The Effect of Fitts’s Law on Timing Sensitive Tasks, and the Feasibility of Using Osu! for HCI Research

Fitts’s Law in Osu!

This paper explores the relationship between the index of difficulty given by Fitts’s law and timing accuracy using data from the rhythm game Osu!, and discusses the potential of using Osu! for future research into human-computer interaction.

Maxwell C. Nieberger

Colorado State University, mxnieberger@gmail.com

Although there is no distinctive header, this is the abstract. This submission template allows authors to submit their papers for review to an ACM Conference or Journal without any output design specifications incorporated at this point in the process. The ACM manuscript template is a single column document that allows authors to type their content into the pre-existing set of paragraph formatting styles applied to the sample placeholder text here. Throughout the document you will find further instructions on how to format your text.

CCS CONCEPTS • Insert your first CCS term here • Insert your second CCS term here • Insert your third CCS term here

**Additional Keywords and Phrases:** Insert comma delimited author-supplied keyword list, Keyword number 2, Keyword number 3, Keyword number 4

ACM Reference Format:

First Author’s Name, Initials, and Last Name, Second Author’s Name, Initials, and Last Name, and Third Author’s Name, Initials, and Last Name. 2018. The Title of the Paper: ACM Conference Proceedings Manuscript Submission Template: This is the subtitle of the paper, this document both explains and embodies the submission format for authors using Word. In Woodstock ’18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY. ACM, New York, NY, USA, 10 pages. NOTE: This block will be automatically generated when manuscripts are processed after acceptance.

1. Introduction

Fitts’s law is a well-established relationship between the distance a cursor is from a target, the size of the target, and the time required to move the cursor to the target. Osu! is a rhythm game which requires players to click on circular targets appearing at scripted locations on the screen in time with the beat of a song. Despite the obvious relevancy, studies utilizing Osu! for HCI research are difficult to find, to the point of being nonexistent. This study uses replays made publicly available by Osu! to study the effect of difficulty as described by Fitts’s law on a user’s ability to time notes effectively, and analyzes the potential for Osu! to be used in future HCI studies.

* 1. Osu!

Osu! (here styled “osu” for simplicity) was initially released for Windows computers in 2007 and has seen widespread success since then. It has ports in development for all major operating systems, is actively maintained by the developers, and has an active playerbase which sees several million unique users each month. In total, this playerbase has amassed tens of billions of song plays, many of which have been recorded through osu’s built in replay system. For every song that has public rankings, replays of the top 50 scores achieved are publicly available to download and they can be parsed by a number of open-source tools to make use of that data.

1. Procedure

For this study I looked for a song from the most recent seasonal spotlight pack, from summer 2020. I decided to use the “Alazy” difficulty of “Sound Horizon - Raijin no Hidariude” (https://osu.ppy.sh/beatmapsets/16792#osu/64589) since it is one of the more popular songs in the pack with over 2 million combined plays, and because it has a much higher number of hit circles compared to sliders making it easier to process. From the webpage for that beatmap I downloaded the replays of the top 50 scores. Using the osrparse python library, I was then able to parse these replay files into their metadata and a list of every action taken by the player over the course of the play. In order to calculate performance for each note of the song, I had to take this parsed replay data and compare it to the list of hit objects in the beatmap itself. Osu beatmaps are stored in a human-readable format, so I was able to add a function to my python script that extracts the location and timing of every target, but sliders required extra processing.

In osu, sliders consist of an initial hit circle followed by a path which the player must trace at a certain rate determined by the beatmap configuration. Since path tracing wasn’t relevant to my investigation, I initially thought to simply exclude all sliders, but that caused additional issues. For this study, I was interested in 5 categories of information for each note; when the note was in the song (time), the distance between that note and the previous one (distance), the time between that note and the previous one (period), the time delta between the actual time of the note and when the player hit it (timing error), and how far the player’s hit was from the center of the note (accuracy error). When excluding sliders from my calculations it changed both the distance and period for the following note. To correct this, I opted to include the start of each slider as though it is a normal hit circle, then calculate the endpoint of the slider to use for the distance and period of the next note. This ended up being the most time-consuming part of the coding process, but I eventually found a method that appeared to match the timings shown in game.

The next problem I encountered was with processing mods. Osu has a number of built-in modifiers which change things such as the speed and difficulty of a beatmap. Since these mods also affect the timing windows of hit circles, they resulted in my program assuming the wrong input was when the note was hit, giving some strange results. The parsing library I used already included which mods were used for each replay, so I added code to correct for the main types of mods used. While this was a minor improvement, unfortunately mods that change the rate of the song still caused the program to assume every note was missed and I was forced to exclude any replays using rate mods from the data. I ended up excluding 7 replays for this reason out of a total of 50.

After these corrections, I was able to parse the vast majority of notes with results that appear reasonably accurate. There were still some inaccuracies in my program, however, since it reported a larger number of notes missed than were listed on the official leaderboards. I believe these discrepancies are either the result of differing rounding behavior between my program and the official game, resulting in some inputs being incorrectly counted as in or out of the timing window, or because the operations I performed to find key presses when parsing the list of player actions had a corner case that meant switching between two keys frame-perfectly wouldn’t register as a new key press. This resulted in 85 notes being incorrectly excluded as misses, with a total of 14,854 notes parsed successfully.

1. Results

I performed a regression test to see how each of the statistics I gathered related to the timing error of a note, including the interactions between each pair of statistics, and calculated the correlations for every pair.

Correlation coefficients

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Timing Error | distance | period | time | accuracy error |
| Timing Error | 1 |  |  |  |  |
| distance | -0.060894091 | 1 |  |  |  |
| period | 0.030193832 | 0.242510236 | 1 |  |  |
| time | 0.014404137 | 0.119250052 | -0.036695661 | 1 |  |
| accuracy error | 0.097644383 | 0.075075245 | 0.003730605 | 0.03508911 | 1 |

Of the measured statistics, the interaction between distance and pointer accuracy was by far the most significant factor influencing timing error (p = 9.94524E-36).

Both of these factors also contributed to timing error individually. Pointer accuracy had the greatest effect on timing accuracy by a large margin (0.283268466) and to a high degree of certainty (p = 1.8986E-15). Distance also had an effect with a high degree of confidence (p = 3.96707E-12), but to a much smaller degree.

The interactions between time and accuracy, and between time and distance also both contributed (p = 2.89979E-09, and p = 3.03087E-07 respectively), though that may be inflated by the importance of distance and accuracy.

Summary output, all statistics

| Regression Statistics | |
| --- | --- |
| Multiple R | 0.178780537 |
| R Square | 0.03196248 |
| Adjusted R Square | 0.031310296 |
| Standard Error | 8.18196786 |
| Observations | 14854 |

ANOVA, all statistics

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | *df* | *SS* | *MS* | *F* | *Significance F* |
| Regression | 10 | 32808.43454 | 3280.843454 | 49.00833747 | 2.67245E-97 |
| Residual | 14843 | 993658.669 | 66.94459806 |  |  |
| Total | 14853 | 1026467.104 |  |  |  |

Regression results sorted by P-value, all statistics

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Coefficients | Standard Error | t Stat | P-value |
| Intercept | 5.769318978 | 0.445448234 | 12.95171592 | 3.6983E-38 |
| distance \* accuracy | -0.002354229 | 0.000188185 | -12.51017518 | 9.94524E-36 |
| accuracy error | 0.283268466 | 0.035602625 | 7.956392735 | 1.8986E-15 |
| distance | 0.020232072 | 0.002913564 | 6.944096332 | 3.96707E-12 |
| time \* accuracy | 2.49494E-06 | 4.19967E-07 | 5.940799812 | 2.89979E-09 |
| distance \* time | -1.62942E-07 | 3.18006E-08 | -5.123866945 | 3.03087E-07 |
| period | -0.00436362 | 0.001380803 | -3.160205292 | 0.001579747 |
| period \* time | 6.24452E-08 | 2.14817E-08 | 2.906905529 | 0.003655635 |
| distance \* period | 7.62258E-06 | 2.82605E-06 | 2.697254875 | 0.006999269 |
| time | -1.02073E-05 | 5.53529E-06 | -1.844044357 | 0.065196593 |
| period \* accuracy | -2.70042E-05 | 3.23596E-05 | -0.834504146 | 0.404010383 |

Period, time itself, and period’s interactions with each of the other stats all had p-values well above those of the stats already mentioned (p > 0.001). Because of this, I re-ran the regression excluding these stats and again found that all of the previously mentioned stats had an effect on timing error in a statistically significant way (F = 1.35127E-89)

Summary output, reduced statistics

|  |  |
| --- | --- |
| Regression Statistics | |
| Multiple R | 0.167929414 |
| R Square | 0.028200288 |
| Adjusted R Square | 0.027873039 |
| Standard Error | 8.196471277 |
| Observations | 14854 |

ANOVA, reduced statistics

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | df | SS | MS | F | Significance F |
| Regression | 5 | 28946.66799 | 5789.333599 | 86.17369851 | 1.35127E-89 |
| Residual | 14848 | 997520.4355 | 67.1821414 |  |  |
| Total | 14853 | 1026467.104 |  |  |  |

Regression results sorted by P-value, reduced statistics

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Coefficients | Standard Error | t Stat | P-value |
| Intercept | 4.859122822 | 0.230348425 | 21.0946648 | 2.37463E-97 |
| distance \* accuracy | -0.002375571 | 0.000180763 | -13.14192028 | 3.13497E-39 |
| accuracy error | 0.289913627 | 0.030700726 | 9.443217369 | 4.14309E-21 |
| distance | 0.021760159 | 0.002470419 | 8.808286935 | 1.40943E-18 |
| time \* accuracy | 2.32443E-06 | 3.03012E-07 | 7.67106406 | 1.81144E-14 |
| distance \* time | -1.30361E-07 | 2.25972E-08 | -5.768909823 | 8.13755E-09 |

Overall, all of these statistics only accounted for a tiny proportion of the variance in timing error, with an Adjusted R Square of 0.027873039.

* 1. Analysis

My hypothesis was that because of Fitts’s law, the distance between one note and the next would have an impact on a player’s ability to time that note correctly. While this did turn out to be the case, I was surprised to find that accuracy had a much stronger effect on timing error and that the period of time to hit the note had very little effect. After further consideration, I realized accuracy is also closely related to Fitts’s law in regards to the size of the target. A worse accuracy is equivalent to hitting a larger target, making it easier.

This also explains the negative coefficient for the interaction between accuracy error and distance. Because the targets are circular and both accuracy and distance are measured from the center, accuracy error will almost always be smaller than distance with the only exceptions being when the pointer starts over the target and the player moves away from the center in anticipation of the following note. This means increases in accuracy error have a larger effect on the interaction with distance than increases in distance do, so said interaction is more closely related to the inverse of the difficulty of a note than the actual difficulty itself. Larger values for the interaction mean the note was easier to hit which correlates with a smaller timing error giving a negative coefficient. I attempted to further explore this relationship by adding the ratio of distance to accuracy error as described by Fitts’s law to the regression, however several data points with an abnormally low accuracy error made it impossible to draw conclusions from the results.

More difficult to explain is why period had such an insignificant effect on the results, since Fitts’s law relates to the time required to move to a target. There are a few likely causes for this, the first is that the period is simply very large compared to the actual time required to hit a note. This could have been amplified by the source of the data collected, since only replays for the best 50 players’ scores were available and any player included in that list would be capable of playing at extremely high speeds. It’s also possible that the rhythmic nature of the task had an effect. Instead of a continuous range of periods, the period between notes typically varies in discrete fractions of a beat and that predictability may have reduced any effect that different periods may have had.

I included time in the calculations to see if notes later into the song had higher errors, possibly due to fatigue or nerves. The fact that timing itself had little effect, but its interactions with distance and accuracy had much more of an effect is interesting. That result suggests that even though being deep into a song doesn’t change a player’s ability to time notes directly, it does amplify the effects of other sources of difficulty. It’s important to note, however, that there is also a slight correlation between distance and time which indicates the song itself gets more difficult towards the end.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  | |

Figure 1: Residual plots from regression with reduced statistics

Graphs of the residuals from the regression with some stats removed show some artifacts which should be addressed. The residuals of distance and its interaction have a noticeable spike for low values which I believe is caused by overlapping circles in the beatmap. The program I wrote to parse the data doesn’t treat these overlaps any differently to more spread-out circles, but when multiple circles overlap there is the unique possibility to hit them without moving the pointer. In this situation the difficulty shifts from accuracy to correctly reading the rhythm when the visual indicators of that rhythm are densely packed and hard to distinguish, something that clearly isn’t accounted for by Fitts’s law.

The residuals for accuracy error also have an artifact in the form of a loose secondary line above the main grouping of the data. This indicates there are additional factors influencing this data which I failed to capture with the statistics I chose. Accuracy error is calculated as the distance between the point the note is hit and the center of the note. It doesn’t account for the direction of the error, the direction the pointer moved to reach the target, or the direction of the next note, any of which could influence the results.

* 1. Using Osu! as a research tool

The study I performed was extremely limited in time and scope, and the Covid-19 pandemic greatly limited the types of testing and data collection available to me. By using osu and open-source parsing libraries I was able to gather a large amount of data quickly and easily, allowing me to get results much stronger than I would have been able to using other possible methods. There are, however, some issues introduced by this method of data collection to take into consideration.

The biggest potential issue is that the data available is only representative of a very small subset of the population. The only replays available for download are the best scores by the top 50 players, which is tiny in the context of a song like the one used in this study which has been played over 2 million total times. Players at such an extreme end of the talent distribution often have equipment chosen specifically to improve their abilities, unique quirks which give them an advantage, and much more experience than the average player which all serve to make them less representative of the total playerbase and the population as a whole.

Another potential issue is the limited information for anything outside a player’s direct actions during the song. There’s no way to check what system hardware they used, what input method, their posture, the environment they played in, etc. all of which can affect the data. In rare circumstances the players themselves may have information on the profile associated with a replay, but that is much more difficult to retrieve and too inconsistent to be useful at a large scale. Related to this is the fact that only a player’s best score is available. It’s extremely difficult, if not impossible, to determine how much a player practiced the song before setting the score in the leaderboards.

Despite these drawbacks, osu still provides an incredible amount of data which hasn’t been studied to a meaningful extent, and the platform itself can be used in more controlled studies instead of using proprietary software that requires additional time or money to create and use. This study only involved a single beatmap of a single song, but there are tens of thousands of beatmaps across tens of thousands of songs which could be used in future studies. Also, the community in general has a number of active forums and social media sites that could be used to recruit users for studies of a wider slice of the population. From cursory browsing of these sites, saving replays appears to be relatively common for individuals outside of the ones made available on the leaderboards so asking for contributions may be an effective way to gather even more data.

Osu also has potential for testing new interaction devices, since it has good support for both traditional mouse and keyboard and many forms of drawing tablets and touch screens. The data already available could serve as a reasonable baseline to compare against, and the built-in replay system already records all user inputs for the duration of a song. For controlled studies, osu also has a beatmap editor that can be used to create custom beatmaps accompanied by any audio and video, allowing researchers to precisely control the tests they present to study participants.

1. Conclusion

The distance and size of a note in osu does have an effect on a player’s ability to correctly time that note in a way that coincides with Fitts’s Law. However, this effect is dwarfed by other sources of variability which couldn’t be determined from the data obtained. In the course of this study, I was able to use data made available by osu with freely available tools to extract useful statistics about user actions. This proves osu is a useful resource for research into Human-Computer Interactions, and opens up a large number of avenues for future study.

References are a placeholder

List of Fitts’s Law related studies: <https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/fitts-s-law>

Osu! Song page:

<https://osu.ppy.sh/beatmapsets/16792#osu/64589>

Osu! Replay parser:

<https://github.com/kszlim/osu-replay-parser>

Osu! Beatmap format:

<https://osu.ppy.sh/wiki/sk/osu!_File_Formats/Osu_(file_format)>

REFERENCES

1. Atul Adya, Paramvir Bahl, Jitendra Padhye, Alec Wolman, and Lidong Zhou. 2004. A multi-radio unification protocol for IEEE 802.11 wireless networks. In Proceedings of the IEEE 1st International Conference on Broadnets Networks (BroadNets’04) . IEEE, Los Alamitos, CA, 210–217. https://doi.org/10.1109/BROADNETS.2004.8
2. Sam Anzaroot and Andrew McCallum. 2013. UMass Citation Field Extraction Dataset. Retrieved May 27, 2019 from <http://www.iesl.cs.umass.edu/data/data-umasscitationfield>
3. Martin A. Fischler and Robert C. Bolles. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24, 6 (June 1981), 381–395. https://doi.org/10.1145/358669.358692
4. Chelsea Finn. 2018. Learning to Learn with Gradients. PhD Thesis, EECS Department, University of Berkeley.
5. Jon M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. J. ACM 46, 5 (September 1999), 604–632. https://doi.org/10.1145/324133.324140
6. Matthew Van Gundy, Davide Balzarotti, and Giovanni Vigna. 2007. Catch me, if you can: Evading network signatures with web-based polymorphic worms. In Proceedings of the first USENIX workshop on Offensive Technologies (WOOT ’07) . USENIX Association, Berkley, CA, Article 7, 9 pages.
7. James W. Demmel, Yozo Hida, William Kahan, Xiaoye S. Li, Soni Mukherjee, and Jason Riedy. 2005. Error Bounds from Extra Precise Iterative Refinement. Technical Report No. UCB/CSD-04-1344. University of California, Berkeley.
8. David Harel. 1979. First-Order Dynamic Logic. Lecture Notes in Computer Science, Vol. 68. Springer-Verlag, New York, NY. <https://doi.org/10.1007/3-540-09237-4>
9. Jason Jerald. 2015. The VR Book: Human-Centered Design for Virtual Reality. Association for Computing Machinery and Morgan & Claypool.
10. Prokop, Emily. 2018. The Story Behind. Mango Publishing Group. Florida, USA.
11. R Core Team. 2019. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/
12. Brian K. Reid. 1980. A high-level approach to computer document formatting. In Proceedings of the 7th Annual Symposium on Principles of Programming Languages. ACM, New York, 24–31. <https://doi.org/10.1145/567446.567449>
13. John R. Smith and Shih-Fu Chang. 1997. VisualSEEk: a fully automated content-based image query system. In Proceedings of the fourth ACM international conference on Multimedia (MULTIMEDIA ’96). Association for Computing Machinery, New York, NY, USA, 87–98. https://doi.org/10.1145/244130.244151
14. TUG 2017. Institutional members of the LaTeX Users Group. Retrieved May 27, 2017 from <http://wwtug.org/instmem.html>
15. Alper Yilmaz, Omar Javed, and Mubarak Shah. 2006. Object tracking: A survey. ACM Comput. Surv. 38, 4 (December 2006), 13–es. https://doi.org/10.1145/1177352.1177355

Patricia S. Abril and Robert Plant. 2007. The patent holder's dilemma: Buy, sell, or troll? Commun. ACM 50, 1 (Jan. 2007), 36-44. DOI: <https://doi.org/10.1145/1188913.1188915>

Sarah Cohen, Werner Nutt, and Yehoshua Sagic. 2007. Deciding equivalences among conjunctive aggregate queries. J. ACM 54, 2, Article 5 (April 2007), 50 pages. DOI: https://doi.org/10.1145/1219092.1219093

David Kosiur. 2001. Understanding Policy-Based Networking (2nd. ed.). Wiley, New York, NY.

Ian Editor (Ed.). 2007. The title of book one (1st. ed.). The name of the series one, Vol. 9. University of Chicago Press, Chicago. DOI:https://doi.org/10.1007/3-540-09237-4

Donald E. Knuth. 1997. The Art of Computer Programming, Vol. 1: Fundamental Algorithms (3rd. ed.). Addison Wesley Longman Publishing Co., Inc.

Sten Andler. 1979. Predicate path expressions. In Proceedings of the 6th. ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (POPL '79), January 29 - 31, 1979, San Antonio, Texas. ACM Inc., New York, NY, 226-236. <https://doi.org/10.1145/567752.567774>

Joseph Scientist. 2009. The fountain of youth. (Aug. 2009). Patent No. 12345, Filed July 1st., 2008, Issued Aug. 9th., 2009.

David Harel. 1978. LOGICS of Programs: AXIOMATICS and DESCRIPTIVE POWER. MIT Research Lab Technical Report TR-200. Massachusetts Institute of Technology, Cambridge, MA.

Kenneth L. Clarkson. 1985. Algorithms for Closest-Point Problems (Computational Geometry). Ph.D. Dissertation. Stanford University, Palo Alto, CA. UMI Order Number: AAT 8506171.

David A. Anisi. 2003. Optimal Motion Control of a Ground Vehicle. Master's thesis. Royal Institute of Technology (KTH), Stockholm, Sweden.

Harry Thornburg. 2001. Introduction to Bayesian Statistics. (March 2001). Retrieved March 2, 2005 from http://ccrma.stanford.edu/~jos/bayes/bayes.html

ACM. Association for Computing Machinery: Advancing Computing as a Science & Profession. Retrieved from http://www.acm.org/.

Wikipedia. 2017. WikipediA: the Free Encyclopedia. Retrieved from https://www.wikipedia.org/.

Dave Novak. 2003. Solder man. Video. In ACM SIGGRAPH 2003 Video Review on Animation theater Program: Part I - Vol. 145 (July 27-27, 2003). ACM Press, New York, NY, 4. DOI:https://doi.org/99.9999/woot07-S422

Barack Obama. 2008. A more perfect union. Video. (5 March 2008). Retrieved March 21, 2008 from http://video.google.com/videoplay?docid=6528042696351994555

Martha Constantinou. 2016. New physics searches from nucleon matrix elements in lattice QCD. arXiv:1701.00133. Retrieved from https://arxiv.org/abs/1701.00133