enforced that the common factors had to be the first terms in both sub expressions (as explicitly written in the law) but discovered that pilot test user's found the extra step of moving via commutative unnecessary and tedious.

3.1.3. Associative Laws

A user can initiate the law of associativity to remove parentheses in an expression by dragging any parenthesis into any operand operator within that associative parent expression. Because a single parenthesis can sometimes be difficult for user's to touch and drag due to its small size, we also support dragging the entire grouped expression into the parent's operator.

If the law is applied, all associative parentheses are removed from the expression. Initially we experimented with dragging a specific parenthesis around to expand or contract a grouping, but we observed from pilot testing that user's were generally confused by this method and preferred to just quickly remove the entire associative grouping altogether to solve the proofs.

3.1.4. Complement Laws

A user can simplify a set that is unioned or intersected with its complement by dragging one into the other. When applied, it will transform the expression into either the empty or universal set depending on which operator joined the two.

3.1.5. Identity Laws

Identities can be simplified by the user dragging the operand into the identity, or the identity into the operand. Identities recognized are intersections into the universal set and unions with the empty set.

3.1.6. Idempotent Laws

Idempotents can be simplified by the user dragging the operand into the matching second operand within the same sub expression. Both union and intersection idempotents are recognized.

3.1.7. Domination Laws

Dominations can be simplified by the user dragging the operand into the dominating value within the same sub expression, or the dominating value into the operand. SetPad recognizes both intersections dominated by the empty set and unions dominated by the universal set.

3.1.8. Absorption Laws

Absorption can be performed by dragging the set operand into the adjacent expression containing that same set. When applied, the expression is simplified to become just the single set. Based on user request, we also added support to drag the set operand into the second matching set within the adjacent expression to apply the law.

3.1.9. De Morgan's Laws

De Morgan's Laws can be initiated by dragging a complement over the operator of the expression it is to be distributed over. If the law is applied, the complement is distributed in the resulting two sub expressions.

Inversely, a reverse De Morgan transformation can be performed by the user dragging the common complement out from either sub expression back into the distributing operator. When applied, the expression is transformed back to the original simplified complementing expression.

3.1.10. Involution Law (Double Negation)

The Involution Law (Double Negation) is enacted when user drags the one complement into the neighboring complement. When applied, the complements cancel each other out leaving the resulting expression.

3.1.11. Relative Complements

Users can swap between the two definitions of relative complement (i.e., $A - B$ vs. $A \cap \overline{B}$) by dragging the first operand into the second operand in the relative complement expression. Only after the expression is in the intersection form can the other laws of algebra be applied.

3.2. Expanding Simple Expressions

Because a proof problem can be solved by starting out with either the left or right hand side of the equation, we believe it is important to be able to expand out a simple expression via identity and other rules. For example, a simple set A is equivalent to $A \cup \emptyset$ via the law of set identities. To support user expansion of the set, we have created a radial menu

© The Eurographics Association 2012.

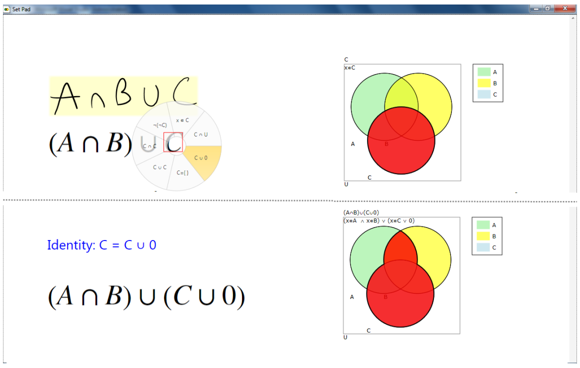

Figure 5: *Expanding expression by double-tapping set C to bring up radial menu and selecting $C \cup \emptyset$.*

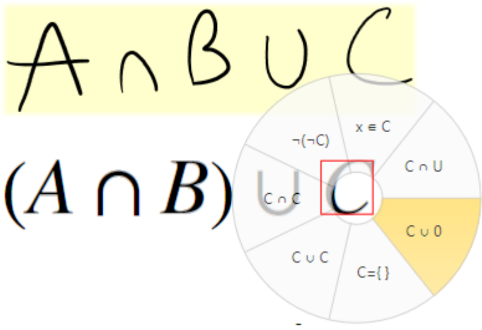

Figure 6: *Radial menu options to expand set C.*

invoked by double-tapping any set or sub-expression. This action brings up a context sensitive radial menu that when an item is selected allows expanding the current expression based on laws of identity, idempotent, and involution (double negation) as shown in Figure 6.

3.3. Defining Subsets

Users of SetPad are also capable of defining that one set is a subset of another. To do so, a user simply needs to sketch the expression showing that the two sets are related via the subset of symbol. After doing this, SetPad changes how that set is rendered in the Venn Diagram when in relation to the set that it is a subset of as shown in Figure 7. Moreover, once a set is defined as a subset of another, SetPad allows the user to perform several transformations to be made allowing simplification of expressions that are necessary for certain proof problems.

3.4. X Element Of Notation

During early evaluations, several students suggested that SetPad should support the current standard set notation as well as Set Builder notation (i.e., "x is an element of").

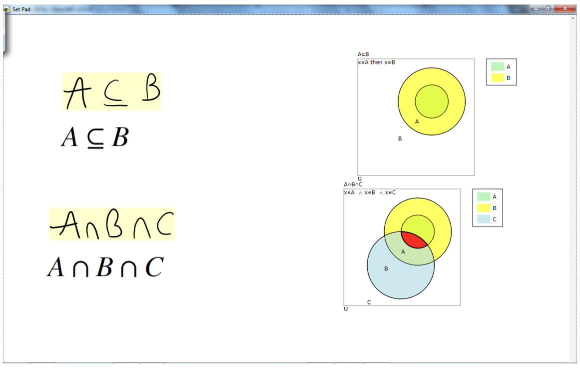

Figure 7: *Defining A as a subset of B with resulting changes to the rendering of those sets.*

We built a special radial menu option (e.g., "$x \in A$") that will convert the selected expression into its logically equivalent Set Builder notation (See Figure 8). Once in this notation, SetPad still supports most of the manipulations including commutative, associative, distributive, and the identities. Users can revert back to the standard notation using the same radial menu option.

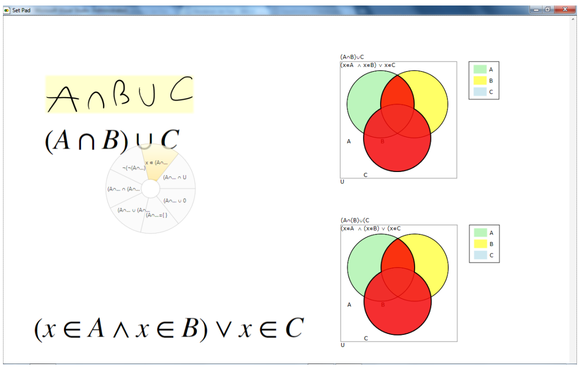

Figure 8: *Converting to "x element of" notation.*

3.5. Elements In Set Problems

Another feature available from the radial menu is the ability to specify the actual elements that exist in a certain set. A user can double-tap on any set to bring up the radial menu, and select the special element entry option (e.g., "A = { }"), which causes a new special pen entry window to appear. In the window, a user can sketch numerical values that exist in the set (See Figure 9). When they are finished, they simply close the entry window and can continue to other sets. Once all set elements have been entered for each individual set, users can sketch a new complex expression containing those sets, (e.g., "$A \cap B$"). A new window will then properly display the elements that exist in that complex expression based

© The Eurographics Association 2012.

on the set operation that was sketched and the elements that were previously entered.

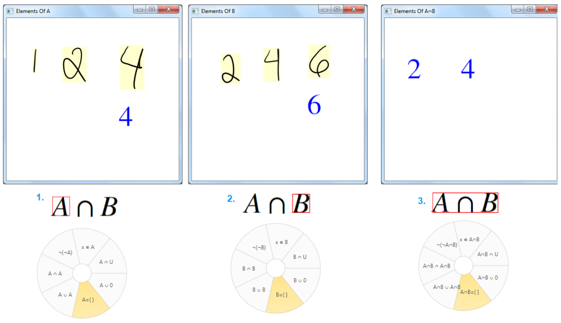

Figure 9: *Entering elements in a set. 1. A user first selects set A, opens the special element entry window, and sketches elements "1 2 4". 2. User selects set B and repeats the steps to add "2 4 6". 3. User selects complex expression and opens to see the intersecting elements between A and B.*

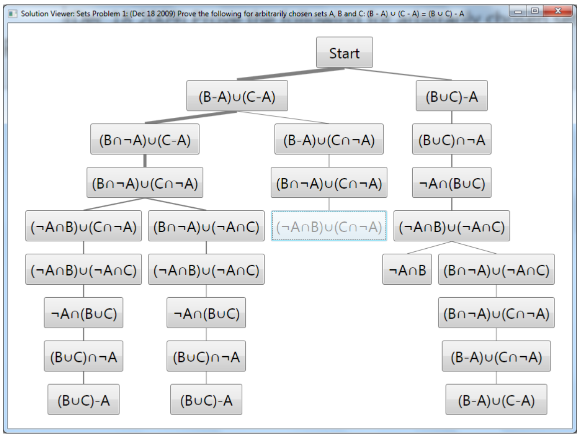

Figure 10: *The "Solution Viewer" showing a visualization of solution paths and their frequency. (Line thickness indicates the number of students that have taken that step.)*

3.6. Previous Step Branching

As students are expected to make mistakes and encouraged to explore the set algebra rules, SetPad also provides a means to branch back quickly to a previous step. By scrolling back to a previous step, users can then double-tap on any previous step to branch out and work from that point on. When this occurs, the application pans to the right of the branched step and replaces the current active expression with the selected branched step. Because SetPad allows a near unlimited workspace, this branching provides a considerable benefit over traditional pen and paper or whiteboard.

3.7. Workspace Maneuvering

SetPad provides several basic features to easily and quickly control the workspace as the user moves from step to step of the problem. By using a multi-touch enabled device, users can zoom in and out using familiar pinch-to-zoom controls. The workspace can also be quickly translated in all directions to review previous steps by dragging a single finger on any part of the workspace not displaying an expression.

3.8. Real-time Observation

Another aspect of SetPad is the "Solution Viewer" feature for instructors (See Figure 10). This separately running process renders a tree showing all steps attempted thus far to solve a specific problem by any number of students running separate SetPad instances. This lets instructors observe in real-time exactly what steps students are taking to solve the current problem, as well as which solution paths are being taken most frequently (via the thickness of the line connecting the step nodes, as well as a tooltip). We believe that instructors would be able to then infer the key algebraic laws that students may be struggling with, by examining the point in the solution tree where students stray from the accepted solution or fail to proceed.

4. Uses For Instructors

4.1. Whiteboard Replacement

In discrete mathematics courses, instructors typically work through set proof problems on a whiteboard or with transparencies. This method can lead to several problems when helping students understand the solution. Most notably, a whiteboard has limited space to sketch and solve the problem, and requires good handwriting skills so that students can follow the steps. Moreover, when an instructor moves from one step to the next, the law that was applied is usually only spoken aloud, and not written as part of the step. In that case, a student could easily mishear or entirely miss the law that was used for the step.

SetPad alleviates these issues when connected to a projector. By virtue of being a digital workspace, there is nearly unlimited room to sketch and branch out. Handwriting is a non-issue as once the problem is sketched, the system recognizes and typesets the problem for students to clearly read. As the instructor works on the next steps, students can see exactly what terms and sub-expressions are being impacted and changed. Lastly, the law that was applied for the step is displayed prominently above the new expression, such that students have time to read and take notes on what was done.

4.2. Set Problem Generation

Because SetPad allows quick manipulation of existing set expressions, instructors can also use the tool to easily generate new problems for students. An instructor starts by sketching a complex expression, and then uses the tool to manipulate it into a different but equivalent form for students to

© The Eurographics Association 2012.

solve in a proof or derivation. Alternatively, by using the Venn diagrams to compare two expressions, an instructor can quickly use the tool to see that these expressions are indeed equivalent.

4.3. Grading Set Problems

Another potential use of SetPad is to aid in the grading of set proof problems by instructors and their assistants. As set proof problems may potentially have many correct solutions, it can be difficult to grade these problems consistently. SetPad via its "Solution Viewer" functionality can provide graders a quick view of the acceptable solutions for a problem, which they can use to more fairly grade the student's work. Additionally, if students were to use SetPad to solve the problems vs. pen and paper, the software itself could be extended to grade the students attempt and report the correctness of their solution directly back to the instructor.

5. Experimental Study

To evaluate the effectiveness of SetPad in solving set proof problems, we conducted a formal user study. Specifically, we compared the performance of students solving five set proof problems using the SetPad tool vs. a control group who attempted to solve the same five problems using traditional pen and paper along with a “cheat sheet” of common set algebraic laws.

5.1. Subjects and Apparatus

We recruited 20 student volunteers (17 male, 3 female) and divided them into the two groups (control and experiment) such that each group would have approximately the same level of experience and background in discrete mathematics and set theory. Participants ranged in age from 19 to 31 and were currently enrolled in either undergraduate or graduate programs of computer science, computer engineering, electrical engineering, or information technology. The experiment took approximately 0.5 to 1 hours to complete and each participant was paid 10 US dollars for their time.

Our experimental setup consisted of one multi-touch/digital pen capable laptop - an HP TouchSmart PC notebook running Windows 7. Only one person was needed to administer the experiment as a proctor.

5.2. Experimental Task

We chose five representative set proof problems from several collegiate discrete mathematics textbooks and course exams. We selected problems that could be solved via direct proof in order to have a consistent, measurable step-by-step solution and that would exercise most of the laws of set algebra. These problems are listed in Table 1.

Table 1: *The five problems that study participants were asked to solve.*

Representative Set Proof Problems	
Problem A	Prove: $(B-A)\cup(C-A) = (B\cup C)-A$
Problem B	Prove: $(A-C)\cap(C-B) = \emptyset$
Problem C	Prove: $A-(C\cup B) = (A-B)-C$
Problem D	Prove: $(A\cap B)$ and AB are disjoint. (Sets are disjoint if their intersection is $\emptyset$)
Problem E	Prove: $A-(A\cap\overline{B}) = A\cap B$

5.3. Experimental Design and Procedure

We used a between subjects design with a control (pen and paper) and experimental (SetPad) group. Both experimental and control participants were first given a pre-questionnaire that gauged their knowledge of set theory and proofs, as well as their previous experience using pen-based and multi-touch interfaces.

The experimental group was first trained with SetPad for up to fifteen minutes by working through a number of manipulation exercises on given set expressions. These samples exercised each of the algebraic set laws that would be needed to solve the five set problems. When the participants finished with the training, they began the timed part of the study in which they had to solve the five set problems as quickly as possible. If the participant solved the problem, or was unable to solve the problem within five minutes, they moved on to the next problem. We recorded the time each participant spent as well as correctness for all five problems. If they were unable to correctly solve a problem within the five minute span, we recorded their time as the five minute max (i.e., 300 seconds). Afterwards, participants were given a questionnaire (shown in Table 4) using a seven-point Likert scale (1 equals strongly disagree and 7 equals strongly agree) asking them to evaluate the tool's effectiveness in solving the five problems and how effective they believed SetPad would be if it were to be used in a classroom.

Similarly, the control group was asked to solve the same five problems, but were asked to do so on paper using pen or pencil along with a special “cheat sheet” that contained all the algebraic laws that would be needed to solve the problems. Again, participants attempted to solve each problem, or if they were unable to solve it within five minutes, they moved on to the next problem. We recorded the time each participant spent as well as correctness for all five problems. There was no post-questionnaire given to the control group participants.

5.4. Results

From Figure 11 and Table 2, we observed that the participants who used SetPad spent significantly less time working on the problems than the control group, with the exception of Problem E.

We also observed a significant difference in the participant's abilities to correctly solve the problems between the

© The Eurographics Association 2012.

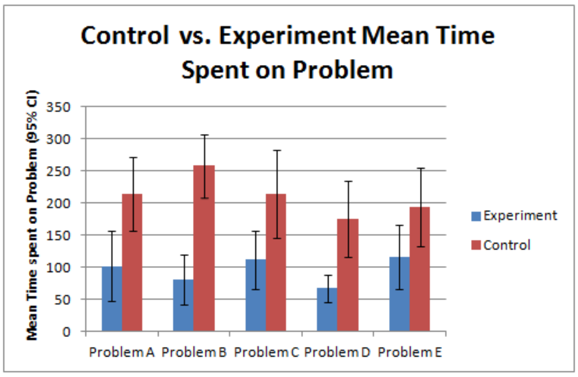

Figure 11: *Results for mean time spent working on the problems.*

Table 2: *One way ANOVA where tool was the independent variable (pen and paper, or SetPad), and time on task was the dependent variable.*

One way ANOVA Results	
Problem A	$F_{1,18} = 7.80, p < 0.05$
Problem B	$F_{1,18} = 31.09, p < 0.05$
Problem C	$F_{1,18} = 5.99, p < 0.05$
Problem D	$F_{1,18} = 11.06, p < 0.05$
Problem E	$F_{1,18} = 3.68, p = 0.07$

two groups as shown in Figure 12. Despite both groups having similar background and experience with set problems, all but one experiment participant was able to correctly answer all five questions in the allotted time. The control group on the other hand only had a 50% success rate for the first three problems, fared slightly better with 80% completing Problem D, and 70% completing problem E. A Fisher's exact test as shown in Table 3 shows that participants did a significantly better job at answering the problem correctly with SetPad for problems A and B and did no worse for problems C,D, and E.

We believe that these results indicate that SetPad is at least as efficient, if not more, in solving set proof problems, and is more effective in helping students discover and find the solutions to complicated algebraic transformations. We witnessed that when the control group failed to solve a problem, it was either due to not knowing where to start (stuck on the first step), or was due to some immediate logic flaw in early steps that led them down a path where they never recovered. It they had a tool to give them immediate feedback, we believe they would have able to see their mistakes and recover and solve the problem.

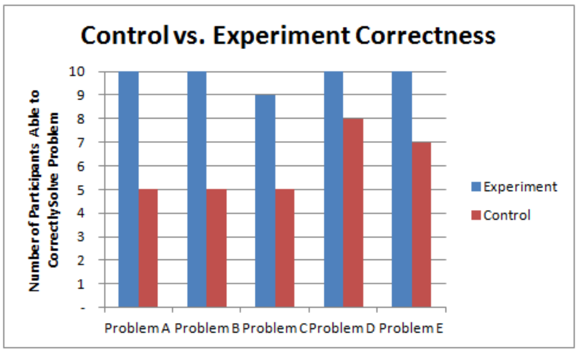

Figure 12: *Number of participants able to correctly solve the problems.*

Table 3: *Fisher's exact test for experiment vs. control correctness results.*

Fisher's Exact Test Results	
Problem A	$p < 0.05$
Problem B	$p < 0.05$
Problem C	$p = 0.14$
Problem D	$p = 0.47$
Problem E	$p = 0.21$

5.4.1. Survey Results

Following the timed trial, the 10 participants using SetPad then answered a questionnaire, based on [Dav89] (see Table 4) using a seven-point Likert scale.

Figure 13 shows the mean responses from the questionnaire in regards to SetPad's effectiveness during the experiment. We see that participants generally favored SetPad as an effective tool to solving the set problems, especially in regards to the"real-time display of algebraic laws." Some students that had trouble with the "touch and drag interface" suggested we increase the spacing of the symbols to make it easier to select, and to allow the use of the stylus instead of the multi-touch interface. Note, for this experiment we had our participants solely use the multi-touch interface for simplicity and consistency in the experiment, but SetPad does support the stylus pen to manipulate the expressions.

Since students had the chance to use SetPad, we also asked them whether the tool would be useful in the classroom. Figure 14 shows the mean responses from the questionnaire questions that asked about their perception of SetPad's effectiveness if used in a classroom.

Again, participants showed general belief that SetPad would be effective in a classroom, particularly in exploring set laws and using it to aid in their homework. Unsurprisingly, there was some general concern that they might become too dependent on the tool, but opinions were mixed

© The Eurographics Association 2012.

Table 4: *Questions presented to participants in experiment post-questionnaire.*

SetPad Effectiveness in the Experiment	
Q1	SetPad made it easier to solve the set proof problems vs. if you had to solve them by hand with pen and paper.
Q2	The SetPad touch and drag interface was easy to use in manipulating the set expressions.
Q3	The real-time display of valid algebraic laws in SetPad was helpful in solving the problems.
Q4	SetPad was effective at visualizing the set relationships presented in the problems.
Q5	I had enough practice to adequately perform and solve the problems in the experiment.
Q6	The HP TouchSmart PC was appropriate for use in testing of SetPad.
SetPad Effectiveness in a Classroom	
Q7	The discrete mathematics course that I took previously adequately prepared me to solve these types of problems.
Q8	Using SetPad in a collegiate Discrete Mathematics course would have helped me learn and understand the material better.
Q9	SetPad would have helped me to complete my Discrete Mathematics coursework assignments more quickly and correctly.
Q10	SetPad would have allowed me to better explore and learn the set algebra laws it was available to use while taking a Discrete Mathematics course.
Q11	I would become too dependent on using a tool like SetPad in solving problems in a Discrete Mathematics course.

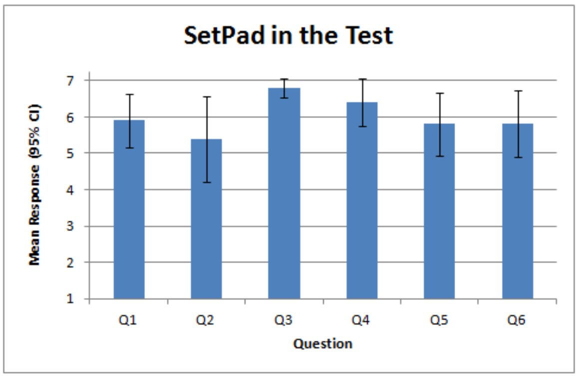

Figure 13: *Mean results for post-questionnaire survey questions of SetPad's effectiveness in the experiment.*

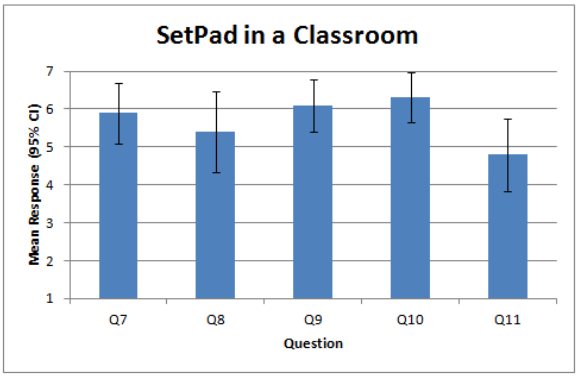

Figure 14: *Mean results for post-questionnaire survey questions of SetPad's potential effectiveness in a classroom.*

and some students suggested a balanced mix of using the tool and working problems by hand would be optimal.

6. Discussion

One criticism of the tool is that SetPad simply addresses a single topic (set theory) from a single course (discrete mathematics.) We believe that most of the insights gained from this research would easily be extended to other applications and subjects. Many areas of mathematics (including traditional algebra, calculus, logic and inference problems, etc.) could easily benefit by presenting students with a digital sketch-based workspace that provides instant feedback as students solve problems in logical steps. Moreover, by being purely digital and step-based as such, the "Solution Viewer" concept could be easily implemented for those subjects as well. Further research is needed in expanding the concepts of SetPad into other branches of mathematics, perhaps even using the rules system that SetPad developed as a common general-purpose framework for future tools.

Lastly, during the user study we observed that students were able to quickly learn and pick up the gestures for solving set theory proof problems. With just a limited amount of training, their solution times were faster than traditional pen and paper, and they achieved a much higher success rate for correctly solving the problems. We believe that if deployed in an actual discrete mathematics course, SetPad could be effective in improving student understanding and performance, as students would be exposed to even longer training and practice using the tool. A future user study that measured students' performance long term with and without the tool would help to support this claim.

7. Future Work

SetPad is still a prototype application, and needs further iteration and testing to improve its effectiveness before becoming a full use educational tool deployed in a discrete mathematics classroom. Additionally, there are several features that SetPad could support to better complement its coverage of the set theory domain of discrete mathematics. Specifically, the tool is heavily focused on direct solutions to set

© The Eurographics Association 2012.

theory proofs (i.e., proving the equivalence of two expressions), and could be expanded with further features to help solve proofs via contradiction, etc.

Another future application of SetPad that was not explored yet is the potential for the tool to discover and illustrate common mistakes that students make. Traditionally with algebra and other step-by-step problem solving, there are many common misconceptions that many students make that lead them astray early on in the problem. SetPad could feasibly learn the common bad paths that a user makes, and notify him or her of those mistakes, possibly in real-time, before or as the mistakes are repeated.

Lastly, the "Solution Viewer" feature concept of the tool is still very preliminary and offers just a glimpse of the possibility of what can be accomplished if SetPad were setup in a distributed network of students. By collecting solution metrics digitally from students in this manner, there exists a plethora of analysis data and applications that could be presented to the educator to improve his or her teaching method and to identify trouble areas in student learning.

8. Conclusions

We have presented SetPad, a new tool that lets both computer science students and instructors interactively sketch and explore set problems. Users are able to freely sketch a set expression, and then manipulate that expression using the pen or touch input to apply the laws of set algebra. The tool provides instant visual feedback to the user in real-time as they drag the touched expression around, indicating if and what algebraic law can be applied with the current gesture. We also conducted a formal user study of SetPad for students that have taken a discrete mathematics course. Based on this study, we believe SetPad is an efficient tool for solving set proof problems and has the potential to help future students understand how sets relate in expressions in addition to the algebraic rules of sets. The evaluation results also indicate that there is strong potential for SetPad to be an effective complement to the typical pen and paper approach to learning to solve set problems, as it provides instant legible feedback to the user while manipulating the expressions.

9. Acknowledgements

This work is supported in part by NSF CAREER award IIS-0845921 and NSF awards IIS-0856045 and CCF-1012056. We would also like to thank the members of the ISUE lab for their support and the anonymous reviewers for their useful comments and feedback.

References

[BE93] Barwise J., Etchemendy J.: *Tarski's World: Version 4.0 for Macintosh (Center for the Study of Language and Information - Lecture Notes)*. Center for the Study of Language and Information/SRI, 1993. 2

[BE02] Barwise J., Etchemendy J.: *Language, Proof and Logic*. Center for the Study of Language and Information, 2002. 2

[BL10] Bott J. N., LaViola Jr. J. J.: A pen-based tool for visualizing vector mathematics. In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium* (2010), SBIM '10, pp. 103–110. 2

[Dav89] Davis F. D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly 13*, 3 (1989), 319–340. 8

[DPSR10] Delaney A., Plimmer B., Stapleton G., Rodgers P.: Recognizing Sketches of Euler Diagrams Drawn with Ellipses. vol. 16 of *International Conference on Distributed Multimedia Systems*, pp. 305–310. 2

[HO00] Hall C., O'Donnell J.: *Discrete mathematics using a computer*. Springer-Verlag New York, Inc., New York, NY, USA, 2000. 2

[HPB10] Hinckley K., Pahud M., Buxton B.: Direct display interaction via simultaneous pen+ multi-touch input. *Information Display 41*, May (2010), 537–540. 2

[LLM*08] Labahn G., Lank E., MacLean S., Marzouk M., Tausky D.: Mathbrush: A system for doing math on pen-based devices. In *The Eighth IAPR International Workshop on Document Analysis Systems* (2008), pp. 599–606. 2

[LZ04] LaViola Jr. J. J., Zeleznik R. C.: Mathpad^2: a system for the creation and exploration of mathematical sketches. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers* (2004), pp. 432–440. 2

[Mic11] Microsoft: Math. Computer program, September 2011. http://www.microsoft.com/math. 2

[Mil09] Miller T.: Microsoft center for research on pen-centric computing: starpad sdk. Computer program, July 2009. http://pen.cs.brown.edu/starpad.html. 2

[PWB11] Power J. F., Whelan T., Bergin S.: Teaching discrete structures: a systematic review of the literature. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (2011), SIGCSE '11, pp. 275–280. 2

[SDRP11] Stapleton G., Delaney A., Rodgers P., Plimmer B.: Recognising sketches of Euler diagrams augmented with graphs. In *2011 International Workshop on Visual Languages and Computing* (August 2011), vol. 17 of *International Conference on Distributed Multimedia Systems*, pp. 279–284. 2

[Sof11] Software R. B.: Algebra touch, for ios. Computer program, August 2011. http://www.algebratouch.com/. 2

[TT05] Thimbleby H., Thimbleby W.: A novel gesture-based calculator and its design principles. In *Proceedings 19th. British Computer Society HCI Conference* (2005), pp. 27–32. 2

[WPS*11] Wang M., Plimmer B., Schmieder P., Stapleton G., Rodgers P., Delaney A.: SketchSet: creating Euler diagrams using pen or mouse. In *2011 IEEE Symposium on Visual Languages and Human Centric Computing (VL/HCC 2011)* (September 2011), IEEE, pp. 75–82. 2

[ZBAK10] Zeleznik R., Bragdon A., Adeputra F., Ko H.-S.: Hands-on math: a page-based multi-touch and pen desktop for technical work and problem solving. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology* (2010), UIST '10, pp. 17–26. 2

[ZMLL08] Zeleznik R., Miller T., Li C., LaViola Jr. J. J.: Mathpaper: Mathematical sketching with fluid support for interactive computation. In *SG '08: Proceedings of the 9th international symposium on Smart Graphics* (2008), pp. 20–32. 2

© The Eurographics Association 2012.