CS:Show – An Interactive Visual Analysis Tool for First-Person Shooter eSports Match Data

Robin Horst, Stefan Manuel Zander, Ralf Dörner

CS:Show – An Interactive Visual Analysis Tool for First-Person Shooter eSports Match Data*

Robin Horst[1], Stefan Manuel Zander[1], and Ralf Dörner[1]

RheinMain University of Applied Sciences, Wiesbaden, Germany
{Robin.Horst,Ralf.Doerner}@hs-rm.de

Abstract. Electronic Sports (eSports) is a fast-growing domain within the entertainment sector and becomes economically relevant in terms of a paying audience, merchandise, and major tournaments with highly endowed prize money. First-person shooter (FPS) games represent a dominant discipline. Professional training methodologies such as post-match analyses and tactics discussions are becoming essential in training sessions besides pure mechanical-oriented exercises such as aiming and movement. Furthermore, professional sports coaches are involved in the training of players. In this paper, we are investigating this newly developing profession, specifically, how multimedia systems can be built to support coaches and players in analyzing data of previous matches for preparing for future ones. In the example of Counter-Strike: Global Offensive (CS:GO), we identified a set of six criteria that can be incorporated into tools to support the analysis of FPS matches. We describe user interface functionalities that allow to interactively analyze the highly multivariate data of FPS matches. We show our concepts' technical feasibility by implementing them within a tool – *CS:Show*. Within an expert user study, evaluate our concepts with professionals. We conclude that our proposed eSports analysis tool was preferred over analysis functionalities built in in CS:GO. Supported by statistically significant evidence, our participants rated our tool more efficient, more usable, and assigned the tool with higher analytical ability than an average tool for analyzing FPS eSports matches.

Keywords: eSports · Information Visualization · Competitive Games · Match Analysis · Coaching Tools · First-Person Shooter · Visual Analytics · Counter-Strike: Global Offensive.

1 Introduction

Electronic Sports (eSports) is a developing industry and is growing rapidly in recent years [6]. Competitive gaming is becoming professional and profitable for teams, managers, and other stakeholders such as event and streaming hosts [12]. Furthermore, traditional sports clubs are already investing in eSports teams

* The work is supported by the Federal Ministry of Education and Research of Germany in the project Innovative Hochschule (funding number: 03IHS071).

[22]. There is also investment in the teams' professional training, for example, coaches who plan and lead the athletes' training sessions [4]. However, only little is scientifically investigated concerning the professional training of eSports athletes when it comes to using the plethora of multivariate data that can be obtained from past matches. It could be used to draw conclusions for future ones, for example, by analyzing upcoming opponents in a league, identifying weaknesses of the team, and developing new tactics.

In this paper, we investigate analyzing matches of first-person shooter (FPS) games and make the following contributions:

– We propose six criteria that allow professional coaches and players to analyze FPS matches systematically.
– In the example of Counter-Strike: Global Offensive (CS:GO), we show the feasibility of our criteria based on available eSports data and implement a match analysis tool – *CS:Show*. We employ the tool within a user study with CS:GO professionals. Based on our study's results, we evaluate our proposed tool and point out other functionalities that should be considered within the design of future eSports match analysis systems.

The paper is organized as follows. We briefly review related work in the next section. Then, we present six criteria we identified for analyzing FPS matches. After that, we describe how we designed and implemented our CS:Show tool. Before we conclude, we report and discuss our findings from our expert user study.

2 Related Work

The training of eSports athletes can easily occupy between 12 and 14 hours per day [18, 9]. However, Kari and Karhulahti [14] could point out that only less than half of this time on average is spent actively playing eSports games. That leaves a significant amount of training time for team meetings, review sessions, video analyses, strategic discussion, etc. [14].

Work by Snavely [21] states that coaches prepare such theory-oriented training sessions by considering past matches of both their own and opposing teams and analyze the players' abilities and habits, such as typical positioning, aim, angles that are observed, decision making, communication between players, among others. The results of their analyses are discussed in the following sessions with the players, where various aspects and situations are pointed out, reviewed, and visually supported by utilizing recorded demos and map-data. Finally, coaches try to develop overall guidelines and strategies with the players based on their analyses.

Overall, map-visualizations and demos are used within the theory-sessions emphasize the analyses and work out novel strategies with the players. A common task that coaches perform during these sessions is drawing rough layouts of player movements and potential routes onto visual map-representations using standard painting programs [7]. These drawings are performed iteratively based on the personal feedback of the players [2].

With regard to software tools that are used for training and analysis, besides painting and drawing programs, there also exist eSports-specific tools that aim at supporting coaches and players in developing new strategies and analyzing matches. Concerning FPS games and CS:GO, we identified tools such as Noesis [3], Skybox.gg [20], Scope.gg [19], AkiVer CS:GO Demo Manager [1], and demoanalyzer-go [16] as current examples. All of these tools support analysts (i.e., coaches or players) similarly to parse the contents of demo files and present the results to the analysts.

In the CS:GO community, the use of demos is established due to their capability of reviewing matches from different perspectives, including any player's perspective, and they usually have a smaller file size compared to video recordings [5]. However, the demo files and their data can only be used and reviewed using proprietary tools or functionalities built within the competitive games themselves. For example, the built-in CS:GO and Overwatch replay viewer provides basic functionalities for loading and reviewing recorded demos. However, they lack advanced analysis features. Providing the analysts with more in-depth information about their games and their opponents would allow them to learn more about their matches and perform better in future matches [15].

Except for demoanalyzer-go, a command-line program, the mentioned tools provide analysts with a graphical user interface (GUI). In terms of functionality, demoanalyzer-go is the only program that provides an automated ranking prediction based on the demo file's statistical data [16]. Noesis, Skybox.gg, and Scope.gg each provide a timeline where analysts can jump to single points in time and get the positions of players and metadata visualized [3, 20, 19]. A selection of multiple timeframes for further exploration [26] of the multivariate and temporal data is not provided. AkiVer CS:GO Demo Manager and demoanalyzer-go do not offer timelines [1, 16].

Noesis, Scope.gg, and CS:GO Demo Manager all feature heat maps for visualizing the data [3, 19, 1], although their implementations differ from tool to tool. They either visualize the players' kills or deaths. Scope.gg's implementation provides its heat maps based on zones, which are customarily defined areas for each map of CS:GO.

The AkiVer CS:GO Demo Manager, Noesis and Scope.gg and all feature a mini-map on which player positions are drawn [3, 19, 1]. On this map visualization, the players' view directions are visualized by lines indicating solely the direction or cones indicating the viewing angle of players. Compared to the 2D map representation of AkiVer CS:GO Demo Manager, Noesis, and Scope.gg, Skybox.gg does not provide a common top-down map view but a 3D view.

Finally, there exists substantial work that focuses on psychological aspects of eSports players (e.g., [11]) or economic and management perspectives (e.g., [8]). Results from related work concerning training and coaching practices and methodologies support the importance of theoretical preparation and the potential of using software tools to support coaches and players in it. However, we identified a research gap in this area since little scientific work was found. Within the investigation of existing tools for practitioners, we pointed out common functionalities that should be considered as a start for investigating tools for pre- and post-analyses of FPS matches.

3 FPS Match Analysis Criteria

In this section, we present six criteria we identified for analyzing FPS games. We do not claim to present a complete list of criteria, but an initial set that serves as a foundation for our FPS match analysis tool.

Criteria 1 – Location Awareness: A common objective in FPS games (e.g., CS:GO, Overwatch, Battlefield, etc.) is to target certain areas or objectives in the level [17]. One team must prevent the other team from accomplishing the objectives. A common tactic for the defending team is *camping*, where players would find hiding spots and wait there for the attacking (*rushing)* team to approach the objective [25]. Within match analysis procedures, it is crucial being able on the one hand to point out hiding spots of the defending team to anticipate their behavior and prepare for them when attacking, and on the other hand, to foresee which routes the attacking team will take to get to the objective to find the best hiding spots when defending.

Criteria 2 – Blind Spots: Due to the dominant shooting mechanics within FPS games, analyzing the vision can give advantageous insights into what areas are commonly watched by the opponents and avoid these areas [13]. This is particularly important in FPS games where single shots can be lethal (e.g., CS:GO) since the first shot may be the fatal one without the possibility to react to fire. Players in FPS games only have a certain view angle (in CS:GO fixed to 90°) so that blind spots exist (e.g., behind the player or far left/right of the viewing direction). Furthermore, even areas that lie within the current viewing direction of players might not be fully perceived. For example, players watch closer areas more thoroughly than farther areas. Information about these blind spots can be aggregated over certain time spans, for example, to provide analysts with information about which areas are more or less observed throughout a match.

Criteria 3 – Patience: Our third criteria, patience, is also related to the location of players. It describes the players' movement behavior in terms of a player's average resting time in one location before switching locations. Knowing the patience factor of specific players may enable coaches to develop tactics to counteract their actions. Patience is not only applicable to positions but also to the actions of players, for example, how frequently players perform weapon switching, jumping, or crouching, etc.

Criteria 4 – Aggressiveness: Aggressiveness is a criterion that measures how brisk a player acts. On the one hand, patience is a sub-aspect, for example, when a player has a low patience factor, a player may be attributed aggressive. But on the other hand, we incorporate another factor in aggressiveness that states how players react to friendly or opponent team members' actions. Such insight can give coaches information for deciding how enemy players can be disrupted and lured into actions. Furthermore, this criterion can be used to coach the own team and give advice on when to be more aggressive and when to stay calm, and carefully consider reactions to enemy activities such as shots fired or grenades thrown.

Criteria 5 – Weapons: The use of different weapons is another elementary mechanic in FPS games, with each weapon having various characteristics [24]. Different weapons can have different trade-offs, such as damage done vs. projectile range vs. accuracy vs. bullet spread, etc. The knowledge of what weapons are preferred by opponents can be used by coaches to make informed decisions on which weapons are suitable and advantageous to the enemies' tactics (e.g., sniper vs. close combat shotgun). However, such decisions are also dependent on the level design.

Criteria 6 – Utility: Our last criteria is the players' use of utilities. As utilities of FPS games, we summarize weapon mechanics and game objects that are not primary shooting or hitting weapons, such as grenades that inflict damage, grenades that impair the vision (e.g., flash grenade), grenades that impair the movement (e.g., stun grenades), or utilities that effect a whole area (e.g., flame (molotov) grenades or smoke grenades). The utility aspect includes when and where such utilities are used and give insight into their trajectories. The anticipation of such utilities can help coaches and players developing strategies to avoid the opponents' successful use of them, for example, avoiding particular routes where opponents regularly use such utilities or dodging them.

4 CS:Show Tool

In this section, we describe the GUI design of CS:Show – an FPS match analysis tool in the example of CS:GO.

Our tool's multi-view GUI consists of eight elements further described in the following. Figure 1 illustrates them. The (1) *top menu bar* at the top of the GUI (Fig. 1) provides system functionality such as loading a demo file into the tool, handling the multiple windows, or closing the application.

The (2) *player list* (Fig. 1 upper left) displays all players that were part of the match. It also illustrates their status with respect to the currently selected timeframe (in CS:GO called *tick*). For example, it reveals whether players are already dead at the current tick. Furthermore, the menu shows which team the players belong to and lets users select one or more players. Based on the player list selection, our tool provides additional information on demand about the current state within the (3) *statistics view* (Fig. 1 lower left).

The (4) *timeline* (Fig. 1 bottom) provides users with the ability to jump to a specific tick of the match, so that information about the players within the statistics view can display tick-specific information. Furthermore, multiple ticks can be selected as a coherent range of ticks (Fig. 1 bottom, orange bar within the timeline). This functionality enables our users to create a heat map from a selected time range of the match rather than only from the data of an

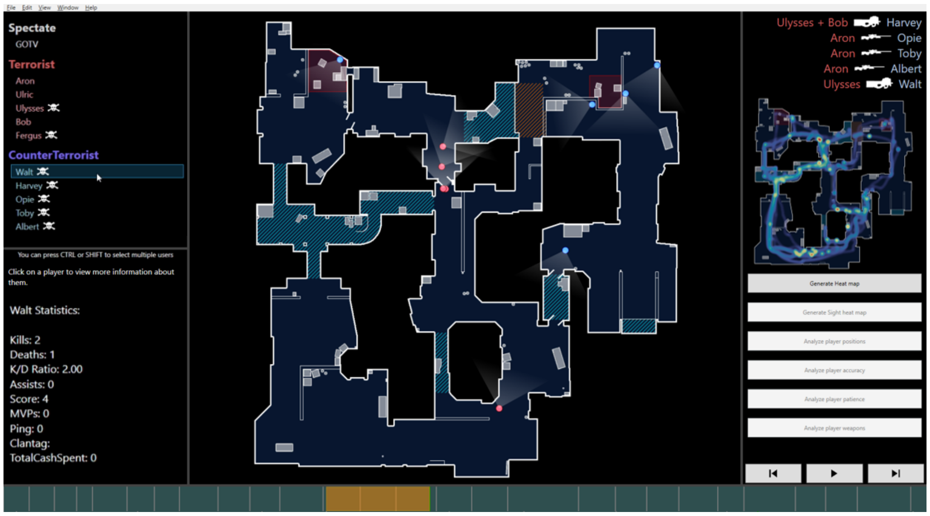


Fig. 1. A screenshot of the CS:Show GUI.

entire match. The timeline also can display important match events such as a round start, bomb plantings, etc., directly on the timeline. For example, the grey strokes in Fig. 1 show the start of rounds to help users get an overview of the data.

The (5) *mini-map view* (Fig. 1 middle) illustrates outlines and important zones of a certain CS:GO level such as bomb or hostage zones and player spawns. Furthermore, it visualizes the players' positions at the current tick of the match and shows players' view frustums to illustrate potential blind spots at a given time.

The (6) *kill feed* shows the sequence of a round's kills. Pictographs visualize which weapon was used for the kill. The order of the player names encodes who was killed by whom similarly to the in-game visualization of kills in CS:GO.

The (7) *heat map view* displays a small version of the mini-map with a heat map overlay. This gives users a glimpse in the match data over the selected range of ticks and the selected players concerning a specific aspect such as the positioning, death, vision, etc. This view can be detached, for example, to view it on a second monitor to see both the mini-map view and the heat map view simultaneously during the analysis process. Furthermore, it can also be switched with the mini-map view so that the heat map is displayed in the middle and the mini-map is displayed smaller and on the side.

Finally, the (8) *meta tools* allow users of our tool to alter the current mini-map or heat map view. For example, drawing tools are provided to draw lines, basic forms, or freehand lines to prepare a specific tick for discussing it within a review session with the team later. Besides drawing-related meta tools, we also provide meta tools that guide users by analyzing specific match aspects and criteria such as player accuracy, patience, or fine-grained heat map adjustments.

The latter includes altering the brush stroke size of heat maps. Fine-grained brush strokes help to analyze the individual players' movements (Fig. 2 left), whereas a coarse brush stroke size is suitable for providing a general overview of the map and its hot spots (Fig. 2 right).

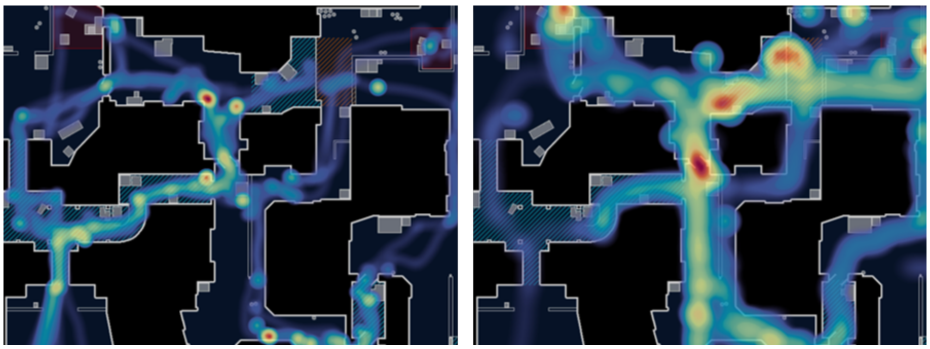


Fig. 2. Two situations with different heat map brush stroke sizes.

We used C# and the GUI framework Windows Presentation Foundation (WPF) to implement the CS:Show visual analysis tool. Each of the mentioned elements was implemented as separate WPF user control to preserve their reusability in future versions of the tool. For drawing functionality such as the heat map generation, we utilized a native heat map library exported as DLL and called via Platform Invocation (P/Invoke).

5 Evaluation

Within an expert user study, we evaluated our proposed CS:Show tool. The study involved ten unpaid and voluntary participants (aged between 16 and 32 years with Ø 22.90 and SD 4.65). They were recruited from professional service providers (e.g., Fiverr and social media platforms (e.g., Reddit). We selected them based on their subjective classification as professional CS:GO coaches and players and objective skill measures (e.g., competitive CS:GO ranking as Global Elite (highest possible rank), total CS:GO playtime, or currently being a professional CS:GO team's coach.). The user study was conducted as a moderated remote study using Discord.

The procedure of the study took place as follows. Firstly, participants were welcomed and then informed about the topic of the study. We introduced them to the UI of the tool and the process of the evaluation. Then the actual task phase of the study started. In this phase, our participants were asked to perform nine tasks with the tool, such as familiarizing with the tool for several minutes, loading and analyzing demos, and drawing conclusions on tactical aspects that could be derived with the tool. Finally, we asked our participants to fill out a questionnaire.

We evaluated four aspects with this questionnaire:

[A1] *Ease of use*: How usable is our tool?
[A2] *Analysis ability*: How well enables our tool analysts to conduct analyses based on the provided functionalities?
[A3] *Efficiency*: How well does the analysis approach with our tool justifies the time it takes?
[A4] *Product character*: The product character [10] is a measure incorporating both pragmatic and hedonic qualities.

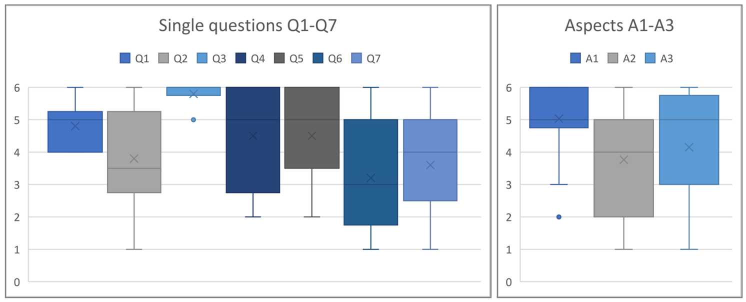
Relating to the product character (A4), we utilized the abbreviated AttrakDiff questionnaire [23] as an established tool for measuring it. Aspects 1-3 were captured with seven questions Q1-Q7 on a 7-point semantic differential scale. They were concerned with (Q1) the GUI quality, (Q2) the analysis ability of our tool compared to other analysis procedures, (Q3) the performance, (Q4) the task efficiency, (Q5) the heat map visualization, (Q6) the displayed statistics, and (Q7) the analysis of player behavior. The questions were clustered to the aspects 1-3, whereas Q1, Q3, and Q4 were clustered to A1, Q5, Q6, and Q7 to A2, and Q4 and Q2 to A3. Finally, space for written comments and demographic questions concluded the questionnaire. A single session of the study was performed within a roughly one hour timeframe.

5.1 Analysis of the Results

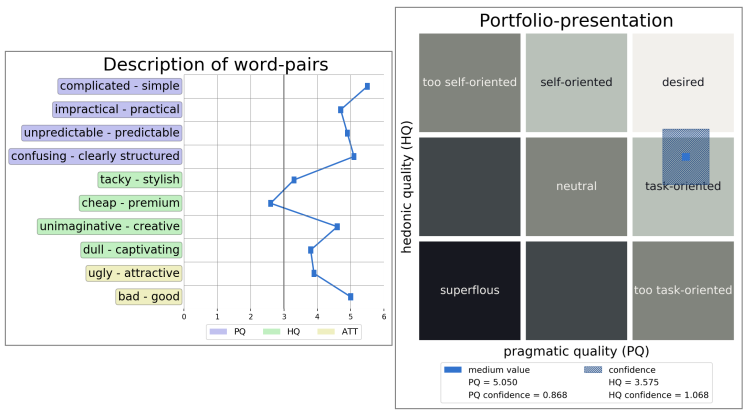
Figure 3a (left) represents the value distributions of the single items Q1-Q7. The box-whisker plots show that the mean values of all single questions lie above the hypothetical neutral value of 3. Wilcoxon signed-rank tests were conducted on the items to analyze how CS:Show was rated by the participants compared to a neutral rating. With a threshold for statistical significance of 5%, the tests for Q1, Q3, Q4, and Q6 did confirm statistically significant differences (Tab. 1). Furthermore, Figure 3a (right) shows the value distributions of the three observed aspects 1-3. Again, all mean values lie above 3. Further Wilcoxon signed-rank tests were conducted to test the outcome for the aspects against a neutral rating. All tests confirm significant differences.

The written comments from the questionnaire, observations, and oral statements during the study were used to capture additional information and were assigned to A1-A3. Concerning A1, we noticed difficulties creating the heat map. For example, two out of ten participants failed to select the players from the player list before creating the heat map. Furthermore, one participant did not figure out that a time span selection on the timeline must be made prior as well. Finally, several minor negative comments were given about the prototypical look of the interface.

Concerning A2, our participants proposed to exclude the *freeze time* from the heat map generation (the time in which players spawn and are unable to move



(a)



(b)

Fig. 3. (a) Box-whisker plots for the single questions 1-6 and the aggregated aspects 1-3. (b) Description of word-pairs and portfolio-presentation of the AttrakDiff values.

to give the teams sufficient time to communicate purchasing gear). Furthermore, they suggested also providing heat maps for player deaths, kills, and *entry frags* (first kill in a round). Our participants also noted that the money players carry and have spent would allow the analysis of an individual player's economy. Finally, one participant asked for insights in data about flashbang grenades, for example, at which times players threw such grenades, which areas were affected, and which players were blinded by them.

Concerning A3, six out of ten stated that they would prefer CS:Show over the built-in demo viewer they often use and found our tool more efficient. Still, they also stated that the workflow would benefit from using drag-and-drop actions, for example, for loading demo files. Furthermore, it was suggested to add fast-forward playback functionality to speed up finding specific demo sections. Some participants also noted that zooming into both the timeline and the map visualization would be beneficial. Finally, it was proposed to add functionality to enter a particular tick by its number to jump to it directly rather than navigating to it using the timeline.

Questions / aspects	Ø-values	SD	P-values
Q1	4.8	0.7888	$p = 0.0020$
Q2	3.8	1.6865	$p = 0.1563$
Q3	5.8	0.4216	$p = 0.0020$
Q4	4.5	1.5811	$p = 0.0195$
Q5	4.5	1.5092	$p = 0.0195$
Q6	3.2	1.7512	$p = 0.7422$
Q7	3.6	1.6465	$p = 0.3828$
A1	5.0333	1.1592	$p \leq 0.0001$
A2	3.7667	1.6750	$p = 0.0230$
A3	4.1500	1.6311	$p = 0.0042$

Table 1. Mean values, SD, and output of the Wilcoxon signed-rank tests for Q1-Q7 and A1-A3. All values are rounded to four decimal places.

The outcome of the AttrakDiff questionnaire was analyzed concerning the product character of CS:Show (A4). The portfolio-presentation (Fig. 3b right) shows that CS:Show was placed within the graph's 'task-oriented' region. The square is shifted towards the 'desired' area, and the confidence rectangle overlaps slightly with it. The word-pair visualization (Fig. 3b left) shows that all mean values lie above 3 and thereby on the positive side of the graph, except 'cheap–premium', which lies between 2 and 3.

5.2 Discussion of the Results

The evaluation results show that professional eSports coaches and players could successfully use CS:Show to analyze CS:GO matches. The ease of use (A1) of our tool was rated the most positive of the three aspects. The overall positive perception is supported by statistical significance and the AttrakDiff evaluation. However, both the medium deviation of Q1 and the AttrakDiff items 'ugly–attractive' and 'cheap–premium' suggest improving the visual GUI quality in future versions of our tool.

Concerning the analysis ability (A2), the results indicate that our participants could develop novel strategies with our tool and that the visualizations within our heat map we provided were positively accepted. This claim is also backed by statistical significance. However, our participants also suggested several aspects that would be beneficial, which we did not include in CS:Show. Furthermore, we could also not find the mentioned functionality in most of the analyzed tools in Section 2. This indicates that improvements such as economic- and utility-related aspects should be investigated in future work.

The efficiency (A3) was also rated positively. While this could also be backed by statistical significance, our participants noted that our tool would be more efficient than existing tools when including the additional functions mentioned in A2. Q2's deviation range also supports this claim.

Finally, concerning our tool's product character (A4), the AttrakDiff results support the high usability of our tool with the pragmatic qualities items. Still, hedonic qualities could be improved. For example, re-implementing the tool with modern web GUI frameworks such as vue.js or react might help creating a more contemporary look and providing subtle visual improvements such as fade- and button animations and layout styles.

6 Conclusion and Future Work

In this paper, we investigated interactive eSports FPS match analysis procedures. In the example of CS:GO, we introduced six criteria that can be incorporated in tools to support the analysis of FPS matches. We described a suitable GUI and aspects concerning the implementation and provision of the criteria concepts for coaches and players within an interactive analysis tool – CS:Show. Based on our user study's results with professional eSports coaches and players and backed by statistical evidence, we conclude that CS:Show could be used successfully by our participants. Furthermore, we pointed out novel features that should be included in future match analysis tools.

Future work should investigate the novel identified features and how CS:Show can include them. Furthermore, based on our work, it should be explored how match data from other FPS eSports games can be included within our analysis tool. We have shown the feasibility in the example of CS:GO, but other competitive FPS games such as Overwatch, Rainbow Six Siege, Call of Duty, Battlefield, etc., might be analyzed using our criteria as well. Finally, the analysis of matches from different FPS games might also bring up novel criteria that can be used to support CS:GO coaches and players. This way, a comprehensive pool of FPS analysis criteria that experts can choose of establishes, and the growing eSports field will further professionalize and advance.

References

1. Akiver: Akiver analysis tool. `https://github.com/akiver/CSGO-Demos-Manager` (2021), accessed: July 29, 2021
2. Andreasen, M.S., Caspersen, T.B.: The human factor in esport: Esport psychology. BoD–Books on Demand (2019)
3. Bang & Jensen ApS: Noesis analysis tool. `https://www.noesis.gg/` (2021), accessed: July 29, 2021
4. Bányai, F., Griffiths, M.D., Király, O., Demetrovics, Z.: The psychology of esports: A systematic literature review. Journal of gambling studies **35**(2), 351–365 (2019)
5. Bednárek, D., Kruliš, M., Yaghob, J., Zavoral, F.: Data preprocessing of esport game records. In: Proceedings of the 6th International Conference on Data Science, Technology and Applications. SCITEPRESS-Science and Technology Publications, Lda. pp. 269–276 (2017)
6. Block, S., Haack, F.: esports: a new industry. In: SHS Web of Conferences. vol. 92. EDP Sciences (2021)

7. Fanfarelli, J.R.: Expertise in professional overwatch play. International Journal of Gaming and Computer-Mediated Simulations (IJGCMS) **10**(1), 1–22 (2018)
8. Hallmann, K., Giel, T.: esports–competitive sports or recreational activity? Sport management review **21**(1), 14–20 (2018)
9. Harrison Jacobs: Here's the insane training schedule of a 20-something professional gamer. `https://www.businessinsider.com/pro-gamers-explain-the-insane-training-regimen-they-use-to-stay-on-top-2015-5` (2015), accessed: July 29, 2021
10. Hassenzahl, M.: The thing and i: understanding the relationship between user and product. In: Funology 2, pp. 301–313. Springer (2018)
11. Hung, J.C., Lin, Z.Q., Huang, C.H., Lin, K.C.: The research of applying affective computing based on deep learning for esports training. In: International Conference on Frontier Computing. pp. 122–129. Springer (2019)
12. Jenny, S.E., Keiper, M.C., Taylor, B.J., Williams, D.P., Gawrysiak, J., Manning, R.D., Tutka, P.M.: esports venues: A new sport business opportunity. Journal of Applied Sport Management **10**(1), 8 (2018)
13. Jordan, C.: Information Models in Multiplayer Gaming: Teaching New Players the Complex In-game Economy of Counter-Strike: Global Offensive. Ph.D. thesis, Auckland University of Technology (2020)
14. Kari, T., Karhulahti, V.M.: Do e-athletes move?: A study on training and physical exercise in elite e-sports. International Journal of Gaming and Computer-Mediated Simulations (IJGCMS) **8**(4), 53–66 (2016)
15. Larsen, L.J.: The play of champions: Toward a theory of skill in esport. Sport, Ethics and Philosophy pp. 1–23 (2020)
16. Quancore: demoanalyzer-go analysis tool. `https://github.com/quancore/demoanalyzer-go` (2021), accessed: July 29, 2021
17. Rayner, D.C.F.: Analysing openings in tactical simulations. Ph.D. thesis, University of Alberta (2008)
18. Rich Stanton: The secret to esports athletes' success? lots – and lots – of practice. `https://www.espn.com/espn/story/_/id/13053116/esports-athletes-put-hours-training-reach-pinnacle` (2015), accessed: July 29, 2021
19. Scope.gg. CS:GO Analytics.: Scope.gg analysis tool. `https://scope.gg/` (2021), accessed: July 29, 2021
20. Skybox Technologies ApS: Skybox analysis tool. `https://landing.skybox.gg/` (2021), accessed: July 29, 2021
21. Snavely, T.L.: History and analysis of eSport systems. Ph.D. thesis, University of Texas at Austin (2014)
22. Ströh, J.H.A.: The eSports market and eSports sponsoring. Tectum Wissenschaftsverlag (2017)
23. User Interface Design GmbH: Attrakdiff questionnaire. `http://attrakdiff.de/index-en.html` (2021), [Accessed: July 29, 2021]
24. Vorderer, P., Bryant, J.: Playing video games: Motives, responses, and consequences. Routledge (2012)
25. Weber, R., Behr, K.M., Tamborini, R., Ritterfeld, U., Mathiak, K.: What do we really know about first-person-shooter games? an event-related, high-resolution content analysis. Journal of Computer-Mediated Communication **14**(4), 1016–1037 (2009)
26. Yi, J.S., ah Kang, Y., Stasko, J., Jacko, J.A.: Toward a deeper understanding of the role of interaction in information visualization. IEEE transactions on visualization and computer graphics **13**(6), 1224–1231 (2007)