Predicting Tetris Performance Using Early Keystrokes

Gianluca Guglielmo
Tilburg University, CSAI Tilburg, Netherlands
G.Guglielmo@tilburguniversity.edu

Michal Klincewicz
Tilburg University, CSAI Tilburg, Netherlands
M.W.Klincewicz@tilburguniversity.edu

Elisabeth Huis in 't Veld
Tilburg University, CSAI Tilburg, Netherlands
E.M.J.HuisintVeld@tilburguniversity.edu

Pieter Spronck
Tilburg University, CSAI Tilburg, Netherlands
P.Spronck@tilburguniversity.edu

ABSTRACT

In this study, we predict the different levels of performance in a Nintendo Entertainment System (NES) Tetris session based on the score and the number of matches played by the players. Using the first 45 seconds of gameplay, a Random Forest Classifier was trained on the five keys used in the game obtaining a ROC_AUC score of 0.80. Further analysis revealed that the number of down keys (forced drop) and the number of left keys (left translation) are the most relevant keys in this task, showing that by merely including the data from these two keys our Random Forest Classifier reached a ROC_AUC score of 0.83. We conclude that the keylogger data during the early phases of a game session can be successfully used to predict performance in longer sessions of Tetris.

CCS CONCEPTS

• **Applied computing**; • **Psychology;**; • **Human-centered computing**; • **Empirical studies in HCI; Laboratory experiments**;

KEYWORDS

Video games, Performance, Expertise, Tetris, Machine Learning, Peripherals

ACM Reference Format:
Gianluca Guglielmo, Michal Klincewicz, Elisabeth Huis in 't Veld, and Pieter Spronck. 2023. Predicting Tetris Performance Using Early Keystrokes. In *Foundations of Digital Games 2023 (FDG 2023), April 12–14, 2023, Lisbon, Portugal*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3582437.3587184

1 INTRODUCTION

The expertise of a player in a certain game can be witnessed by their behaviour throughout the game. Recent studies where specific in-game behaviours were extracted using a python-implemented version of Tetris called Meta-T showed that even at the very start of the game, very proficient Tetris players show differences in behaviour and performance already at Level 0 and Level 1 [1, 2].

The aim of our study was to assess to what extent it is possible to accurately predict performance in the classic version of NES Tetris using just the keystrokes pressed in the first 45 seconds at the start of the game and whether this would be enough to discriminate between different levels of performance for an entire 13 minutes gameplay session. For these purposes, we trained a Random Forest Classifier, a classifier able to perform well on relatively small datasets like ours [3], using the information extracted from the keystrokes as features to discriminate different levels of performance. Further analysis will also provide insights into which keys are the most relevant for this task and the reasons for their importance.

FDG 2023, April 12–14, 2023, Lisbon, Portugal

ACM ISBN 978-1-4503-9855-8/23/04.
https://doi.org/10.1145/3582437.3587184

2 RELATED WORKS

2.1 Tetris, Performance, and Expertise

Tetris has already been used to study expertise and specific behaviors that may generally characterize proficient Tetris players [1, 2, 4]. For example, it was found that participants proficient in Tetris perform more rotations than needed, which may be aimed at unloading their working memory, thereby improving their ability to process incoming information more efficiently [4]. Behaviors that take advantage of cognitive architecture in this way are sometimes called "epistemic actions" since, unlike "pragmatic actions", they are not used to reach a goal as such, but rather to facilitate information gain about the goal-related environment. However, a more recent study examined epistemic actions in Tetris and showed that proficient players optimize and then reduce the number of extra rotations, and non-pragmatic actions [6], shedding some doubt on the role of epistemic actions in Tetris expertise.

Expertise in Tetris, based on in-game performance, has been already studied [1, 2, 4, 5]. For example, in a study by Lindstedt and colleagues, a principal components regression on 35 features extracted from the Tetrazoids (henceforth zoids), was used to predict performance, operationalized as a mean score obtained during a session [1] finding that the most relevant components are "disarray", "4-line planning", and "decide-move-place". Another study assessed which factors experts may use to overcome performance plateaus, such as the minimum lines cleared [2].

Based on this research, we hypothesize that proficiency in Tetris on an entire session may be predicted using early players' behavior in terms of keystrokes.

3 METHODS

3.1 Participants

A total of 80 participants were recruited at Tilburg University and among the authors' contacts. The average age of the participants was 22.27 (SD = 5.92), of which 40 participants were males, and 39 participants were females; 1 participant did not declare their

biological sex. The study was approved by the Tilburg University ethics committee under the REDC 2021.35a decision.

3.2 Data Collection Procedure

The experiment lasted 40 minutes on average and was carried out using an online version of NES Tetris[1]. After having filled in the informed consent, participants were asked to report their biological sex, age, and self-assessed Tetris experience on a Likert scale between 1-5 (not experienced at all-really experienced). Before starting their session, the experimenter explained each of the 5 keys that can be pressed during the game and their effects on the Tetris zoid. The Down key was used to perform forced drop increasing the fall rate of the zoid, the left and right keys were used to move the zoids left and right while the X and Z keys were used to respectively perform clockwise and counterclockwise rotations.

The Tetris session started with a test period of 2 minutes and ask the experimenter questions about the game. Participants were instructed they could use the "Next" information provided by the Tetris environment, to obtain insights into where to place the next zoid. Information about the zoids placed was also provided directly by the game session in the “Statistics” square. Then, the participants started their actual Tetris session.

The experimenter informed the participants they would play for less than half an hour without specifying the exact amount of time the participants would play. During their Tetris session, the participants were asked to restart from level 0 in case they lost and to stop when the experimenters asked them to. The session’s duration adopted for this study was 13 minutes. During the Tetris sessions, the keyloggers of the participants were collected using the Recording User Input (RUI) software[2] [7].

3.3 Data Preprocessing

Performance in the game was operationalized as the mean score of all completed games played in the 13-minutes session. The only exception to this were 2 players who played just one game throughout the 13 minutes, without losing. For those players, we used the score obtained at the 13th minute. Players had a mean score of 4291.93 (SD = 6771) and played an average of 2.76 matches (SD = 1.88). No participant moved beyond level 0 of difficulty during the first 45 seconds. The keylogger data were extracted from the first 45 seconds of the session.

3.4 Levels of Performance

Levels of performance were defined by clustering two variables; the number of matches played [2] and the average score calculated on the matches played [1, 2, 4, 5] which also provides indirect information about the level the player reached during a match retaining at the same time higher inter-matches variability. We would expect proficient players to obtain higher scores than less proficient ones but we would also expect that proficiency in Tetris is associated with fewer matches played in a session [2].

Afterward, a K-means clustering algorithm was used to create groups with different expertise levels as done in a previous study with 33 participants [6]. The average score data were log-transformed as they were found to be skewed (skew = 2.09) based on the limits of + 1.5 and -1.5 [8]. Since K-means is affected by severely skewed data [9], the average score data were log-transformed [2]. After the transformation, the data had an approximately symmetric distribution (skewness = 0.35). On the contrary, the number of matches played was not severely skewed (1.06) [8]; for this reason, these data were not transformed.

Three clusters were detected using the elbow method [2, 6]. The three clusters were renamed Experts (n=20), Intermediates (n=38), and Novices (n=22) [1, 2, 4, 6]. Experts played on average n = 1.1 matches with an average score of 14,013 (SD = 7556), which was n = 2.32 matches (M = 1483.26, SD = 971.1) for Intermediates and n = 5.22 (M = 147.01, SD = 147.07) for Novices. Furthermore, an ANOVA on the self-assessed experience showed a main effect of group ($F(2,77) = 9.32$, $p < .001$), showing that the Expert players seem to perceive themselves as more experienced in the game (M = 2.80, SD = 075) than the players of the other two groups (Intermediates; M = 2.21, SD = 0.87, Novices: M=1.73, SD=0.69), where the 3 groups, after having run a post hoc test, showed a significant difference between Novices and Intermediates ($p < .05$), Intermediates and Experts ($p < .05$), and Experts and Novices ($p < .001$).

Before proceeding with the classification task, we analysed the groups’ differences in the keystrokes’ information (see table 1). These analyses were run using an ANOVA or the non-parametric Kruskal-Wallis test, and the Holm and Dunn test as respective post hoc tests. The aforementioned types of tests were adopted according to the residuals’ distribution, evaluated using the Kolmogorov-Smirnov test, and the homogeneity of variance using the Bartlett test.

Table 1: Mean number of keys pressed per type across the three groups.

	Novices	Intermediates	Experts
Down key	17.23 (SD: 23.37)	38.45 (SD: 28.95)	89.35 (SD: 42.46)
Left key	18.27 (SD: 7.50)	23.03 (SD: 10.85)	32.35 (SD: 12.68)
Right key	16.64 (SD: 6.09)	17.73 (SD: 7.86)	21.95 (SD: 7.28)
X	7.75 (SD: 6.62)	9.92 (SD: 8.35)	9.50 (SD: 10.32)
Z	8.37 (SD: 7.23)	7.24 (SD: 7.37)	9.30 (SD:10.30)

4 RESULTS

4.1 Keystrokes

Statistical analyses were run on the keystrokes pressed by the players to detect differences between groups (see Table 1).

The results revealed that the number of times the down key (forced drop) was pressed differed between groups ($H(2) = 33.94$, $p < .001$). Experts used the down key more often than Intermediates ($p < .001$) who in turn used it more than Novices ($p < 0.01$). A similar

[1]Available at the following website: https://arcadespot.com/game/classic-tetris/
[2]All the information about RUI and the software itself are available at the following link: https://acs.ist.psu.edu/projects/RUI/

Table 2: The classification results for RF-5 and RF-2 compared against the DC used for baseline.

	Accuracy	ROC_AUC	Precision	Recall	F1 score
DC	0.47 (SD: 0.03)	0.50 (SD: 0.00)	0.23 (SD: 0.03)	0.47 (SD: 0.03)	0.31 (SD: 0.03)
RF-5	0.71 (SD: 0.10)	0.80 (SD: 0.05)	0.71 (SD: 0.08)	0.66 (SD: 0.08)	0.67 (SD 0.07)
RF-2	0.69 (SD: 0.04)	0.83 (SD: 0.03)	0.70 (SD: 0.05)	0.66 (SD: 0.08)	0.66 (SD: 0.08)

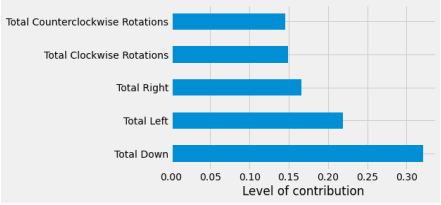

Figure 1: The features (keys) used to train the Random Forest Classifier and their level of contribution.

main effect was found on the number of left key presses ($F(2,77) = 8.81$, $p < .001$). Again, Experts pressed the left key more than Intermediates ($p = .011$) and Novices ($p < .001$), but no difference was found between Intermediates and Novices ($p = .08$). No other statistically significant result was found when analyzing the other keys used to play the game (right key, X, and Z).

4.2 Classification task

In order to estimate the Random Forest Classifier performance, we trained a Dummy Classifier (DC), implemented with the majority vote, to evaluate the performance against the baseline. the Dummy classifier is a classifier that ignores the input features and is used to establish a baseline against which more complex models are compared[3]. The metrics used to evaluate the performance are balanced accuracy, weighted ROC_AUC score (one versus rest methodology which should be insensitive to class imbalance [10]), weighted recall, weighted precision, and weighted F1 score. The weighted metrics were used given the imbalanced classes in our dataset. The machine learning task was implemented with Python (Scikit-learn library) balancing the weights of the classes and running a randomized grid search for hyperparameter tuning before evaluating the model using nested cross-validation to optimize the ROC_AUC score[4] and evaluating the model performance [11]. Before evaluating the performance of the Random Forest classifier, we used the Extra Tree, a method that allows to estimate the importance of each feature [12], to obtain insights about which keys may be more relevant to discriminate between the three groups; the results of Extra Trees are shown in Figure 1.

The results in Figure 1 seem to point in the same direction as our statistical analysis presented in the previous section of this work. The down key (forced drop) and the left key (left translation) seem to be the most predictive keys when discriminating between different levels of performance. To evaluate if these two keys alone provide satisfactory results, we compared the DC against the performance obtained by the Random Forest Classifier trained both with the 5 keys (RF-5) and with the 2 keys most contributing keys (RF-2) (see Table 2).

The results obtained show that both RF-5 and RF-2 outperform the baseline classifier.

5 DISCUSSION

Our aim was to find out whether the first 45 seconds of keystrokes in a Tetris session can be used to discriminate between players based on their performance. Our results show that they can when keystrokes are used as features in a Random Forest Classifier. Furthermore, we found out that the down and left keys are the most important to this task. The press-frequency of these two keys in the first 45 seconds of gameplay is sufficient to discriminate between different levels of player performance.

However, it is not entirely clear what explains our pattern of results and particularly the importance of the down and left keys. We speculate that more skilled players press the down key more to increase the speed of the game, thereby gaining points faster. Pressing the down key makes it more likely to make mistakes in controlling the zoids, so it is something that inexperienced players or those with less confidence would not do. The importance of the left key, on the other hand, is likely connected to a spatial asymmetry of the Tetris board relative to the size of a zoid. Most zoids that first appear are located 4 moves from the left side of the board and 3 moves from the right; the only exceptions are the line zoid, which is 3 moves away from each side, and the square zoid which is 4 moves away from each side. This means moving zoids to the left more often reflects either a tacit or explicit understanding of the Tetris board, which provides more space and hence more combinations for placement on the left. If the left and right keys' contributions were symmetrical, we would expect similar results in statistical analyses and in the Extra Trees' results. On the contrary, our results show that only the left key and down key are relevant to discriminate between the 3 groups. However, this effect may also be due to the specific position of the fingers on the keyboards' arrows. Future studies should investigate if more experienced players are aware of this asymmetry and whether they exploit it intentionally when placing zoids.

Our results also show that rotations performed by players having different levels of performance are not statistically significant, at

[3]More information is available at the following scikit-learn link: https://scikit-learn.org/stable/modules/generated/sklearn.dummy.DummyClassifier.

[4]The procedure followed to perform the nested cross-validation can be found at the following link: https://inria.github.io/scikit-learn-mooc/python_scripts/cross_validation_nested.html

least when evaluating the first 45 seconds of the Tetris session. This seems to indirectly support previous results, which showed that better-performing players optimize the number of rotations without using extra rotations to gain insights about the zoids positioning [5]. This means that our results go in the opposite direction of what was suggested by Maglio and colleagues [4] about the use of epistemic actions. However, epistemic actions may still occur in the later stages of the game.

Our study also makes a methodological contribution by providing evidence that information extracted from peripherals, such as keystrokes on a keyboard, can be used to discriminate between players with different levels of performance at the early stages of a game session. Allegedly, these features may be used to track performance and expertise in noisy and dynamic environments such as tournaments.

Finally, there are important limitations of our study. First, we had a relatively small sample compared to previous studies using Tetris [1, 2]. Second, our results are specific to Tetris which has a limited number of mechanics and in which skilful behaviour can manifest early in a game session [1, 2]. However, similar methods may be applied to more complex games, for example tracking early decisions made by the player during the game and their effect on the entire game session or for example to determine which commands are the most relevant ones when the player has more than 5 keys to use as it occurs in Tetris. As a consequence, future studies may investigate if the methods suggested in this work are applicable to other video games with different mechanics and performance profiles.

6 CONCLUSIONS

The results reported in this study show that keystroke data extracted from the first 45 seconds of a Tetris session can be used to discriminate between participants having different levels of performance in a 13-minutes Tetris session. Future studies may provide new evidence to our results by collecting a bigger sample and evaluating if the method here conveyed may be used to detect performance and expertise in the early stages of other video games.

ACKNOWLEDGMENTS

The research reported in this study is funded by the MasterMinds and Data2Game projects, part of the RegionDeal Midland WestBrabant, and is co-funded by the Ministry of Economic Affairs, Region Hart van Brabant, REWIN, Region West-Brabant, Midpoint Brabant, Municipality of Breda, Netherlands Research Organisation (NWO), and Municipality of Tilburg awarded to MML

REFERENCES

[1] John K. Lindstedt and Wayne D. Gray. 2019. Distinguishing Experts from Novices by the Mind's Hand and Mind's Eye. Cognitive Psychology 109 (2022): 1-25
[2] Wayne D. Gray and Banerjee Sounak. 2021. Constructing expertise: Surmounting Performance Plateaus by Task, by Tools, and by Techniques. Topics in Cognitive Science 12.4 (2021): 610-665.
[3] Qi, Yanjun. "Random forest for bioinformatics. 2012." *Ensemble machine learning: Methods and applications*. Boston, MA: Springer US, 2012. 307-323.
[4] Paul P. Maglio & David Kirsh. 1996. Epistemic Action Increases with Skill. Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society (1996).
[5] Mark Destefano, John K. Lindstedt, and Wayne D. Gray. 2011. Use Complementary Actions Decreases with Expertise .in Proceedings of the Annual Meeting of the Cognitive Science Society. Vol. 33. No.33. (2011)
[6] Paul P. Maglio, 1995. The Computational Basis of Interactive Skill. University of California, San Diego. (1995).
[7] Urmila Kukreja, William E. Stevenson, and Frank E. Ritter. 2006. RUI: Recording User Input from Interfaces under Windows and Mac OS X. Behavior Research Methods 38.4. (2006)L 656-659.
[8] Barbara G. Tabachnick and Linda S. Fidell. 2013. Using Multivariate Statistics (sixth edition). Pearson College. New Jersey, Usa
[9] Ch. N. Santhosh Kumar, K. Nageswara Rao, A. Govardhan, N. Sandhya. 2015. Subset K-means Approach for Handling Imbalanced-Distributed Data. in Emerging ICT for Bridging the Future-Proceedings of the 49th Annual Convention of the Computer Society of India CSI Volume 2. Springer, Charm, (2015) .
[10] David J. Hand and Robert J. Till. 2001. A Simple Generalization for the Area Under the ROC Curve for Multiple Class Classification Problems. *Machine Learning* **45**, 171–186 (2001). https://doi.org/10.1023/A:1010920819831
[11] Ioannis Tsamardinos, Amin Rakhshani, and Vincenzo Lagani. 2015. Performance-estimation Properties of Cross-validation-based Protocols with simultaneous Hyper-parameter Optimization. International Journal on Artificial Intelligence Tools 24.05. (2015): 1540023
[12] Ahmad, Muhammad Waseem, Jonathan Reynolds, and Yacine Rezgui. 2018. "Predictive modelling for solar thermal energy systems: A comparison of support vector regression, random forest, extra trees and regression trees." *Journal of cleaner production* 203 (2018): 810-821.