

CADament: A Gamified Multiplayer Software Tutorial System

Wei Li, Tovi Grossman, and George Fitzmaurice
Autodesk Research, Toronto, Ontario, Canada
firstname.lastname@Autodesk.com

ABSTRACT

We present CADament, a gamified multiplayer tutorial system for learning AutoCAD. Compared with existing gamified software tutorial systems, CADament generates engaging learning experience through competitions. We investigate two variations of our game, where over-the-shoulder learning was simulated by providing viewports into other player's screens. We introduce an empirical lab study methodology where participants compete with one another, and we study knowledge transfer effects by tracking the migration of strategies between players during the study session. Our study shows that CADament has an advantage over pre-authored tutorials for improving learners' performance, increasing motivation, and stimulating knowledge transfer.

ACM Classification:

H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

Keywords:

Game; Multiplayer; Tutorial; Learning.

INTRODUCTION

Learning to use feature-rich software applications is a notoriously difficult task for new users. As such, there has been a long line of research on understanding and improving the learnability of software. As a result, numerous mechanisms for providing assistance have been proposed, such as online help, interactive tutorials, and video-based assistance.

Unfortunately, the process of learning to use such traditional mechanisms can be tedious, and research has shown users are resistant to relying on help systems [2]. To address this limitation, gamified approaches have been recently explored, in an effort to increase the engagement level of learning [13, 20, 23, 32]. Such approaches rely on design elements that are typical in games, such as feedback, guidance, time pressure, and rewards.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
CHI 2014, April 26 - May 01 2014, Toronto, ON, Canada
Copyright 2014 ACM 978-1-4503-2473-1/14/04...\$15.00
<http://dx.doi.org/10.1145/2556288.2556954>

However, existing gamified approaches for software learning have not embraced the benefits of social and crowd-sourced learning solutions. In particular, there have been a number of studies that have shown the benefit of "over-the-shoulder" learning environments, where users can observe and acquire skills directly from their peers or user community [36]. Since many existing gamified approaches to software learning have been "single-player", they miss out on this potential benefit of social learning.

In this paper, we present CADament a new multiplayer online game system for learning to use AutoCAD, a feature rich software application for creating mechanical and architecture design. CADament is designed to be an online game that any number of players can join. All active players compete head-to-head, attempting to complete a series of short time-coordinated levels, faster than an opponent they have been matched up against. Two variations of this game are explored – one in which players see a split screen of their opponent's video while completing the levels, and one in which players review their opponents' video after each level has been completed.

By adding a multiplayer element, players have the opportunity for not only independent learning in an gamified environment, but also have the opportunity of acquiring new skills by observing the workflows and strategies of their opponents. Furthermore, the multi-player environment adds an additional competitive aspect to the gaming environment, potentially increasing a player's motivation to learn and improve.

After describing our system, we present an empirical evaluation of our system. In this study, we compare the two variations of our game, with a baseline help condition, where participants complete levels independently. We propose a user group study methodology where participants compete with one another, and we study knowledge transfer effects by tracking the migration of strategies between players during the study session. Our study shows that CADament has an advantage over pre-authored tutorials for improving learners' performance and stimulating knowledge transfer.

RELATED WORK

In this section, we review previous research in software learning, tutorial systems and gamified tutorials.

Software Learning

Since the early days of HCI research, software learnability has been an important research topic [12, 26]. Early results include Carrol et al.'s investigations of the learnability of word processing tools [2, 3, 24]. A full review of research on software learnability is beyond the scope of this article. We direct the reader to a recent survey on this topic [18].

Also particularly relevant to our work is the research of Twidale and colleagues on "over-the-shoulder" learning [36]. Such work states that although some people like to learn a system on their own, many prefer learning in a more social context. As such learning how to use a computer application is often a collaborative activity. Our multiplayer game system is designed upon this premise.

Software Tutorial Systems

Early work in software help indicated the importance of minimalist and task-centered help systems [2]. This is because of an active user's production bias [2, 3, 35], where a user does not want to take time away from the task to focus on learning about the system. Although static online help supports quick access, it can be difficult to communicate complex graphical operations through text.

Animated assistance strategies have been explored [5, 17, 25, 28, 29]. There has been some debate over the benefits of animated or video based assistance. Earlier work argued that such materials enforce a passive learning process, force users to work at the pace of the video, and may be detrimental to long term learning. However, recent work, such as ToolClips [17] and MixT [29], have shown clear benefits of using video based assistance for certain graphical operations. Pause-and-Play also provides a unique way to deliver video content, automatically pacing the video based on user progress [29]. Ambient Help [25] mimics an over-the-shoulder learning environment by looping videos of other user's workflows on a secondary ambient display. By automatically loading videos related to the user's current task, it was found that users discovered new concepts and workflows from the videos.

Alternatively, interactive tutorial systems can be integrated within the software application itself [1, 14, 20, 29, 31]. Stencils-based tutorials [20] overlay a stencil over a software application with a hole to click-through and guide a user to perform the correct steps. Fernquist et al. suggest that tutorial experiences should be as compelling as possible, so that users stay engaged [14]. The concept of flow [8] provides a general guideline for creating an engaging user's state of "optimal experience". Qualifying factors for achieving flow include: perceptions of clear goals, immediate feedback, and matched skills and challenges [4]. Lafreniere et al. [21] integrate video based demonstrations in interactive tutorials, where pre-recorded videos are generated by the user community.

Gamified Tutorials for Software Applications

Gamification is the use of design elements, which are characteristic for games, in non-game contexts [11]. Gamified systems have been used in many domains including educations [16, 30, 33]. The level of engagement that tutorials provide could potentially be increased by integrating gaming elements. To create engaging learning experiences, several games were designed for learning software, including the Microsoft Ribbon Hero [32], Adobe LevelUp [20], Jigsaw [13], and GamiCAD [23].

The above systems were primarily designed for single player user experiences, so their competition level is limited. Also, because the content of those games are pre-designed, the learning experience is not flexible and adaptive. When the player successfully covers the content, the educational experience has come to an end. Here, we proposed a general approach to apply multiplayer gamification in learning software applications. By providing a view into other player's strategies, users are continuously exposed to new content that they can adopt.

Because learning often happens in social environments, such as classrooms and schools, the multiplayer environments of modern games show promising applications in the educational domain [9, 15, 34]. However few games have been developed to support an over-the-shoulder learning experience. For example, multiplayer games for teaching math may not show exactly how other students do the addition tasks.

Recent research work has also investigated the educational benefits of 1-on-1 competitive games such as chess and StarCraft II [15]. However, in such games, one experienced player can only be directly challenged by another player in the same round. CADament leverages video streaming and broadcasting technology, and it allows one player to have many challengers.

Software Skill Contests

CAD drafting contests¹ are popular events among CAD software learners. Usually, competitors need to draw specified views of a given drawing problem, or model small objects within a designated time period, and submit their drawings or models using emails or a website. Informal CAD drafting contests often happen in CAD forums, where users challenge each other by posting and completing drawing and modeling tasks. Similarly, image editing contests, such as Layer Tennis², exist in the professional designer communities. The Photoshop contest is played through sequential alternating editing of an image using email or a web server. A common problem of those software contests is that the usage skills and techniques are difficult to be transferred to other learners, as players and spectators cannot observe their opponent's workflows.

¹ <http://www.idea-online.org/competitions.html>

² <http://layertennis.com/>

In CADament, videos are recorded and live streamed from software users working on the exact same tasks, then videos are integrated in a competitive multiplayer gaming environment. We hope that the synchronized and context-based videos can improve the user's over-the-shoulder learning experience, while multi-player gaming creates a feeling of excitement and engagement.

MULTIPLAYER GAMIFICATION ELEMENTS

The emergence of the world-wide-web has made multiplayer games extremely popular. In general, the game elements in single player games are also available in multiplayer games, such as an epic background story, rewards, time pressure, feedback and clear goals [23]. But multiplayer games have several unique elements.

Competition with Real players

Previous research into the experience of video games has shown the importance of the role of challenge in the engaging game experience [7, 19]. A game that allows for diverse people to play diverse ways is often more interesting and rewarding. Single player games challenge players through pre-designed levels and time pressure. Players are motivated by rewards and stimulus and play against pre-programmed challenges and/or AI-controlled opponents, which often lack the flexibility and ingenuity of regular human thinking.

Multiplayer games introduce different challenges in direct general competitions and head-to-head competitions. General competitions are based on the individual's score or relative standing [1]. Head-to-head competitions exist in many types of multiplayer games, such as racing, sports tournaments and MMORPG (massively multiplayer online role-playing game). In those games, people have different personal styles when playing against each other.

Sharing Game Play Experience

Sharing is a core element of multiplayer games, where players often share the same task space, common goals and challenges, unified feedback and performance metrics (such as leaderboards). Many multiplayer games setup real-time interactive communications among players to help them observe their opponents' actions and provide an immersive experience. This actually encourages learning by watching other players who exhibit different strategies.

For example, in Need for Speed World, a popular massively multiplayer racing game, drivers all share the same racing tracks and real time graphical effects during a race. Drivers not only see other cars' current positions and rankings in a map, they can also observe real time actions and techniques used by other players. In some other games, player's actions are shared after or between tasks. In Wordament, a successful word spelling game, players can only see other players' performance and answers during the breaks between tournaments. Some games have both forms of sharing. For instance, many racing games also provide

recorded game replay for players to review their performance in the last racing event.

Pacing

The pace of video games is the rate at which players experience new challenges. It can be maintained at appropriate levels throughout the game [27], so that a game's design applies pressure but does not frustrate the player. Player fatigue can be also minimized by varying pacing during gameplay [10].

Multiplayer games often do not have a "pause key" as in a single player-game, since the progress of the game is synchronized among all players. For example, MMORPG games, such as World of Warcraft, never stop within a virtual persistent game world and continue to exist and evolve while a player is offline and away from the game. Other multiplayer games, such as Wordament, generate rapidly-short-iterating rounds. Slow periods follow intense ones and forced "time-outs". This type of lightweight design can offer opportunities to socialize, catch your breath and anticipate things to come. It often achieves a balance between competitive and enjoyable experiences while softening the impact of defeat. Rapidly-short-iterating rounds also assist the knowledge transfer by learning skills from many different players and help new users practice the same tasks multiple times.

DESIGN GOALS

In this section, we define four design goals, to guide the design of our multi-player software learning game.

Engaging

In terms of engagement, we aim to provide an environment that users will enjoy and be motivated to learn the software and improve their skill level. Competitive game play, including both head-to-head competition and general competition, helps learners strive to be better and creates a feeling of excitement and increases engagement level.

Knowledge Transfer

In general, we hope that the system can assist transferring knowledge among users. Over-the-shoulder learning [25, 36] is informal, spontaneous help – a given interaction that is often used by people to learn from their colleagues in their workplace. We create a similar environment to show other players' solutions within the context of the game.

Low Cost

Authoring tutorials for software applications is a time consuming process. We wish to create a system that can help experienced users easily share their knowledge and skills to novice users and lower the cost of authoring learning content. In multiplayer online games, every competitor is implicitly incentivized to be a content producer and create learning experience for other players.

Flexible

We hope the learning content and interactions are rooted in tasks that are relevant to the learner's needs and interests. We also want to balance between competitive and

enjoyable experiences. The learning experience should match a learner’s skill level to create an “optimal experience” or flow [8].

CADAMENT SYSTEM DESIGN

We design CADament, a competitive multiplayer online game for learning AutoCAD, which is a widely used software application for both 2D and 3D drafting and design. We choose AutoCAD for this project because 1) AutoCAD represents a large number of software applications with a direct manipulation interface; and 2) it is considered challenging to learn for novices [18]. A car racing theme was used.

Game Structure

CADament runs continuously in 30-seconds rounds, with a 30 second break between rounds. Players can join the game anytime and participant in the immediate next round with all online players. In each round, players solve a drafting task using AutoCAD. Each task, T_n , repeats in 3 consecutive rounds, with a break B in between each round, before being replaced by the a new task T_{n+1} (Figure 1). This rapidly-iterating round format should help the players learn skills from different challengers and allow new users to practice the same tasks multiple times.

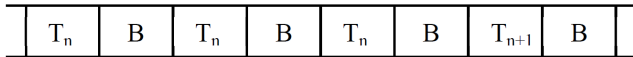


Figure 1. Players have 30 seconds to complete the round’s task (T_n) followed by a 30 second break (B).

Tasks

The tasks in the game were designed to help players to continuously increase their skills through progressive disclosure. For our prototype, we developed a set of tasks for the TRIM tool. An actual deployment could iterate through tasks for many different tools, or contain tasks requiring multiple tools. Tasks gradually increased in difficulty, and eventually looped back to the initial task.

Player Progress Feedback

CADament does not enforce a specific workflow, or a single approach that all players must follow. Instead, CADament compares each player’s drawing with the pre-generated final result every time a command is executed, and reports the percentage of the task that has been completed. For example, if the task requires the user to trim 4 lines, then after trimming 3 lines a player’s progress would be 75%.

Screen Sharing Game Modes

In our design we leverage a sharing mechanism to assist software learning and knowledge transfer. By sharing the screen of online players, we can create a mixed experience of head-to-head competition and over-the-shoulder learning. We explored two different game modes to allow players to see each other’s screens, *Record & Replay (Replay)* and *Split Screen (Split)*.

In *Record & Replay*, the players’ screens are recorded during the competition. Players can watch their opponent’s recorded video during the breaks between rounds. This allows users to focus on the skills used by their opponents.

In *Split Screen*, players can see their opponent’s screen in real-time while completing the tasks. The real-time sharing simulates a head-to-head competition experience, similar to many online multi-player games.

Both designs can simulate an over-the-shoulder learning experience. In the *Replay* mode, learning happens during the slower pace of the break period, and the player can focus entirely on completing the level during the round. In the *Split* mode, users have the opportunity to learn from their opponents strategies right away in the same round, instead of waiting for the next break.

User Interface During Rounds

During the rounds, each player competes against another online player to complete a drafting task using AutoCAD. At the beginning of each round, the initial drawing data will be automatically opened in AutoCAD.

In *Split* mode, every player produces a live broadcast of their AutoCAD screen area. A player sees their opponent’s screen next to their own (Figure 2). Through the live streaming video, the player is able to observe her/his opponent’s detailed actions. An overlay indicates the opponent’s mouse behaviors and keystrokes (Figure 2).

In *Replay* mode, the opponent’s live video stream is not displayed but is replaced by a blank image. Additionally, the players’ application screen videos are saved and played during the 30 seconds break period.

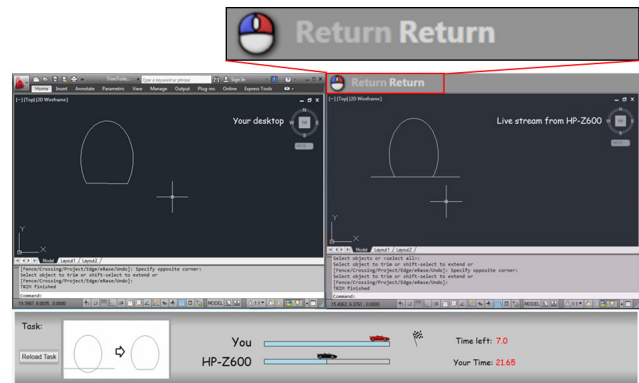


Figure 2. Competition mode. *Split* mode has players see their opponent (right side). For *Replay* mode, the region is replaced by a static image. Inset: Mouse/key-stroke visualization bar.

In both game modes, a task bar is also displayed below the main screen (Figure 3). A before-and-after image describes the goal of the current task. A timer starts counting down as soon as the round starts. The players’ real time progress is visualized using two progress bars, and a waving checkered flag is displayed once a task is finished.

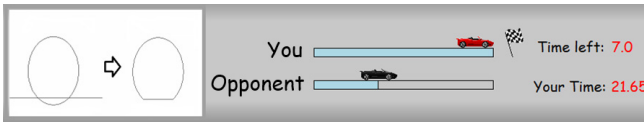


Figure 3. Task bar of CADament UI.

User Interface During Breaks

Figure 4 shows the view during the breaks between rounds. The performance data of a player and the player’s opponent for the previous round are displayed on the left side of the screen. A ribbon is displayed for the winner of the head-to-head match-up. A leaderboard shows the rankings of all players currently competing, sorted first by the task completion percentage and then by speed (Figure 4). Additionally, in the *Replay* mode, a video player automatically plays the opponent’s video after the round on the right (Figure 4A). Users can also replay the opponent’s video during the 30 second break time by clicking the ‘replay video’ button.

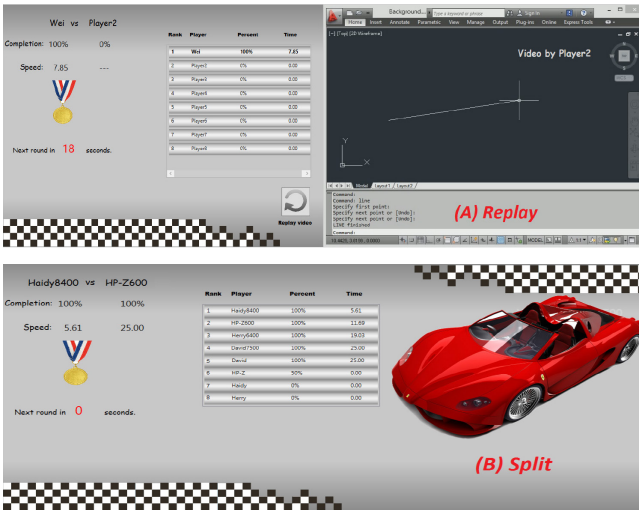


Figure 4 Screen of (A) *Replay* and (B) *Split* during breaks.

Matchmaking

At the beginning of each round, CADament matches each player with an opponent. The mapping from player to opponent is not one-to-one. That is, if player A is matched to player B, player B is not necessarily matched to Player A. This allows for more flexibility in the matchups. For example, all players (except the leader) could be matched against a player slightly better than themselves. Many-to-one matchups are also possible. For example, all players could be matched against the leader in a single round.

To create a balanced gameplay experience, CADament helps each player find the opponent 1) who can provide useful, diversified skill and knowledge to help the player complete the next task; and 2) whose skill level matches the player’s level. In addition to the above requirements, there are two types of situations we try to avoid. First, if everyone plays against the top ranking player, a novice player may feel frustrated if she/he keeps being defeated by the best player. Second, if two novice players are always

matched together, they might not learn anything from seeing each other’s workflows.

Our matching algorithm is based on the rankings for the immediately preceding round:

1. The leader is matched to the second place player.
2. A player that successfully completed the last task will be matched with the player who was one rank above.
3. A player that did not complete the task will randomly play with one of the players who completed the task.
4. If no player completed the task, everyone will be matched with the player who has the highest task completion percentage.
5. When there are multiple candidate opponents, based on the above rules, the opponent is randomly selected from the candidates.

SYSTEM IMPLEMENTATION

We developed a system framework for our real-time multiplayer game using video sharing. In this framework, game progress controls, game data (tasks, player profiles), video streaming, and player matchmaking are managed using the cloud-based service, Microsoft Azure (Figure 5).

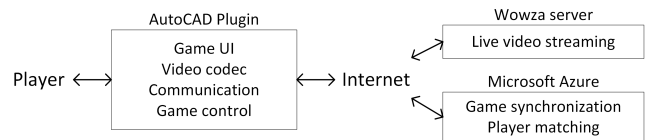


Figure 5. CADament system.

CADament updates a large amount of player’s game data frequently in real time and synchronizes the game progress on every client. We used WebSockets (SignalR) in our client-server communication component, which enables high frequency updates from the server to support a real-time gaming experience. For sharing the players’ screens and actions, screen captured videos need to be broadcasted over the internet. In the *Split* game mode, the action-viewing delay between one online player’s action and the generated video being played at another player’s game needs to be within an acceptable range. In order to achieve this goal, we deployed a live video streaming server (Wowza). Videos are encoded at each player’s computer in real time, and then “fed” to the streaming (broadcasting) servers. Finally, they are distributed to multiple clients for viewing. We optimized the whole process so that the action-viewing delay is less than 2 seconds in our local lab intranet environment. In the *Replay* game mode, CADament records every player’s screen stream to their local computer first, and then starts streaming based on demand when the round ends and an opponent needs to view the video.

USER STUDY

We performed an empirical user study to evaluate the CADament system. In particular, our goal of conducting this study is to answer three main questions: 1) Does our multiplayer software game improve learning, in comparison

to single user learning experiences? 2) Which game mode is more effective, *Record & Replay* or *Split Screen*? 3) What is the nature of knowledge transfer when players of different skill levels compete? To answer these questions, we performed a between-participants study with 3 conditions – one condition for each game mode, and a third baseline condition consisting of a single player tutorial environment.

Participants and Pre-Training

Twenty-four participants (13 female) between the ages of 19 and 30 were recruited via online postings in two universities. None of them had used AutoCAD. All participants played video games occasionally. Five of them play multiplayer online games regularly. One participant reported not playing any multiplayer video games at all. To conduct the study, we divide the 24 participants into 3 groups of 8, with each group being assigned to one of the conditions (Figure 6). For each group, we randomly choose 4 of them to serve as “trained” users, and spent 15 minutes to teach them how to use the TRIM tool. This allowed us to simulate an environment where both novices and experienced users were participating, introducing knowledge into the gaming environment.

The TRIM tool works by selecting a set of “cutting edges” and then selecting a set of lines to trim, at the point which they intersect a cutting edge. The order of steps which must be performed can be difficult for a new user to master, and there are many shortcuts which could be used to improve efficiency with the tool. As such, it serves as a good tool to use within a learning study [23].

To further understand the effects of knowledge transfer, we gave the trained participants two additional “tips” which we could reliably track the usage of throughout the study. The trained participants were split into two pairs. Pair A was taught Tip #1 and pair B was taught Tip #2. The two tips are typically exhibited by expert users, and can help users finish trimming tasks efficiently, but may not be immediately obvious to novices:

Tip #1 (Crossing), users can select multiple objects to trim by dragging the mouse over the drawing area.

Tip #2 (Select all), users can press the ENTER key and select all objects as cutting edges.

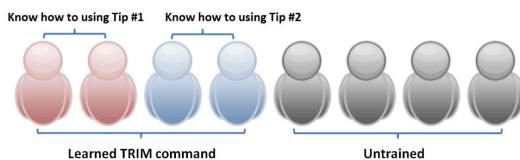


Figure 6. A user group has 4 untrained people and 4 trained people (two trained with tip #1, two trained with tip #2).

Conditions

The first two conditions were based on the two game modes of our system (*Replay* and *Split*). The group of 8 participants assigned to each of these conditions all

completed the study together, in an 8-player session. Participants in this session could communicate, but could not verbally share any information related to learning how to complete the tasks.

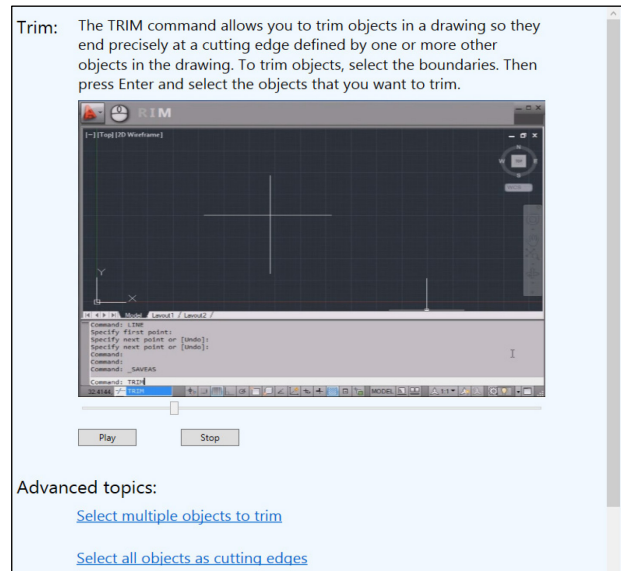


Figure 7. The tutorial webpage used in the Tutorial condition.

The third condition (Tutorial), which served as a baseline, was a non-gamified single-user environment. The 8 participants (4 of them were untrained) in this condition completed the study individually at separate times. The Tutorial condition was also performed using the CADament framework, but we removed all of the game elements (graphics, leaderboard, progress bars, racing theme and animations) from the system. Instead of playing with other people, participants in the 3rd condition worked alone on the drafting tasks, and were provided with a tutorial website (Figure 7). The website consisted of a main page that explained how to use the TRIM tool with text and a video. In addition, at the bottom of the page there were two links to pages that explained the two tips. These pages also included both text and a video. As such, the amount of information within the tutorial was equivalent to the information that the trained users were seeded with in the other two conditions. The webpage was available during the 30 second breaks. Similar to the Replay condition, participants could review material after the task, and between rounds. Apparatus

The study was conducted in a room with 8 PCs. A standard mouse and keyboard were connected to each PC. All PCs were connected to a local Gigabit Ethernet and the internet through a regular office connection. All displays had the exact same size (19 inches) and resolution (1600 by 1200). Separating panels were set up to prevent participants watching their neighbors’ screens (Figure 8). While this environment is not a true representation of a distributed online environment, it provides a close approximation for the purpose of our research questions.



Figure 8. Study environment.

Procedure and Design

The study consisted of a series of 30-second rounds separated by 30-second breaks, as described in the design of the CADament system. A total of 12 training tasks were designed for the TRIM tool, and were presented in increasing order of difficulty. Each task was repeated for 3 rounds before moving to the next task.

To measure the users' learning progress, we also included a testing task. The same testing task was injected before the first task, and after the fourth, eighth and twelfth task. Each of these four instances was repeated for 3 rounds.

In addition, a warm-up task was included at the beginning of the study. The warm-up task used the LINE tool, and was used to familiarize participants with the study environment. The warm-up task was repeated 3 times.

In total, participants completed 48 training and testing rounds. Since the system ran on a timer, the study took 48 minutes. Participants answered a questionnaire and provided comments after all training rounds and test rounds were finished. The entire session took approximately 75 minutes, including the pre-training time.

Quantitative Results

Testing task times

Figure 9 shows the average completion times across all untrained users in each of the testing tasks. A time of 30s is assigned to rounds which were not completed. By the last task, untrained *Replay* users completed the testing tasks in 6.4 seconds on average, which is less than half of the time for the two other conditions. However, this effect did not reach a level of significance ($F_{2,9} = 3.80, p = 0.064$).

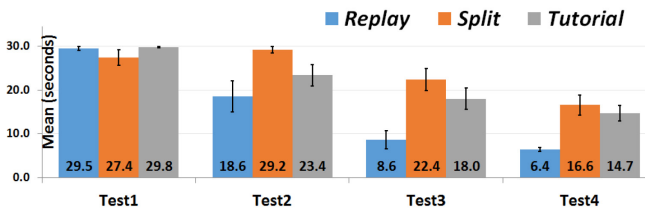


Figure 9. The average test completion times of untrained users in the three conditions.

Figure 10 compares the mean test completion time of all trained and untrained users for Test 4. It is interesting to see that in the *Replay* condition, the untrained users are faster

than the trained users. In the other two conditions, the untrained users remained slower than the trained users. The post-hoc pairwise differences here, however, did not reach significance.

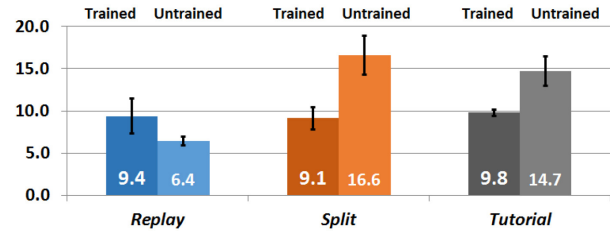


Figure 10. Compare trained and untrained group in Test 4.

Knowledge Transfer

To better explain the testing task results, we wanted to study how knowledge was being transferred from experienced users to novice users. In Figure 11 and Figure 12, we visualize each player's usage and exposure to a tip. Usage of a tip is shown as a solid red dot and exposure to the tip as an outlined dot (the player's opponent used the tip). Each row represents one of the six users who were not initially trained with the tip, and we show the trials in which either event occurs (usage or exposure) until the player has successfully completed three tasks using the tip, which indicates that a player has adopted that tip.

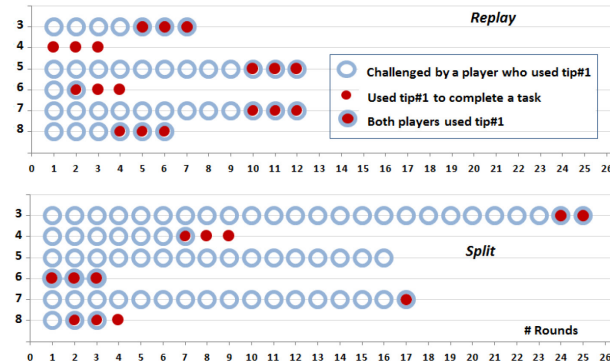


Figure 11. Tip #1 skill adoption progress for *Replay* and *Split*. The top two users were pre-trained, but not on Tip #1.

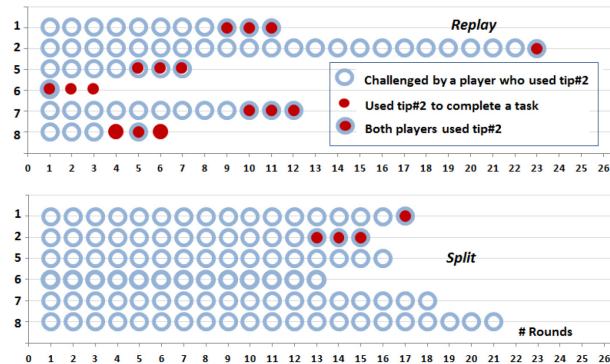


Figure 12. Tip #2 skill adoption progress for *Replay* and *Split*. The top two users were pre-trained, but not on Tip #2.

The top chart in Figure 11 shows that the 6 players in the *Replay* condition, who did not have any training on Tip #1 adopted this skill after an average of 5.7 exposures. Among those 6 participants, 2 participants learned the basic TRIM command during the pre-training (see Figure 6). The bottom chart shows that only 3 out of 6 players in the *Split* condition adopted Tip #1, after an average of 11.8 exposures. This difference is significant ($t=2.05, P=0.043$).

The results for Tip #2 are even more drastic. Figure 12 shows that 5 out of 6 players in the *Replay* condition adopted Tip #2, after an average of 9 exposures while in the *Split* condition, only 1 player adopted it, after an average of 16.6 exposures. This difference was also significant ($t=-2.05, P=0.033$).

Together, this indicates that the *Replay* condition provides for better knowledge transfer. Note that Tip #2 (Select All) seemed to be harder to distinguish from watching a video, which may explain the lower adoption rate.

Task Completion and Tip Usage

We next analyzed the untrained participants' skill progression over the training and testing tasks for the three conditions (*Replay*, *Split* and *Tutorial*). Specifically, we wanted to track and visualize if an untrained participant failed or succeeded to complete the task and if they used one or both tips (Figure 13).

The blue lines in Figure 13 indicate the outcome of a given trial for each untrained participant. Participants either failed, or succeeded using no tips, one tip or two tips, which is represented in the vertical axis. Each marker represents the exhibited behavior of the player's opponent for each round. For the *Tutorial* condition, a marker shows which webpage was displayed on the screen (main page, tip 1 or tip 2) for a trial. If the marker is filled-in, the user played the video associated with the tutorial content. It can be seen from Figure 13 that by the end of the study, all participants were able to complete the tasks successfully.

More importantly, Figure 13 again indicates that the *Replay* condition is more effective for learning, because every participant in this condition progressed from failing the tasks, to succeeding using both tips. In contrast, no untrained participant in the other two conditions ever completed a trial using both tips.

We can also compare the content that participants were exposed to throughout the study. In both of the multiplayer game conditions, the skill level of the opponents was adapted to each player's skill progress. For example, there are only a few markers under the blue lines in *Replay* and *Split* conditions. This implies that our match making algorithm tends to select more knowledgeable opponents for each player. However, in the self-learning tutorial condition (*Tutorial*), we can see that the tutorial content was very static. It shows that participants rarely accessed their tutorial content even as their skills progressed.

Qualitative Results

We measure perceptions of enjoyment, frustration, and usefulness through a survey given after all training rounds and test rounds were finished.

In the post-study questionnaire, participants were asked to rate 5 questions on a 1 (strongly disagree) to 5 (strongly agree) Likert scale. We ran Wilcoxon Signed Ranks tests to compare their answers (Figure 14). The responses to question 1 and 3 were significantly different ($p<0.05$) between *Replay* and *Split*, and the responses to question 1, 3, and 5 were significantly different ($p<0.05$) between *Replay* and *Tutorial*.

The main result is that participants found the *Replay* system significantly more enjoyable and engaging than both the *Split* system and the *Tutorial* system. Although people did not indicate significant difference in terms of frustration level, it was clear that, in the *Split* condition, learners have difficulty to learn from their opponent's video while they were working on a cognitively demanding task at the same time.

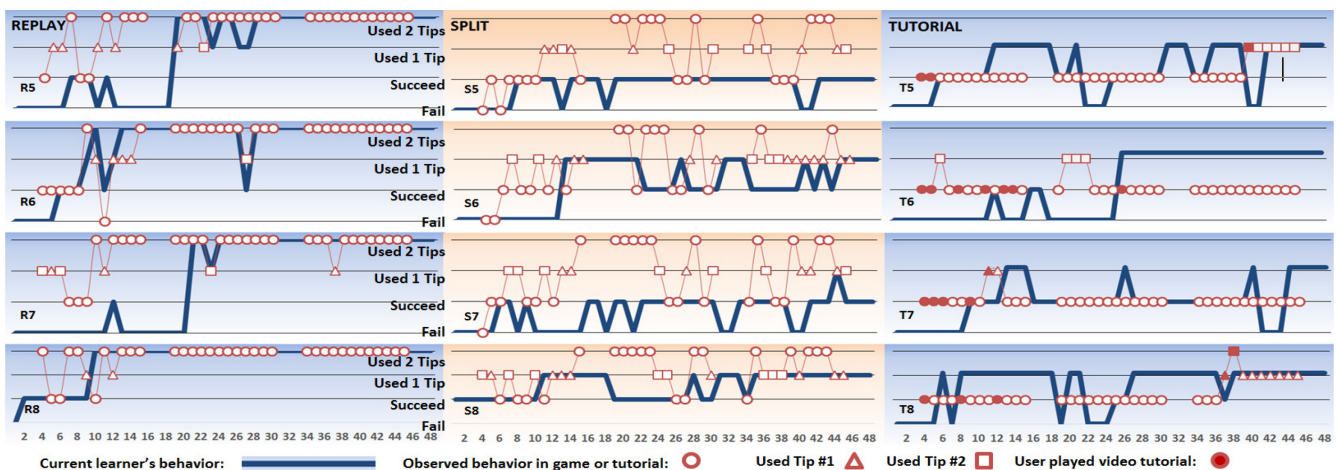
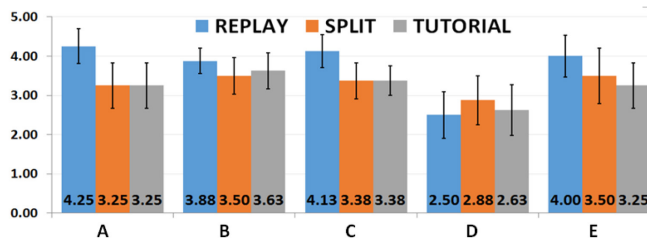


Figure 13. Skill level progression for *Replay* (left), *Split* (middle) and *Tutorial* (right) at 48 testing and training tasks.



- A. I enjoyed using this system.
 B. This system is an effective learning tool.
 C. The system provides an engaging experience.
 D. Completing individual tasks was frustrating.
 E. Seeing other people's video (tutorial) is helpful for learning how to use the TRIM tool.

Figure 14. Subjective results form the post study survey.

Comments and Observations

Participants' comments were also gathered inside the questionnaire. In general, many users commented that watching the video replay of how other people performed the same task was an effective way for learning software. One user said:

"...it is not easy to follow the prompt from the cmd (command). After a couple of game rounds, I figured it out by watching the others' replays."

CADament promoted competition. A participant commented:

"I wanted to win. This probably made me look for tricks to try to get ahead."

The design of repeating the same task multiple times did help novice users learn:

"Rationale for repeating tasks multiple times helped me feel more comfortable using the program. [I] was able to gradually improve [my] skill over time."

Participants in the *Split* condition suggested why they felt their performance was hindered:

"A window showing other player's actions is very useful but as the window [video] is real-time, it is hard to watch other player's action as you're doing your own task."

"During your task it is tough to work and see what other people were doing simultaneously."

But it is interesting to note some players developed their own strategy.

"I had watched half before trying the tricks out by myself with the remaining time."

"As there are 3 tries to each task, I watched in the 2nd trial and tried to imitated it during the 3rd if I didn't manage to get it in the first place"

Study participants were not allowed to talk about how to use the TRIM tool. However, they still were allowed to talk. During the study, we observed rich interactions among players. After the first few rounds, some players started to verbally challenge other players, especially when the leaderboard showed up. We summarized these behaviors

into three main categories: 1) celebrations; 2) challenges to other players; and 3) comments on player ranking changes. Many online games offer a voice chatting room function. It would be reasonable to expect players to use similar verbal interactions during real online game play. Chatting could also be used as an extra channel for learning.

DISCUSSION AND CONCLUSION

Authoring software tutorials is a time consuming process. CADament presents a method for developing tutorials using multiplayer gamification. As an important advantage over pre-authored tutorials, each player generates a learning experience for the other player. On the other hand, when there is a shortage of experienced players, based on our video sharing design, pre-recorded video could also be used to simulate real players in the head-to-head competitions.

Software users also have difficulty to find tutorials based on their current skill level. It is encouraging that both multiplayer game designs of CADament generate adaptive learning content through match making and competing with other players (see Figure 13).

There are several reasons that performance did not improve as much in the baseline traditional tutorial condition. First, users may have been less motivated to work on improving their performance, since they were not in a competitive environment. Second, users may have seen a video for a tip, but not realized the implications of adopting that tip on their performance. In contrast, users in the two CADament conditions would see the tip by an actual user, and realize immediately that adopting the tip would be necessary to improve performance and stay competitive in the social learning environment.

In this paper we focused on three elements of multiplayer games: competition, sharing, and pacing. In the future, we would like to investigate other important elements in multiplayer online games, such as collaborative task solving and social communities. It would be interesting to further compare CADament to single player gamified tutorials and investigate the effect of skill retention. We also want to generalize the multiplayer gamification attributes to other software applications and higher level learning tasks.

We designed CADament, a multiplayer gamified tutorial system. Compared with existing gamified software tutorial systems, CADament generates engaging learning experiences through competitions. Active players compete head-to-head, attempting to complete a series of short time-coordinated levels. CADament is also a crowd sourcing platform for players to collaboratively produce learning content while they are competitively playing the game. We investigated two game designs, where over-the-shoulder learning was simulated at different pacing stages in the game. Our study shows that the multiplayer game has advantages over pre-authored tutorials for improving learners' performance. Specifically, our analysis and

visualization shows that our multiplayer game prompted knowledge transfer.

ACKNOWLEDGEMENT

Special thanks to my colleague, Justin Matejka and John Yee, for their continuous support throughout this project.

REFERENCES

1. Bergman, L., Castelli, V., Lau, T., and Oblinger, D. (2005). DocWizards: a system for authoring follow-me documentation wizards. *ACM UIST*. 191-200.
2. Carroll, J. M., and Rosson, M. B. (1987). *Paradox of the active user. Interfacing Thought Cognitive Aspects of Human-Computer Interaction*. MIT Press, 80-111.
3. Carroll, J. M. (1990). *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill*. MIT Press.
4. Chen, H., Wigand, R.T. and Nilan, M.S. (1999). Optimal experience of web activities. *Computers in human behavior*. 15(5), 585-608.
5. Chi, P., Ahn, S., Ren, A., Dontcheva, M., Li, W. and Hartmann, B. (2012). MixT: automatic generation of step-by-step mixed media tutorials. *ACM UIST*. 93-102.
6. Clark, R.E. (1983). Reconsidering research on learning from media. *Review of Educational Research*, 53, 4, 445-459.
7. Cox, A., Cairns, P., Shah, P. and Carroll, M. (2012). Not doing but thinking: the role of challenge in the gaming experience. *ACM CHI* 79-88.
8. Csikszentmihalyi, M. (1988). The flow experience and its significance for human psychology. *Optimal Experience: Psychological Studies of Flow in Consciousness*. 15-35.
9. Delwiche, A. (2006). Massively multiplayer online games (MMOs) in the new media classroom. *Educational Technology & Society*, 9 (3), 160-172.
10. Desurvire, H., Caplan, M., and Toth, J.A. (2004). Using heuristics to evaluate the playability of games. In *ACM CHI*. 1509-1512.
11. Deterding, S., and Dixon, D. (2011). From Game Design Elements to Gamefulness: Defining "Gamification". *ACM MindTrek*. 9-15.
12. Dix, A., Finlay, J. E., Abowd, G. D. and Beale, R. (2003). *Human Computer Interaction*. Prentice-Hall.
13. Dong, T., Dontcheva, M., Joseph, D., and Karahalios, K., Newman, M. W., Acherman, M. S., (2012). Discovery-based Games for Learning Software, *ACM CHI*. 2083-2086.
14. Fernquist, J., Grossman, T. and Fitzmaurice, G. (2011). Sketch-sketch revolution: an engaging tutorial system for guided sketching and application learning. *ACM UIST*. 373-382.
15. Foster, S., Esper, S. and Griswold, W. (2013). From competition to metacognition: designing diverse, sustainable educational games. In *ACM CHI*. 99-108.
16. Gee, J.P. (2003). *What Video Games Have to Teach Us About Learning and Literacy*. Palgrave Macmillan.
17. Grossman, T., Fitzmaurice, G. (2010). ToolClips: an investigation of contextual video assistance for functionality understanding. *ACM CHI*. 1515-1524.
18. Grossman, T., Fitzmaurice, G. and Attar, R. (2009). A survey of software learnability: metrics, methodologies and guidelines. *ACM CHI*. 649-658.
19. Juul, J. (2009) Fear of failing? The many meanings of difficulty in video games. *The video Game Theory Reader 2*. New York: Routledge, 237-252.
20. Kelleher, C. and Pausch, R. (2005). Stencils-based tutorials: design and evaluation. *ACM CHI*. 541-550.
21. Lafreniere, B., Grossman, T., and Fitzmaurice, G. (2013). Community enhanced tutorials: improving tutorials with multiple demonstrations. *ACM CHI*. 1779-1788.
22. Levelup, <http://success.adobe.com/microsites/levelup>
23. Li, W., Grossman, T., and Fitzmaurice, G. (2012). GamiCAD: a gamified tutorial system for first time autocad users. *ACM UIST*, 103-112.
24. Mack, R. L., Lewis, C. H., and CaREPLAYoll, J. M. (1983). Learning to use word processors: problems and prospects. *ACM Trans Inf Syst*. 1, 254-271.
25. Matejka, J., Grossman, T. and Fitzmaurice, G. (2011). Ambient help. *ACM CHI*. 2751-2760.
26. Nielsen, J. (1994). *Usability Engineering*. Morgan Kaufmann.
27. Pagulayan, R., Keeker, K., Wixon, D., Romero, R. and Fuller, T. (2003). User-centered design in games. *The HCI Handbook: Fundamentals, Evolving Techniques and Emerging Applications*. 883-905.
28. Palmiter, S. and Elkerton, J. (1991). An evaluation of animated demonstrations of learning computer-based tasks. *ACM CHI*. 257-263.
29. Pongnumkul, S., Dontcheva, M., Li, W., Wang, J., Bourdev, L., Avidan, S. and Cohen, M. F. Pause-and-play: automatically linking screencast video tutorials with applications. *ACM UIST*. 135-144.
30. Prensky, M. (2001). *Digital Game-Based Learning*. McGraw-Hill.
31. Ramesh, V., Hsu, C., Agrawala, M. and Hartmann, B. (2011) ShowMeHow: translating user interface instructions between applications. *ACM UIST*. 127-134.
32. RibbonHero, <http://www.ribbonhero.com/>
33. Rosas, R., Nussbaum, M., Cumsille, P., Marianov, V. etc. (2003) Beyond Nintendo: design and assessment of educational video games for first and second grade students. *Computers & Education*, 40 71-94.
34. Steinkuehler, C. (2007). Massively multiplayer online games & education: an outline of research. *CSCL*, 675-685.
35. Tuck, R. and Olsen, D. R. (1990). Help by Guided Tasks: Utilizing UIMS Knowledge. *ACM CHI*. 71-78.
36. Twidale, M. (2005). Over the Shoulder Learning: Supporting Brief Informal Learning. *CSCW*, 505-547.
37. Vorderer, P. and Bryant, J. (2006). *Playing Video Games: Motives, Responses, and Consequences (Routledge Communication Series)*, Routledge