COGNITIVE SCIENCE
A Multidisciplinary Journal

Cognitive Science 48 (2024) e13454

ISSN: 1551-6709 online
DOI: 10.1111/cogs.13454

Tracking the Cognitive Band in an Open-Ended Task

John R. Anderson, Shawn Betts, Daniel Bothell, Cvetomir M. Dimov, Jon M. Fincham

Department of Psychology, Carnegie Mellon University

Received 11 December 2023; received in revised form 20 April 2024; accepted 30 April 2024

Abstract

Open-ended tasks can be decomposed into the three levels of Newell's Cognitive Band: the Unit-Task level, the Operation level, and the Deliberate-Act level. We analyzed the video game Co-op Space Fortress at these levels, reporting both the match of a cognitive model to subject behavior and the use of electroencephalogram (EEG) to track subject cognition. The Unit Task level in this game involves coordinating with a partner to kill a fortress. At this highest level of the Cognitive Band, there is a good match between subject behavior and the model. The EEG signals were also strong enough to track when Unit Tasks succeeded or failed. The intermediate Operation level in this task involves legs of flight to achieve a kill. The EEG signals associated with these operations are much weaker than the signals associated with the Unit Tasks. Still, it was possible to reconstruct subject play with much better than chance success. There were significant differences in the leg behavior of subjects and models. Model behavior did not provide a good basis for interpreting a subject's behavior at this level. At the lowest Deliberate-Act level, we observed overlapping key actions, which the model did not display. Such overlapping key actions also frustrated efforts to identify EEG signals of motor actions. We conclude that the Unit-task level is the appropriate level both for understanding open-ended tasks and for using EEG to track the performance of open-ended tasks.

Keywords: Cognitive Modeling; Open-ended Tasks; EEG; Model Tracing; Video Games

Correspondence should be sent to John R. Anderson, Department of Psychology, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA, 15213 USA. E-mail: ja@cmu.edu

1. Introduction

One goal of this research is to use EEG to track what is happening as someone performs a complex task. There are two interconnected motivations behind this goal. The theoretical motivation is to test model predictions about what events happen and when they happen. For instance, a model may make predictions about the steps that will happen in solving a complex mathematical problem. The applied motivation is to recognize when a critical event has happened and respond appropriately. For instance, in the mathematical problem-solving context, one might detect when the student has reached a dead end and suggest a new approach.

The advantage of EEG over other imaging modalities is its temporal resolution and its relative practicality. It has been used in many controlled laboratory settings to track processes that happen in trials of no more than a few seconds. While such studies have some advantages, we are interested in tracking cognition in longer, more opened-ended tasks where events emerge as an interaction between an agent and the environment. This is what happens in more naturalistic and ecologically valid contexts such as driving, decision-making in complex social interactions, or problem-solving in uncontrolled settings. In such situations, there is not a single sequence of correct actions or even a few possible sequences. Rather, there are exponentially exploding paths of possible action choices and responses from the environment. These are circumstances that allow us to study how the detailed processes revealed in the short-trial-based studies are combined to produce coherent behavior. Understanding how to use EEG in such tasks also has applications to the development of brain–computer interfaces (for reviews, see Abiri, Borhani, Sellers, Jiang, & Zhao, 2019; Lotte et al., 2018).

Given the time scale over which such tasks play out, one needs to consider the temporal detail to which one aspires in tracking cognition. It is useful to place such aspirations into Newell's (1990) scales of cognition (see Table 1). The open-ended tasks reside in his Rational Band, which ranges from minutes to hours. Below, the Rational Band is the Cognitive Band, which he placed as spanning from Deliberate Acts which happen at the rate of around 100

Table 1
Newell's time scales of human action

Scale (s)	Time units	System	World (theory)
10^7	months		
10^6	weeks		Social Band
10^5	days		
10^4	hours	Task	
10^3	10 min	Task	Rational Band
10^2	minutes	Task	
10^1	10 s	Unit Task	
10^0	1 s	Operations	Cognitive Band
10^{-1}	100 ms	Deliberate Act	
10^{-2}	10 ms	Neural circuit	
10^{-3}	1 ms	Neuron	Biological Band
10^{-4}	100 μs	Organelle	

ms to Unit Tasks which happen at the rate of about 10 s. It is our belief (Anderson, 2002) that applications can be achieved by decomposing what is happening at the Rational Band into activities happening in the Cognitive Band. However, it is uncertain what level within the Cognitive Band is the most appropriate level.

The Cognitive Band is the focus of many cognitive architectures (Kotseruba & Tsotsos, 2020 for a review), which break Rational Band activities into the performance of tasks at the Cognitive Band. We will be working with the ACT-R architecture (Anderson, 2007, Anderson et al., 2004) to find an interpretation of the subject's behavior on a specific task in terms of execution of activities in the Cognitive Band. This paper will report considerable success in doing so at the Unit-Task level of the Cognitive Band, limited success at the Operation level, and a roadblock at the Deliberate-Act level. While, in a certain sense, the results are specific to ACT-R (as well as to details of our approach), they are representative of the issues in tracking cognition in open-ended tasks.

The open-ended task in this paper is a video game. Video games are becoming increasingly popular vehicles for the study of cognition (e.g., Altarelli, Green, & Bavelier, 2020; Boot, 2015, Gray, 2017). Many video games are excellent examples of open-ended tasks, in that behavior emerges as an interaction between the game software and the players. Such games combine the learning of complex strategies and perceptual-motor skills. Dealing with the demands of such games has led to advances in the ACT-R architecture (Anderson, Betts, Bothell, Hope, & Lebiere, 2019, Anderson, Betts, Bothell, & Lebiere, 2021; Gianferrara, Betts, Bothell, & Anderson, 2021). The video games we have studied have involved a high rate of action from the players. The high rate of action also provides a solid ground truth for judging the accuracy of tracking from EEG. While our ultimate goal is to extend to tasks where there are not such visible signatures of mental progress, video games serve as a good test bed.

1.1. *Using EEG to track cognition*

While the more typical application of EEG for tracking cognition in open-ended tasks has been to identify relatively enduring states like fatigue in driving (e.g., Jap, Lal, Fischer, & Bekiaris, 2009) or workload while managing a system (Baldwin & Penaranda, 2012), our concern is the identification of specific cognitive events localized in time. We have had success in localizing events like memory retrieval in short trials of a few seconds (Anderson, Zhang, Borst, & Walsh, 2016, Borst & Anderson, 2021). In these well-defined tasks, a combination of multi-variate pattern analysis (MVPA) and hidden semi-Markov models (HSMM) can localize Deliberate Acts. This approach treats the Deliberate Acts as forming a semi-Markov process where the probability of the next act and the time to that act depend only on the previous act. The analysis discovers *bumps* in the EEG that are event-related potential (ERP)-like responses that mark individual cognitive steps. The MVPA provides the probability that the EEG signal reflects the bump associated with a particular act at each point in the trial. The HSMM combines these probabilities with estimates of the temporal distribution of the acts to localize the acts in the trial.

We were not able to extend these methods directly to open-ended tasks which do not have just one or a few solution paths. As these tasks get longer, uncertainty increases about the temporal location of the acts which overwhelms the signal-to-noise ratio (Fincham, Lee, & Anderson, 2020). Anderson, Betts, Fincham, Hope, and Walsh (2020) reported what they called the *sketch-and-stitch* method for localizing acts in such a task. For the goals of this paper, we will only be considering the sketch portion of this method. Anderson et al. (2020) focus on *critical events* that happen relatively infrequently (at the Unit-Task level time scale) and whose distribution can often be given a semi-Markov treatment. Also, these critical events often produce strong EEG signals. An HSMM-MVPA can localize the critical events with relatively high levels of precision. There are two major differences between this approach and the earlier bump approach. First, this approach is not discovery-based, but requires training on the signals that are associated with these critical events. This was possible because there is information in the game record as to when these critical events happened. Second, the EEG signals of critical events extend over 500 ms rather than the 50 ms associated with bumps.

The video game in this paper is a version of the Space Fortress game (e.g., Mane & Dunchin, 1989) which has been used for decades to test training regimes: subjects fly a ship around a central fortress trying to destroy it. In the last decade, detailed modeling of Space Fortress has become the target of cognitive modeling because it reflects a complex mixture of perceptual, motor, and cognitive skills, how they are integrated, and improve with practice (e.g., Anderson et al., 2019; Rahman & Gray, 2020). Anderson et al. (2020) demonstrated the sketch-and-stitch method with a rather simple version of the game which is called Autoturn. While the choice of simplicity was strategic for the initial exploration, its simplicity had limitations for the purposes of a strong evaluation. There were significant constraints on key patterns and flight path, leaving the game at the low end of open-endedness.

1.2. Co-op Space Fortress

The video game in this paper is a 2-player version of Space Fortress, called Co-op Space Fortress,[1] which has a recent ACT-R model (Dimov et al., 2023). The data in this paper provide a further test of the Dimov model. In Co-op Space Fortress, depicted in Fig. 1, two players control two identical, but differently colored spaceships and their goal is to destroy a fortress located in the middle of a gray hexagon. The distinctive feature of Co-op Space Fortress is that the two players need to coordinate their actions to destroy the fortress. The fortress is surrounded by a shield, which is impenetrable to a ship's missiles, but the back portion of the shield disappears when the fortress fires shells at a ship. One player, the *bait*, tempts the fortress to lower its shield in order to shoot at the bait. This exposes the back of the fortress and the other player, the *shooter*, can move behind the fortress and shoot it with a missile.

As in Space Fortress, these spaceships fly in a frictionless environment and are controlled with four keys: W to thrust, D and A for clockwise and counterclockwise turns, respectively, and the space bar to launch a missile. The Supplementary Material describes the detailed effects of these key presses as well as other details of the game. Navigating the ship without friction is counterintuitive. For example, the ship never stops unless the player turns 180

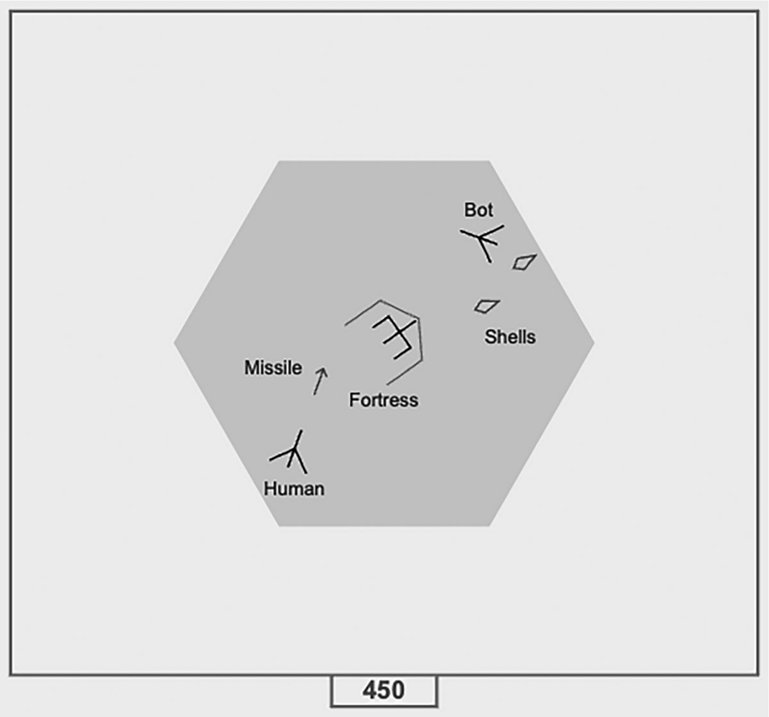

Fig. 1. The Co-op Space Fortress screen, showing the hexagonal battle area with the Fortress in the middle. The Fortress has turned and has shot shells at the bait. The shooter is behind the Fortress and has shot a shell at the exposed rear of the fortress.

degrees relative to the flight direction and thrusts for just the right amount of time. Consequently, as in previous Space-Fortress-based games, learning to navigate has been a major challenge. This is even a greater challenge in this game because the needed flight paths are much more open, reflecting a dynamic game situation that is partly determined by the other player.

At the beginning of each game, the two players start outside (i.e., to the left and right of) the gray hexagon. They both need to enter the hexagon, because it defines the effective range of fortress shells and ship missiles. The fortress will not take aim at the bait while it is out of the hexagon and the shooter cannot destroy the fortress when it is outside of the hexagon. If the fortress is destroyed, 100 points are gained and both players must leave the hexagon before the fortress respawns and they can attack again. Players lose 100 points if a ship is shot by the fortress or if the ship crashes into the fortress or the outer boundary of the game. After a ship death, the ship will respawn 1 s later in the starting position. In addition to points gained by kills and lost by deaths, 10 points are lost for every shot that does not destroy the fortress.

Unlike in the original Co-op game, one player is always the shooter and the other always the bait. In the current study, the bait is an artificial agent developed in Dimov et al. (2023), *the* *bot*, to minimize variation in success due to the skill of the other player. The motor timing of the bot was based on the ACT-R model including randomness in exact motor execution. The Appendix contains details of bot variability. It basically behaves as a highly trained model and was better than most subjects or ACT-R models. The diversity of behavior in Co-op Space fortress is quite large. The randomness in both bait and shooter magnify each other quickly putting the players in novel situations. No two games are identical or even just slight variants of one another. While there are tasks even more open-ended than this game, it is fairly far along the open-ended continuum while still possessing an analyzable structure.

The ACT-R model for Co-op Space Fortress leveraged many of the features of past models to play other Space Fortress games (see Dimov et al. for a detailed description of the model). As other ACT-R models of skill acquisition, it starts with a declarative representation of how it should act. As role choice is fixed in this version of the game, this declarative representation is simpler than in the original Dimov model, because it skips information about role determination. The model transitions with practice from a beginning stage of retrieving this knowledge and deciding how to use it in the game to a final stage of *direct actions* which simply recognize what to do in a game state. Also, with practice, the model learns control parameters that determine exactly how it performs actions such as how fast to fly or what an adequate shooting aim is. A major extension of this model over previous models was learning to predict where the ship would be in the future and where the other ship would be. This information is used to plan and adjust flight trajectory.

EEG data were collected after subjects (and models) had achieved a fairly stable and successful level of play. Subjects played a series of 30 3-min games. The first 10 games were played without EEG and subjects had to average at least 200 points per game in games 8–10 to go onto the final 20 games which were played with EEG. Most of the learning for these successful subjects was complete by the 10th game.

2. Methods

2.1. Subjects

A total of 27 subjects were recruited from the CMU population of students and staff between the ages of 18 and 40. Five subjects were excluded because they failed to meet the performance criterion on the first 10 games, leaving 22 subjects (13 male, 9 female, mean age 23.9 years). All were right-handed. None reported a history of neurological impairment. Subjects were paid $60 for participation in the experiment that lasted about 2 h. In addition, they earned a bonus for performance (4 cents per 100 points, a total bonus range $3.41–$21.29 with an average of $13.43). All subjects signed informed consent and the experimental procedure and data handling was approved by the ethics committee of Carnegie Mellon University.

2.2. Game play

After subjects studied game instructions, they played 10 3-min games, choosing to move on to the next game at their own pace. If they passed the criterion of an average of at least 200 points in the last three games, they went into the EEG lab and were fitted with the headset. Then, they played 20 more games again at their own pace. Every 60th of a second, the game records the state of the game (e.g., whether the ships and fortress are alive, where the ships are and their direction and speed of movement, whether shells or missiles are on the screen, whether a key is pressed, and whether the fortress has opened its shield). This serves as the ground truth both for training the EEG decoder and for testing its predictions.

2.3. *EEG analysis and classifier*

The EEG was recorded from 64 Ag-AgCl sintered electrodes (10-20 system) using a Biosemi Active II System (Biosemi, Amsterdam, Netherlands). The EEG was re-referenced online to the combined common mode sense and driven right leg circuit. Electrodes were also placed on the right and left mastoids. Scalp recordings were algebraically re-referenced offline to the average of the right and left mastoids. The EEG and electrooculogram (EOG) signals were digitized at 512 Hz and were subsequently filtered with a bandpass filter of .1–40.0 Hz. The vertical EOG was recorded as the potential between electrodes placed above and below the left eye, and the horizontal EOG was recorded as the potential between electrodes placed at the external canthi. The EEG recording was decomposed into independent components using the EEGLAB FastICA algorithm (Delorme & Makeig, 2004). Components associated with eye blinks were automatically identified and manually confirmed. All but the marked Independent Component Analysis (ICA)s were projected back to the EEG signal.

The EEG signal was recorded continuously for the entire experimental session and broken into 3-min games. To match the rate of game recording, the data were down-sampled to 60 Hz with default EEGLab anti-aliasing filtering applied (Finite Impulse Response (FIR) low-pass filter, 45 Hz cutoff frequency [−6 dB] and 10 Hz transition bandwidth). We excluded any 60 Hz sample that had an electrode with a value below −500 microvolts or above 500 microvolts. This resulted in loss of data for 60 of the 440 games (22 subjects × 20 games). For these cases, the number of samples lost from the 10,800 ticks (3 min × 60 s × 60) varied from 1 to 1591 with a mean of 244. A 1-s window around each game tick (30 game ticks before, the game tick, and 30 game ticks after) was used to classify whether a game tick contained a critical event. This means that each game tick had associated with it a vector of $61*64 = 3904$ electrode readings, representing regional effects, frequency effects (below 30 Hz), and their interactions. Because the vector associated with a game tick requires a complete signal for 1 s, game ticks at the beginning and end of a game do not have corresponding vectors, nor do game ticks in or near periods of deleted signal. The available vectors for each game were z-scored to standardize them across games. To reduce dimensionality and filter out noise, the vectors from all games were subject to a principal component analysis (PCA) and the 1000 top dimensions were kept. Depending on how many points were excluded, the result was 8572−10,740 (mean 10,689) 1000-element vectors associated with game ticks. These are what were used for all classification analyses. The classifications were performed on a per subject basis, which resulted in better classification results than combining data over subjects in Anderson et al. (2020).

Two linear classifiers were trained, one for identification of successes and failures of kill attempts at the Unit-Task level of and one for identification of beginnings and ends of leg of flight at the Operation level. These classifiers learned the conditional probabilities of the EEG patterns associated with different categories of tick events. Assuming normal distributions of the 1000 PCA dimensions for any category, the classifier estimated the parameters of the distribution and so could map any vector onto conditional probabilities for the various categories. For any game that the subject played, the parameters were estimated from the other 19 games of that subject and used to classify the vectors of the target game. The Unit-Task classifier learned four categories: the patterns associated with kills, deaths, and missed shots, plus the pattern of other ticks called *Null* events. The Operations classifier learned three categories: ticks where the subject began a new flight path, ticks were the flight path ended, and Null events. Most of the game ticks are Null events (an average of 99.8% of the ticks for the Unit-Task classifier and 98.6% of the ticks for the Operation classifier). Anderson et al. (2020) only used a subset of the Null events for training. However, using all Null events leads to better estimates of the covariance matrix used by the linear classifier.

2.4. Model data

We ran the model[2] using the same combination of 25 individual-difference parameters as in Dimov et al. Even within a particular individual difference setting, there was substantial variability in the games the model played, reflecting randomness both in the model and the bot. Just as subjects, models played 30 games improving with practice. We repeated these 30-game sessions 300 times at each of the 25 parameter settings, for a total of $300 \times 25 = 7500$ model runs. However, 206 of these models failed to meet the criterion of an average of 200 points over games 8–10. Just as poor-performing subjects were excluded, these model runs were removed, and model behavior was calculated from the remaining 7294 models.

3. Results

Fig. 2a shows the growth in points by subjects and models who met the selection criterion. Points reflect kills (+100 points), deaths (−100 points), and misses (−10 points). The EEG data come from games 11 through 30. Most of the learning is complete by game 11, although there is a weak but significant increase in performance after the 10th game (6 points per game, $t(21) = 2.97, p < .01$). Figs. 2b–e display the data for subcomponents of the score. The majority of deaths (66.5%) are deaths of the human shooter. The figures also show the data from the model, which generally match up, although the model reaches a lower death rate in Fig. 2c.

The remainder of this Results section will consist of a more detailed discussion of this task at the three levels in Newell's Cognitive Band: first, the Unit-Task level, then the Operation level, and then more briefly the Cognitive-Act level. We will be concerned with games 11–30 where EEG was collected.

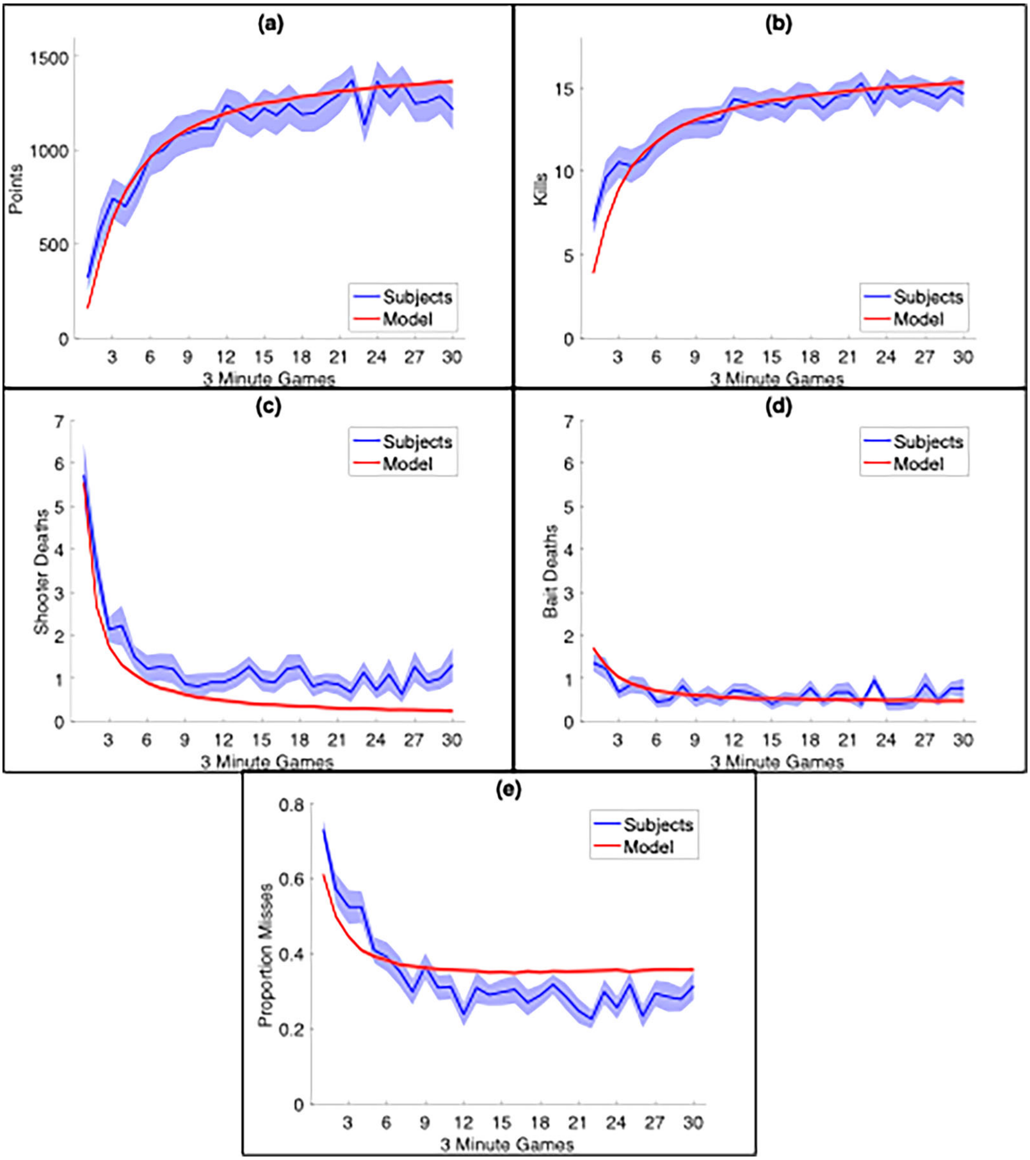

Fig. 2. Mean values (lines) and standard errors (area around lines) per game for subjects and models as a function of game (a) points; (b) number of fortress destructions; (c) number of shooter (human) deaths; (d) number of bait (bot) deaths; (e) proportion of shots that are misses.

3.1. Unit-tasks marked by critical events

Newell took his concept of a unit-task from the earlier work of Card, Moran, and Newell (1983), where they noted that human−computer interaction tasks tended to decompose into subtasks that users define for themselves and took on the order of 10 s to achieve. A classic example would be performing an edit in a text-editor. Achieving this edit would involve achieving a set of subgoals like identifying the edit to be performed, indicating the location of edit in the editor interface, and indicating the content of the edit.

In this game, there is one task that repeats over and over again during the 3-min game. This is destroying the fortress. This decomposes into subgoals of flying along different paths, aiming, and shooting. Just as a text edit may not always be successful and require a repair which is often like the original task, in this game, the ship might be killed, or the shot might miss. Then, the subject tries to complete the task again, but now starting in different circumstances. Thus, the unit tasks in this game end in kills, deaths, or misses. Continuing the terminology of Anderson et al. (2020), we will refer to these three outcomes as *critical events*. These events are rare—on average, only 19.2 of the 10,800 ticks in a game are associated with critical events. These critical events break the game up into units close to that approximate 10 s Newell associated with Unit tasks. As we will see, there are strong EEG signals associated with these terminating events and this allows us to parse the game into its unit tasks.

3.1.1. EEG analysis

Fig. 3 shows 1 s of activity centered on critical events for three central electrodes plus the scalp profiles 19 ticks (.32 s) after the events. Fig. 3d shows the average pattern for the remaining ticks which is essentially 0, reflecting the normalizing of the EEG signal. Fig. 3 shows activity averaged over subjects but note that the classifier estimates patterns subject by subject and individual subjects may show different patterns and the classification will exploit the subject-specific patterns. Nonetheless, the relatively narrow standard errors displayed in Fig. 3 indicate somewhat consistent patterns.

The patterns associated with the critical events in Fig. 3 seem to be variants of the P300 event-related potential (Verleger, 2020). They vary in the timing and magnitude of the positive deflection, with the largest response to deaths and the weakest to misses. While the peaks are the most striking feature, the classifier is based on the activity of all 64 electrodes over all 61 ticks. Essentially, the four patterns in Fig. 3 can be conceived of as occupying points in a 64 × 61 = 3904-dimensional space (or a 1000-dimensional space after the PCA compression). These four points in high-dimensional space can be projected down into a three-dimensional subspace, and as it turns out, that three-dimensional subspace can be projected down into two dimensions with little loss. Fig. 4a illustrates the location of the points in a 2-D space with an attempt at an informative choice of axes and scale. The 0-0 point reflects the zero activity that the Null pattern approximates. The horizontal dimension reflects the magnitude of game points, and the vertical axis represents loss. In terms of EEG activity, as one moves up, one adds proportionally more of the pattern in Fig. 4b, and as one moves to the right, one adds proportionally more of the pattern in Fig. 4c. A death which is at the 1-1 point can be reconstructed as the sum of these two patterns. The two patterns 4b and 4c reflect two peaks of activity at different points in time—the vertical dimension peaking 26 ticks after the event (.43 s) and the horizontal dimension peaking 14 ticks after the event (.23 s). The activity across the 64 electrodes at the two peaks are strongly correlated and no electrode shows a significant difference between the two peaks (maximum t for a difference is $t(21) = 1.53$).

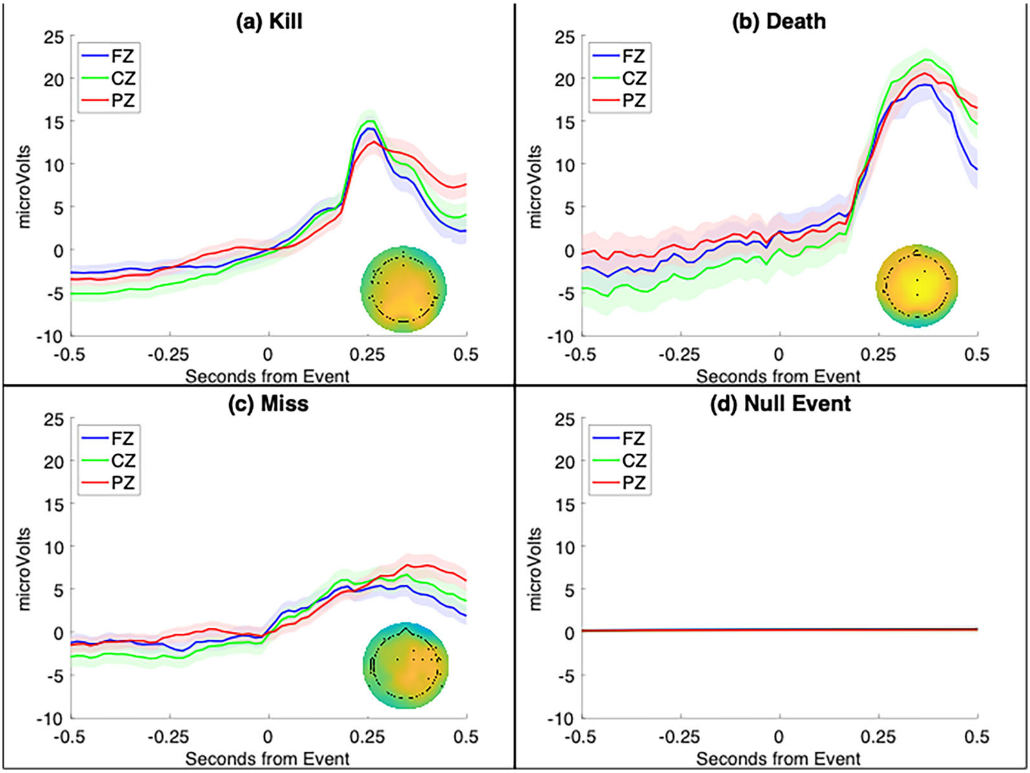

Fig. 3. EEG activity (mean values and standard errors) of three central electrodes from a half second before a critical event to a half second afterward. The scalp profiles illustrate activity at .32 s after the event. (a) Kill: scalp profile is scaled to range between −15 and +15 microvolts; (b) Death: scalp profile is scaled to range between −20 and +20 microvolts; (c) Miss: scalp profile is scaled to range between −10 and +10 microvolts; (d) Null Event.

While Figs. 3 and 4 display average patterns, the challenge for the classifier is to identify individual game ticks using signals which may depart substantially from the subject's average. Table 2 summarizes the success of the classifier using patterns from other games of that subject to classify a particular game. Tables 2a and b give the counts with which various types of ticks were assigned various labels. They use different thresholds for assigning a label to a tick. Table 2a assigns it to the category with the highest conditional probability, while Table 2b assigns it to the category that is most likely given the different frequencies of the categories. Our d-prime measure of discriminability (Wickens, 2002) is 2.65 for Table 2a and 2.58 for Table 2b.[3] Table 2c gives the pair-wise d-primes reflecting the discriminability of the pairs of categories and Table 2d gives the pair-wise area under the curve (AUC), which does not depend on the threshold. The discriminability is quite high and better than in Anderson et al. (2020), which reflects, at least in part, the refinements in the procedure (training using only data from other games of the to-be-classified subject, using all Null events in estimating covariance matrix). Despite their striking appearance in Fig. 3, deaths

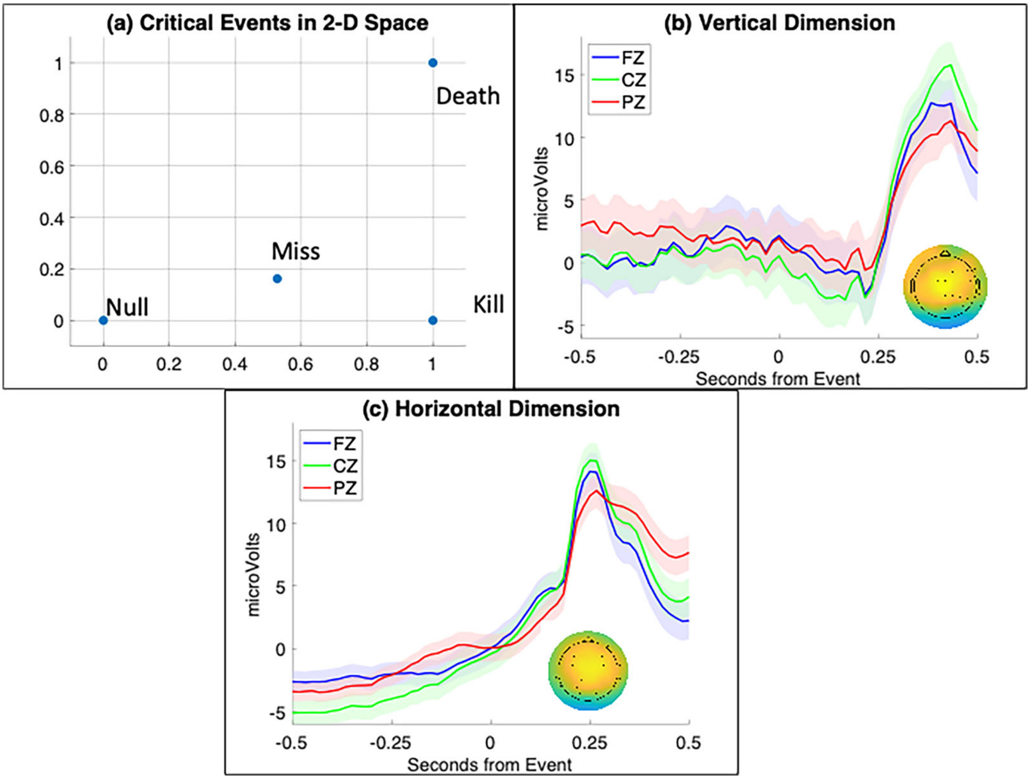

Fig. 4. (a) A representation of the four patterns in Fig. 3 in a two-dimensional space. (b) A representation of the pattern associated with the vertical dimension—the scalp profile is at .43 s and is scaled to range between −15 and +15 microvolts. (c) A representation of the pattern associated with the horizontal dimension—the scalp profile is at .23 s and is scaled to range between −15 and +15 microvolts.

are poorly identified because of their rarity. Classification is done on a per subject basis. The number of deaths per subject varies from 3 to 54 with a mean of 19.5. There is a strong correlation ($r = .809$) between the number of deaths for a subject and the proportion correctly classified.

Looking at average patterns in Fig. 3, it is not surprising that these critical events can be discriminated with high accuracy from other Null game ticks. What may seem surprising is that they are quite discriminable from one another. Part (e) of Table 2 shows the three-way classification by likelihood which has a d-prime of 2.50. Discrimination between the three critical events depends on capitalizing on their distinctive features as illustrated in Fig. 4.

While good, the classification results by themselves do not provide a useful basis for identifying critical events. Many Null events are erroneously labeled as critical events in Table 2a. While Table 2b has somewhat fewer such false positives, it is at the cost of missing many critical events. The HSMM below will not use any particular assignment of ticks to categories, but rather will use the conditional probabilities of the tick pattern for each category.

Table 2
Classification of game ticks according to critical events

(a) Classification by likelihood		Labeled by classifier			
		Null	Kill	Death	Miss
True class	Null	4,584,478	34,938	2188	72,632
	Kill	547	5677	34	41
	Death	112	207	98	4
	Miss	1150	62	0	446

(b) Classification by posterior		Labeled by classifier			
		Null	Kill	Death	Miss
True class	Null	4,726,498	13,170	769	3117
	Kill	1492	4829	14	5
	Death	244	124	61	1
	Miss	1610	10	0	56

(c) Pairwise dprimes		Class 2		
		Kill	Death	Miss
Class 1	Null	3.75	2.91	1.55
	Kill		1.85	3.07
	Death			2.72

(d) Pairwise AUC		Class 2		
		Kill	Death	Miss
Class 1	Null	0.994	0.881	0.831
	Kill		0.854	0.984
	Death			0.895

(e) Three-way classification		Labeled by classifier		
		Kill	Death	Miss
True class	Kill	6134	38	127
	Death	277	110	34
	Miss	252	4	1402

3.1.2. Using an HSMM to localize critical events

The function of the HSMM is to combine the probabilities from the classifier with information about the probability of these events happening at different time points. Fig. 5 shows fitted distributions[4] between events and the probabilities of one event following another. Fig. 5a includes both starts from the beginning of the game and from when the ship reappears outside the hexagon after a death. The ships take a fair amount of time (Fig. 5a) to fly in from the start position and be in position to destroy the fortress. It takes even longer to achieve another kill after a kill (Fig. 5b) because the two ships must first fly out of the battle zone so that the fortress respawns and then they can fly back in. In other cases, one event can rapidly follow

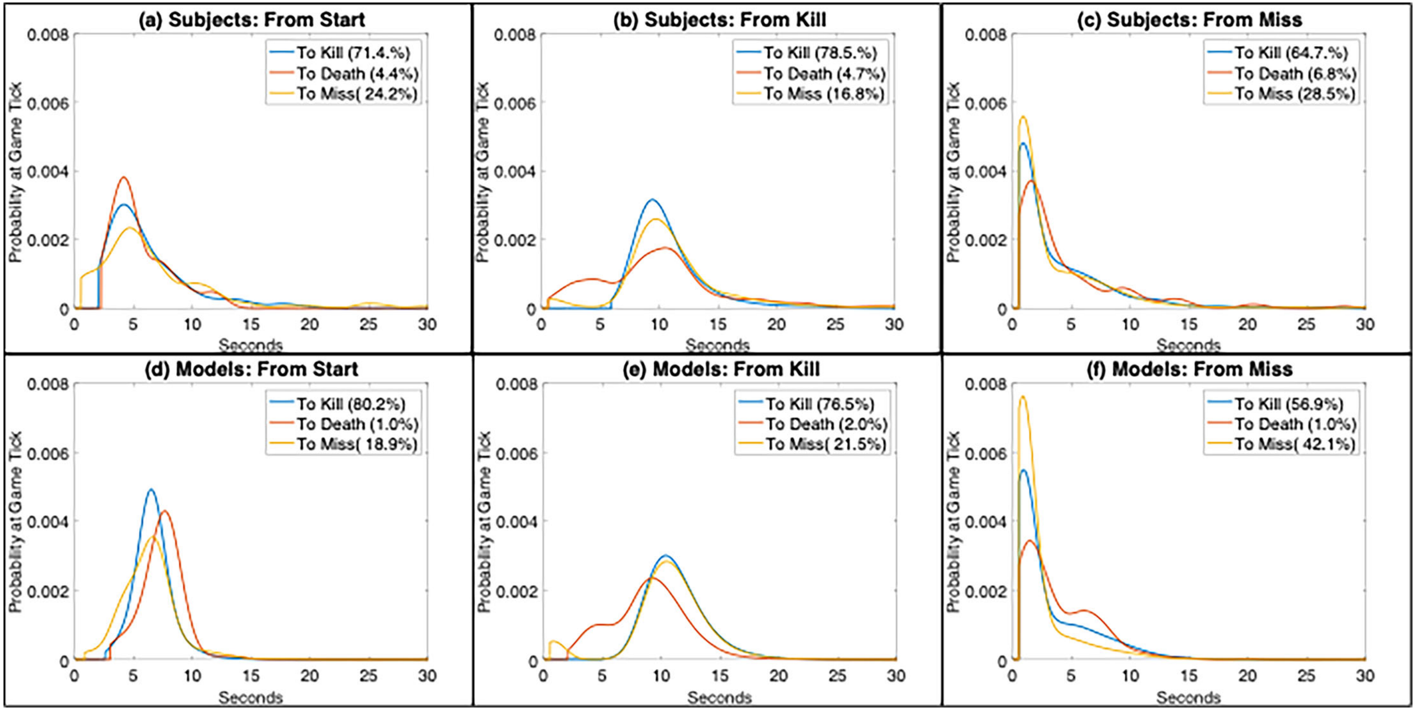

Fig. 5. Probabilities of transitions (given in parentheses) between events and distribution of intervals between events. (a)–(c) give the data from the subjects and (d)–(f) the data from the model.

another (although the HSMM was set to impose a .5 s minimum interval between events). Figs. 5d–f show the corresponding distributions from the model. There are small differences in the distributions, but the overlap is dominant.

While the distributional information shown in Figs. 5a–c is averaged over all subjects, the HSMM only used distributions estimated from other games of the subject in fitting the data for any game from the subject. The HSMM also used conditional probabilities of the EEG that were estimated from other games of that subject. Such per subject estimation allows for tuning to the specifics of a particular subject's game play. The Viterbi algorithm (Rabiner, 1989) was used with the HSMM to identify the maximum likelihood placement of events in the target game. The Appendix explains the probability function that is being optimized by the Viterbi algorithm.

In a semi-Markov model, the probability of the next event depends on the time since the last event. However, it remains Markov in that what happens after an event only depends on that event and not prior events or their timing. This assumption is reasonable for the timing between kills and deaths. After a death, the ship respawns at its starting position and carries with it no effect of what happened before the death. After a kill, critical events prior to the kill have little if any influence on what the next event will be or how long it will be. However, the Markov assumption becomes problematical when misses are included. For instance, suppose a miss intervenes between one kill and another kill. After any kill, the ship must fly outside the hexagon, wait for the fortress to respawn, and re-enter. It matters little where on this journey the ship has its miss. The time of the next kill depends more on the time of an event two back, the prior kill, and not the event one back, the miss.

We used a hierarchical HSMM (Fine, Singer, & Tishby, 1998; Johnson & Willsky, 2013) to avoid violating the semi-Markov assumption. The first HSMM found the most probable locations of kills and deaths ignoring misses. Then, with these anchored, another HSMM placed misses in the intervals using information about the distribution of these misses relative to the other events. This two-pass HSMM identified a mean of 14.62 kills per game compared to a mean of 14.41 in the actual games; a mean of 0.44 deaths per game compared to 0.98 actual deaths, and 1.49 misses compared to 3.81 actual misses. Maximum likelihood estimation is biased against rarer events.

For each game, as in Anderson et al. (2020), the alignment between the assigned events and the actual events was calculated as a combination of a recall and a precision measure (Buckland & Gey, 1994) determined by the locations of kills, deaths, and misses. The measure of recall focused on the events that occurred in the game and identified the closest predicted event. If that predicted event was the same category as the actual event and was within 1.5 s (90 game ticks), it was scored according to how many game ticks it was away—thus, the maximum mismatch score was 90. If the closest event was further away or the closest event was a different category, the match for that event was also scored 90. The average of these recall scores for a game can vary from 0 (perfect match to all actual events) to 90 (worst possible). The measure of precision applied the same scoring procedure but now started with all predicted events and found the closest actual event. Fig. 6 shows the distribution of the recall and precision scores. The recall score tends to be larger (worse) because the reconstructed games tended to have fewer events, leaving fewer potential matches to the actual events.

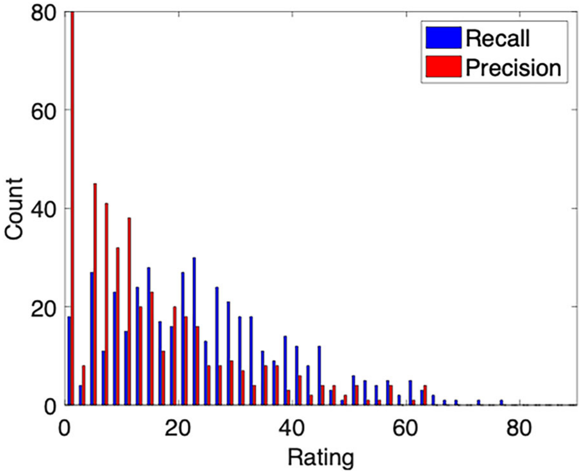

Fig. 6. Distributions of recall and precision ratings on a scale where 0 is perfect match between game and its inferred sketch and 90 is the maximum possible mismatch score.

The average measure of recall was 24.59 and the average measure of precision was 14.88 for an average alignment rating of 19.74. This amounts to being about one-third of a second off on average. The alignment rating of 19.74 between a game and its reconstruction compares to an average alignment of 79.63 to reconstructions of other games. All but two games were best matched by their own reconstructions than any reconstruction from other games. The remaining two games' second best match was their own reconstruction. The accuracy of the reconstructions is considerably better than Anderson et al. (2020), even though that attempt was only localizing events in a 1-min period rather than a 3-min period. The improvement reflects two refinements in the procedure—performing these analyses on a per subject basis and dealing with the violation of the Markov assumption. Aspiring to predict critical events to a 60th of a second is a high standard. Still, 44.6% of all events are being identified to the game tick and 71.4% of the events are being identified to within a tenth of a second (six ticks).

Nonetheless, the reconstructions are not perfect. Even the best reconstruction has a nonzero alignment rating (.24 reflecting all 21 events predicted but five off by one game tick). Fig. 7 illustrates the reconstruction of games whose ratings span from the 5th percentile (better scores) to the 95th percentile (worse scores). As seen in these examples, the best performance is achieved for kills, which are the most frequent events.

Fig. 8 compares the scores subjects got for their games and the score their game reconstructions would have gotten (ignoring bait deaths in both cases). The correlation is strong although the HSMM tends to overpredict subject scores (mean score is 1401 points for HSMM compared to 1315 for subjects). The overestimation reflects the fact the HSMM underestimates the infrequent categories of deaths and misses. Nonetheless, Fig. 8 gives testimony to the ability of the HSMM analysis to identify how well a subject is doing in a particular game.

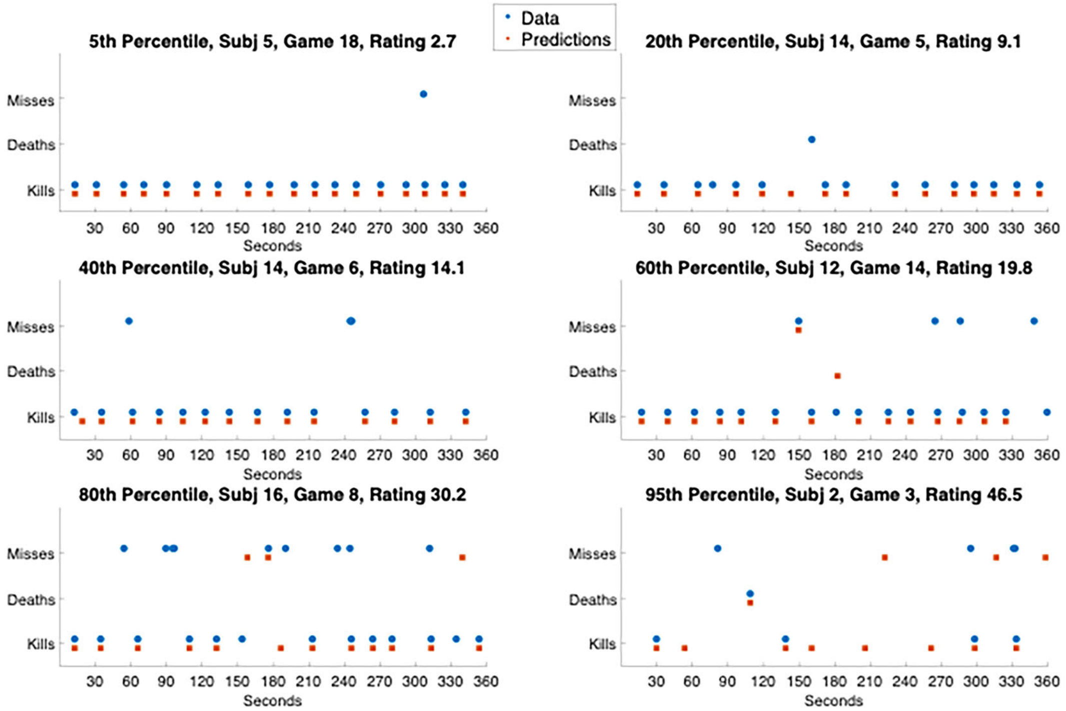

Fig. 7. Examples of differential success in reconstructing the critical events of a game.

3.1.3. *Alternative sources of information*

The good alignment in Section 3.1.2 leveraged information from a particular subject. We considered three other possible sources of information. The first source was information from other subjects rather than other games from the subject. It is possible that information from other subjects might be better because the estimation is based on more than 20 times the data. If subject-specific information is better, it is an indication of substantial differences among subjects. Second, we can merge the data over the three critical events and simply try to identify critical events ignoring their identity. All the temporal distributions are basically unimodal, and all the EEG signals are variants of P300s. The degree to which performance is worse using a single temporal distribution or a single EEG pattern for all critical event types provides a measure of how important it is to distinguish among critical events. Third, to provide a baseline, we used uninformed temporal distributions and EEG patterns that treated all possible[5] intervals as equally likely and all EEG signals as equally reflective of all categories including the Null category.

Table 3a shows the mean alignment ratings for all combinations of these sources for distributional and EEG pattern. Table 3a also shows the results if we estimate the distributional information from the model. In the case of an uninformed source, even if that source of information cannot help localize events, the other source can. Chance performance is given by the one combination where both sources are uninformed. Looking at the effects of different

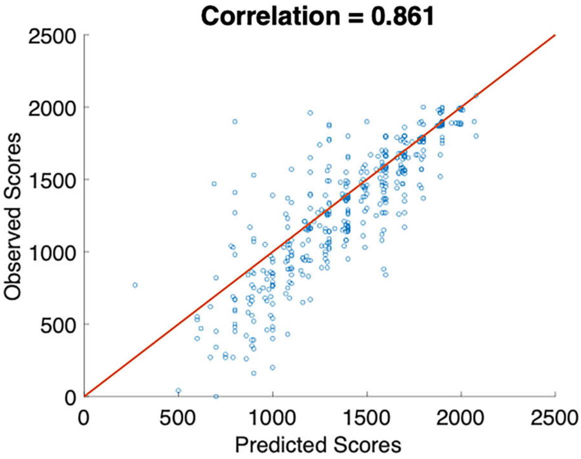

Fig. 8. Relationship between points earned in the actual game versus the EEG-based reconstruction of the game.

sources of EEG pattern information (comparing columns) holding distribution information constant (same row), subject-specific EEG information is always superior to average information from other subjects (for 21 or 22 of the 22 subjects for comparisons across the five rows). We examined the P300 for kills for which there is ample per-subject data. There was between-subject variation in the exact timing of the peak, the magnitude of the signal, and which electrodes showed the maximum response. While magnitude differences might reflect differences in electrode placement and conductivity, timing reflected differences in when subjects had processed the kill. Different subjects had P300 peaks anywhere from 233 ms after the kill to 350 ms.

While using subject-specific EEG patterns is best, in many cases that would not be practical. Using a single EEG pattern estimated from other subjects (third column) tended to be only a little worse than estimating separate EEG for each critical event estimated from other subjects (second column). Only in the case of using model distributional information (row 5) is the difference significant ($t(21) = 2.76$, $p < .05$). Using a single EEG pattern is basically using a generic P300. Both columns 2 and 3 were superior to column 4 (the uninformed case) for all subjects for all rows (distributional choices).

The differences among the distributional choices (comparing rows) are smaller but are important for understanding subject behavior, particularly when the EEG information is good. For all choices of EEG information, using subject-specific distributional information was superior to using information averaged over subjects, which was superior to using information also averaged over events, which was superior to the uninformed case. In most cases, these contrasts were significant at least at the $p < .01$ level. There were two marginal contrasts involving the difference between subject-specific and average-over-other-subjects: when using subject-specific EEG ($t(21) = 1.79$, $p < .1$) and when using uninformed EEG ($t = 1.95$, $p < .1$). Using data averaged from the model (row 5) was no worse than using data averaged from other subjects (row 2) and in two cases significantly better: when using EEG

Table 3
Contribution of different options of information (best possible = 0; worst possible = 90)

(a) Alignment ratings of critical events (max 90)		EEG pattern information			
		Other games of subject	Games of other subjects	Averaged over events	Uninformed
	Other games of subject	19.74	31.75	34.20	76.70
	Games of other subjects	20.39	33.42	36.27	77.65
Distributional information	Averaged over events	22.96	40.08	42.07	80.48
	Uninformed	27.13	44.69	44.46	83.87
	Model games	20.41	32.06	35.15	77.41

(b) Alignment ratings of leg boundaries (max 90)		EEG pattern information			
		Other games of subject	Games of other subjects	Averaged over events	Uninformed
	Other games of subject	43.88	55.02	58.88	64.19
	Games of other subjects	44.94	58.44	63.29	68.30
Distributional information	Averaged over events	46.00	59.97	64.58	66.81
	Uninformed	53.95	64.50	67.05	67.97
	Model games	47.11	60.74	64.14	69.02

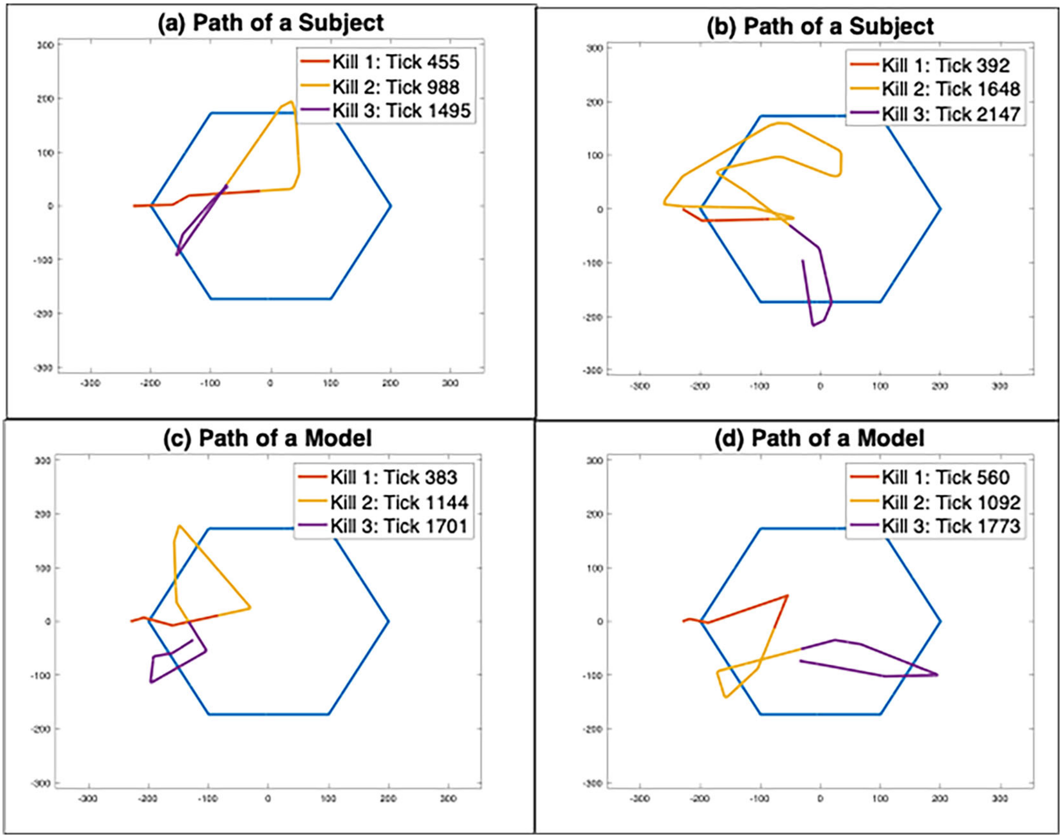

Fig. 9. Examples of beginning flight paths up to the third kill: (a) and (b) show subjects; (c) and (d) show models.

patterns of other subjects (column 2, $t(21) = 3.13$, $p < .01$) and when using a single EEG pattern for critical events (column 3, $t(21) = 4.42$, $p < .005$). This supports the conclusion that the model accurately captures average subject behavior and may even be capturing things missing in a sample of 21 subjects.

3.2. Operations: Legs of flight

Fig. 9 illustrates the flight paths that two subjects and two models took at the beginning of games while achieving the first three kills. The three segments (from start to first kill, from kill to kill, and from kill to kill) are color-coded. The individual segments contain linear legs of flight that are usually quite apparent in such a display. These legs are connected by brief periods of turning which are too short to easily discern in the flight path when examined on the scale of Fig. 9.[6] The Appendix describes the algorithm for identifying legs of a flight. In the model, and we assume in subjects, a switch from one leg to another reflects a change in subgoal: there are legs that get the subject into the hexagon, legs to get behind the fortress, legs to get out of the hexagon, legs to avoid crashing, and legs to correct these legs when they turn out not to achieve their subgoal (sometimes because of unexpected actions of the bait). In the model and we assume in subjects, the recognition that a new leg is required begins toward the end of the prior leg and typically the model will turn the direction in which the ship is aiming before thrusting to begin the turn. Thus, the beginning of the subgoal is before the beginning of the turn, but the beginning of a turn is a marker of a new navigation subgoal set a little bit earlier.

The mean times for these legs and their preceding turns (1.12 s in subjects, 1.17 s in model) are close to the 1 s duration that Newell associated with the Operation Level. The navigation subgoals associated with legs are similar to Newell's definition of an operation as a sequence of actions put together to achieve a response function. Therefore, we make this correspondence between legs and Newell's Operation level.

We will focus our analysis of leg structure on segments of behavior that go from a kill to another kill (included would be kills 2 and 3 in each of Fig. 9a–d). On average, 61% of game time is spent in such segments. These segments reflect the successful completion of a unit task that achieves a kill. These intervals start with a segment where the subject continues the path that led to a kill, followed by an alternating sequence of turns and legs which get the ship out of the hexagon and then back into the hexagon, and end with the segment of a leg that takes the ship to the next kill. An analysis of how these components add up to a kill provides a case of looking at the decomposition of a unit task into operations.

Fig. 10 shows the distributions of these durations in subjects and models. The starting segments, the contained legs, and the ending segments are all about a second that we associated with operations. (The turn can be considered part of the next leg that gets the ship to the next desired position.) While there are differences between subjects and models for all intervals, the striking differences involve the starting segments (Fig. 10a) and the turns (Fig. 10d). There are no brief starting segments in the model because it waits until confirming a kill before planning its next leg. Some subjects tend to behave similarly: 9 of the 22 subjects have less than 5% (average 2.3%) of their starting segments under 50 game ticks. On the other hand, seven subjects have more than 40% (average 70.4%) of their starting segments under 50 game ticks. The turn discrepancy involves the very long turns that subjects sometimes show, but models do not. Compared to the discrepancy involving starting segments, there is less variation among subjects in this measure. Only two subjects show less than 5% (mean 2.1%) of their turns over 50 ticks and only one shows more than 40% of these long turns (47.3%). Subjects who deviate more from the model on one measure tend to deviate more on the other measure—correlation across subjects is −.623 between mean length of starting segments and mean length of turns.

3.2.1. EEG analysis

We investigated whether there was any EEG signal that corresponded to the leg structure of a game. Legs are segmented by turns. A turn involves a sequence of keying actions that will send the ship on a new path. The beginning of a leg is defined as the end of that turn sequence and the end of a leg is defined as the beginning of the next turn sequence. Figs. 11a and b show the EEG patterns associated with the beginning and end of legs. These patterns are much weaker than the patterns associated with the critical events (Fig. 3). Table 4 shows the

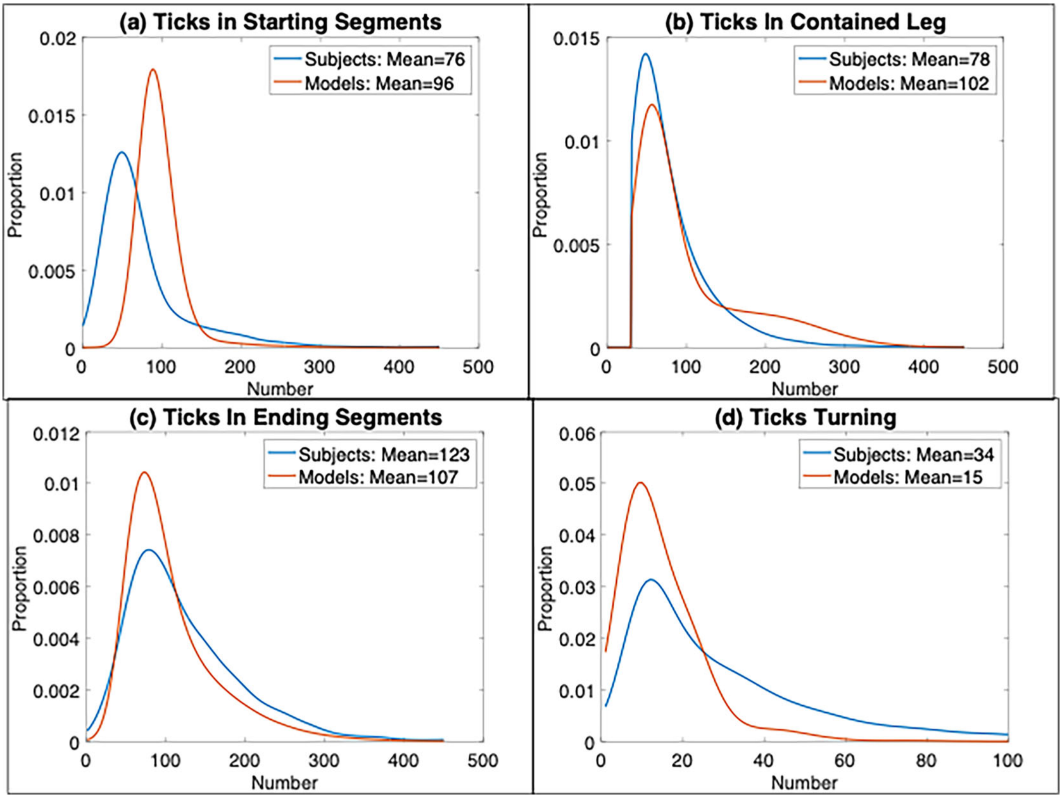

Fig. 10. Comparison of subjects and models in terms of length (number of game ticks) of their leg structures.

results using the same classification methodology that was used for critical events (Table 2). Despite the weaker signal, there is still some discriminability among the categories. The classifier only uses other games of a subject to classify a target game and these may have some distinctive properties not in the averages over subjects displayed in Fig. 11. Figs. 11c and d attempt to display the basis for the discriminability in the average data. Fig. 11c shows the average pattern displayed by EEG activity across both begin-leg and end-leg events, which is part of the basis for discriminating these points from other Null points. Fig. 11d displays the difference between the activity patterns for the beginning and end of legs. The activity in frontal electrodes is more negative 100 ms after the end of a leg than 100 ms after the beginning of a leg. The negative dips at point 35 (.13 s) for electrode FZ and CZ are significant (mean of points 33 and 37 minus point 35, $t(21) = 4.62$ and 4.38, $p < .0005$).

3.2.2. *Using an HSMM to localize legs*

As in the analysis for the unit tasks, we investigated how well we could combine distributional information (Fig. 10) and EEG information (Fig. 11) to determine where the legs were in the game. We applied such an HSMM to each kill-to-kill sequence within each game (overall there are 4495 such sequences). After a kill, there would be a starting segment

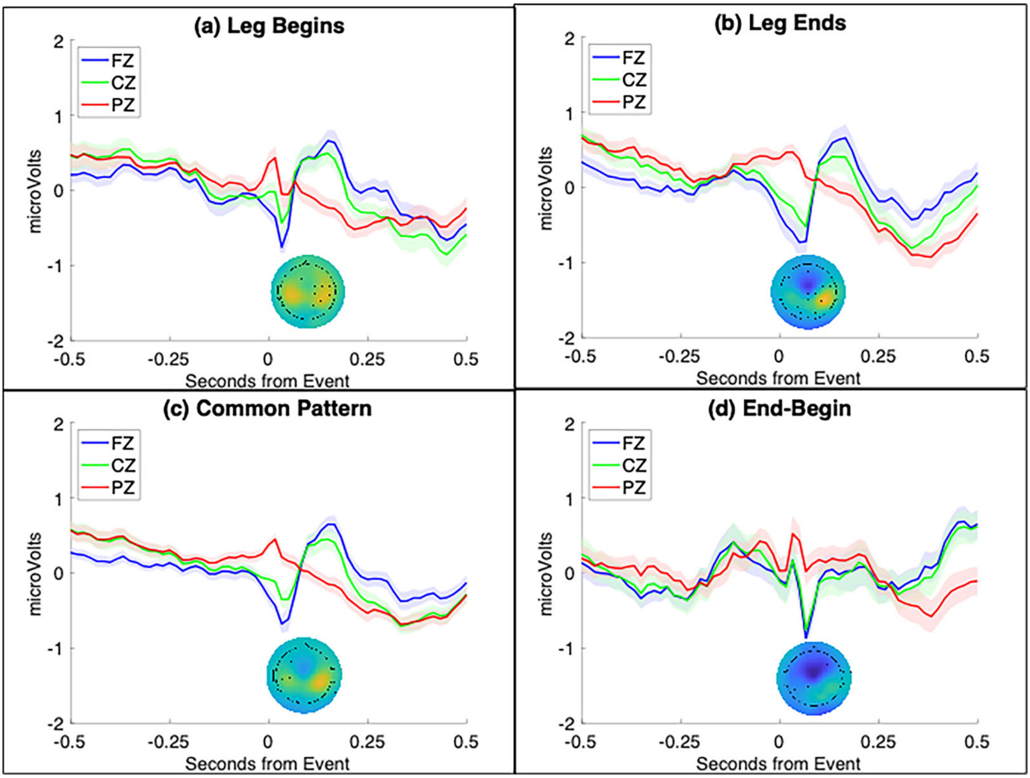

Fig. 11. (a) EEG activity (mean values and standard errors) of three central electrodes from a half second before the beginning of a leg; (b) EEG activity (mean values and standard errors) of three central electrodes from a half second before the end of a leg; (c) common pattern; (d) difference between ends and beginnings. The scalp profiles in all cases illustrate activity at .1 s after the event and are scaled to range between −1 and +1 microvolts.

(distribution in Fig. 10a) terminated by the beginning of a turn (distribution in Fig. 10d, EEG pattern Fig. 11b), then some number of alternating legs and turns (distributions in Figs. 10b and d) marked by EEG signals (Figs. 11a and b), and then a final ending segment (distribution in Fig. 10c, EEG pattern Fig. 11c). As was the case for critical events, we estimated the temporal distributions separately for each game from other games of the subject and the EEG patterns for a game from other games of the subject. In addition, there was the probability that there would be different numbers of turns defining legs. Minimally, there must be one turn of the ship so that it can go back in the hexagon after exiting the hexagon. On average, there were 4.7 turns. We modeled the number of turns more than one as a Poisson distribution with a parameter estimated from the mean number of turns for other games of the subject.

The HSMM produced the most probable location of the beginnings and ends of legs or equivalently the ends and beginnings of turns. Fig. 12 shows examples of the reconstructions with periods of turning marked at the top and the legs marked on the bottom. Note the durations on the x-axes vary depending on how long the kills took. We rated these reconstructions with the same rating scheme as we had the whole game reconstructions (Fig. 7). The

Table 4
Classification of leg boundaries

(a) Classification by likelihood		Labeled by classifier		
		Null	Begin	End
True class	Null	1,936,984	436,301	482,795
	Begin	6586	11,601	2834
	End	7473	2838	10,720

(b) Classification by posterior		Labeled by classifier		
		Null	Begin	End
True class	Null	2,845,761	5411	4908
	Begin	19,653	1343	25
	End	20,125	22	884

(c) Pairwise dprimes		Class 2	
		Begin	End
Class 1	Null	1.19	1.00
	Begin		1.38

(d) Pairwise AUC		Class 2	
		Begin	End
Class 1	Null	0.807	0.768
	Begin		0.846

average rating was 43.88 which compares to the average of 19.74 for critical events. The poorer reconstruction reflects the weaker EEG signals. The examples in Fig. 12 illustrate the range of quality of reconstructions according to this metric. The HSMM misses some turns and hallucinates others. Reflecting the similarity of the signal for beginnings and ends of turns, it sometimes mistakes an end for a beginning or vice versa. Still, as the next section will show, the HSMM is doing much better with the information that it has than it would do with no information.

3.2.3. Alternative sources of information

One way of assessing the performance in the previous section is to consider how well an HSMM would do given alternative sources of information. We considered the same combinations of sources as we had for critical events. For EEG pattern information, we considered four possible sources: other games of the subject, games of other subjects, averaging EEG signals for beginnings and ends (i.e., what is illustrated in Fig. 11c), and an uninformed pattern that considered any EEG signal equally representative of beginnings, ends, and Null events. For distributional information, we considered five possible sources: other games of that subject, games of other subjects, averaging the beginnings and ends of turns to have a single distri-

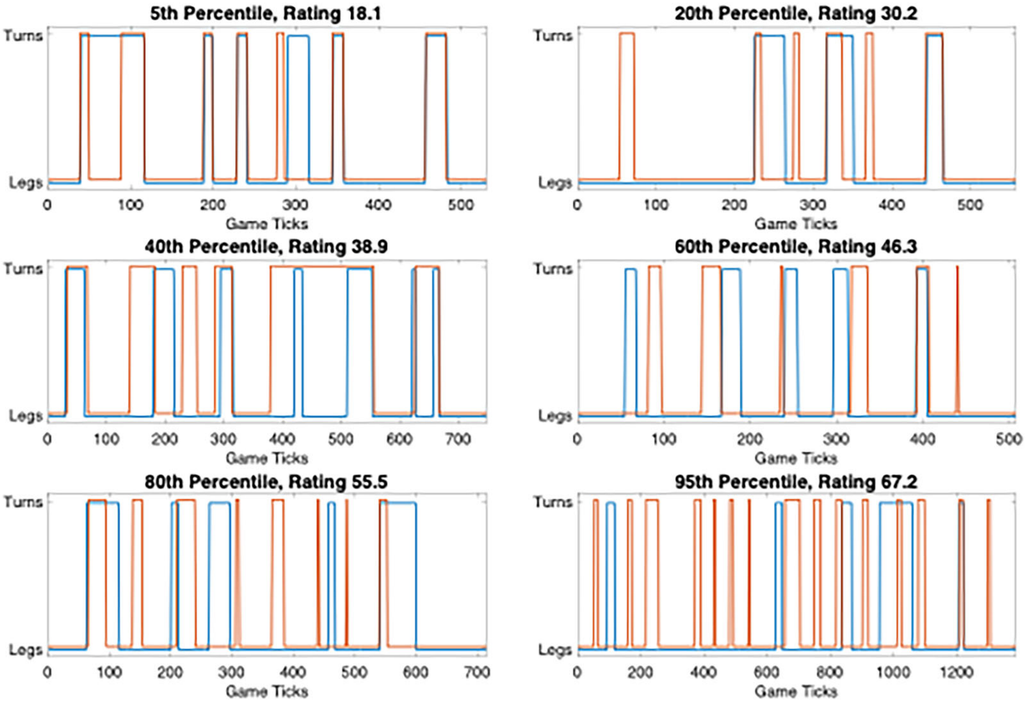

Fig. 12. Examples of differential success in reconstructing the turns and legs in kill-to-kill intervals.

bution of events, using an uninformed distribution where all possible durations are equally probable, and estimating distributions from the model.

The alignment ratings for leg structure are given in Table 3b. Comparing columns in Table 3b for legs, there is the same pattern we see in Table 3a for critical events: Using subject-specific information is superior to using information from other subjects, which is superior to using information averaged over beginnings and ends of turns, which is superior to the uninformed case. The differences are significant at .001 or lower for any contrast that holds constant sources of distributional information. The first four rows in Table 3b also show the same pattern as in Table 3a: using subject-specific distributional information is superior to using distributional information averaged from other subjects, which is superior to averaging distributional information over events, which is superior to the uninformed case. With only one exception, these distributional contrasts are significant holding constant source of EEG information. The one exception occurs in the case of an uninformed EEG signal: the contrast between distributional information averaged over subjects with distributional information also averaged over other all events. In this case, the contrast is nonsignificantly reversed. Unlike the pattern for critical events, the alignment ratings using distributional information from other subjects are significantly better than the alignment ratings using distributional information from the models. In the cases of using subject-specific EEG patterns or EEG patterns averaged over subjects, the significance levels of the contrast are less than .0005. This is

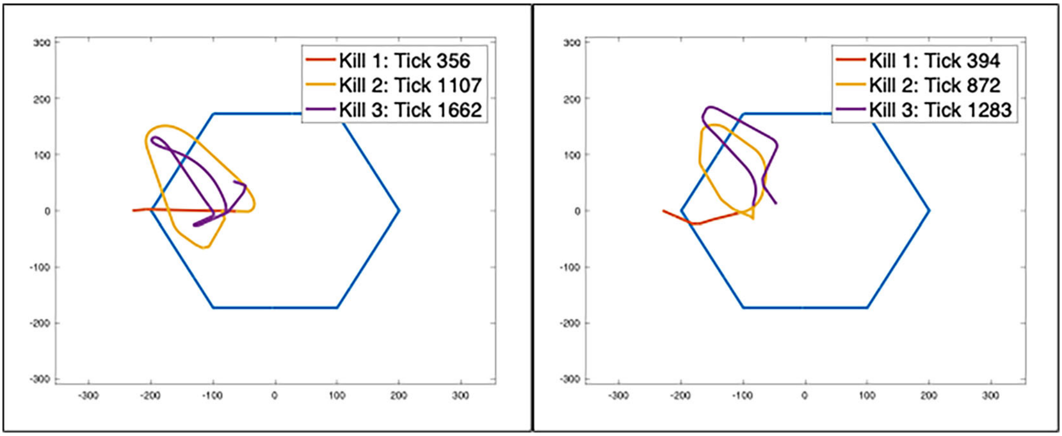

Fig. 13. Two examples of curved flight from a subject with the highest tendency to overlap turn and thrust keys.

consistent with the conclusion from Fig. 10 that there are major differences between the models and subjects at this level.

3.3. *Approaching deliberate acts: Individual keying actions*

In Newell's temporal hierarchy, deliberate acts are the smallest temporal level where information is being brought together to enable a step of cognition. This aligns with a production firing in ACT-R which involves reacting to a cognitive pattern of information. This is the point where parallel processing of information at finer grain sizes becomes serial. It is something that Anderson et al. (2016) tried to identify with a bump analysis of brief experimental trials, but as we noted in the introduction that approach breaks down for longer periods of time. The current approach involves aligning the EEG analysis with specific events in the game or actions of the subject, but many production rules do not leave any such behavioral markers. Nevertheless, we had hoped to extend our approach toward the deliberate act level by looking at the keying actions which are associated with individual productions. They occur at the rate of 1 per 323 ms during a leg and subsequent turn. Our thought was to identify EEG signals for pressing the three keys (clockwise turn, counterclockwise turn, thrust) and the release of these keys. We had hoped to combine the EEG information with information about their temporal distribution and transition probabilities to apply an HSMM at this finer temporal grain size.

However, when we examined the individual keying actions, we found a discrepancy between subjects and models that frustrated both detailed comparison of subject and model and blocked a simple EEG analysis. Subjects sometimes press and release turn keys (either clockwise or counterclockwise) while they are holding down the thrust key or press and release thrusts while holding down a turn key. The model follows a conservative navigation strategy that requires one key to be released before initiating a press of another. Figs. 13a and b display the flight paths for the first three kills in two games of the subject with the strongest tendency to overlap keys (pressed a turn key 47% of the time while pressing thrust). As can be seen, these flight paths contain long curved trajectories, although there are also linear legs.

Nine of the 22 subjects showed no instances of such curved trajectories,[7] while four subjects had more than 10. The subject illustrated in Fig. 13 had 209 curved trajectories across the 20 games. The algorithm that identifies leg structure identifies these trajectories as long turns (Fig. 10d). The correlation across subjects between proportion key overlap and proportion of turns longer than 60 ticks is .74.

The overlapping of key presses meant that it was not possible to simply identify EEG patterns associated with individual key actions. Different key actions were sometimes happening nearly simultaneously. Also, it is problematic to treat an action for one key while another was being held down (e.g., pressing the clockwise key down while the thrust key was being held down) as the same as that key action in isolation. While there might be informative analyses of the EEG signals associated with such keying combinations, they were not ones that would further the decomposition of operations into deliberate acts.

4. Conclusions

The successes and difficulties we had in our analyses point to the appropriate level within Newell's band for tracking open-ended tasks. The strong success involved identifying critical events at the highest Unit-Task level. Killing a fortress has the characteristics of a Unit Task (Card et al., 1983). Misses and deaths are failed Unit Tasks. The EEG-based MVPA did well identifying the boundaries of these unit tasks. The HSMM-MVPA analysis takes advantage of both the constraints on the timing of these critical events and the strong EEG signals associated with their appearance. The improved success over Anderson et al. (2020) reflects focusing on subject-unique patterns in classification and satisfying the semi-Markov assumption. The ACT-R model at this level also seemed a good characterization of subject behavior. While it was not as good as using subject-specific behavioral patterns in the HSMM, it was as good as using data from other subjects.

Many but not all open-ended tasks decompose into a unit-task structure. After a unit-task, one can judge whether one has made progress toward the overall goals of the open-ended tasks. There is reason to believe that one can pursue such HSMM-MVPA analysis at the Unit-Task level for such open-ended tasks. If the task decomposes into Unit Tasks (like the classic Card, Moran, and Newell text editing), each Unit Task will result in either success or failure which should produce a strong EEG signal. Often, the next Unit Task and how long that takes depends only on the last Unit Task, satisfying the semi-Markov assumption. However, the independence of Unit Tasks may only be partial, leaving violations of the semi-Markov assumption as in this task. At least sometimes, these violations can be dealt with by going beyond a single HSMM as we did here with a hierarchical HSMM. Recently, Anderson, Fincham, Fox, Stevens, and Swan (2023) have extended the HSMM-MVPA approach to identify critical events in the Mat-B task (Miller, Schmidt, Estepp, Bowers, & Davis, 2014) focusing on the correction of faults which appear as Unit Tasks in a 6-min task. The structure of that experiment creates long-term dependencies between different types of faults. Rather than a hierarchical HSMM like used here, we used parallel HSMMs tracking different faults to remove these dependencies.

We have found that the end of a unit task is associated with a strong EEG signal. These EEG signals can be seen as variants of the P300 ERP (Fig. 3) which are often used in brain-computer interface (BCI) applications (Fazel-Rezai et al., 2012). As the critical events reflect positive or negative outcomes, one can ask how they relate to the ERN and FRN (error-related negativity and feedback-related negativity (Holroyd & Coles, 2002; Walsh & Anderson, 2012), which appear as a negative frontocentral deflection 80–150 msec after an error and 200–350 msec after negative feedback. Such a negative deflection can be found in the vertical dimension that separates negative from positive events (Fig. 4c). There is approximately a 2.5 negative microvolt deflection in the FZ and CZ electrodes about .25 s after the event. This is dwarfed by the following large positive deflection reflecting that dimension's P300.

The results were more mixed when we went down to the Operation level of Newell's time bands. The EEG signals associated with the beginning and ends of turns are weaker resulting in poorer identification of leg structure. Still, identification was much better than chance. Once again, we found that using subject-specific distributional information and EEG patterns led to better results. However, unlike the unit-task level, there were substantial differences between the behavior of subjects and models (Figs. 10a and d). Reflecting these mismatches, using distributional information from the model was not as good as using distributional information from other subjects.

Results continued to deteriorate when we attempted to approach the deliberate act level by identifying individual keystrokes. Deliberate acts occur serially one after another. While overlapping key actions were a minority of the cases, they identified a problem with associating individual keystrokes with deliberate acts. Subjects were sometimes performing actions that involved keystroke combinations. This was a behavior that the model did not display. Their emergence for some subjects in this game probably reflects the greater navigational demands and freedom in Co-op Space Fortress.

There were substantial individual differences among subjects that went beyond just success at the game. Different subjects had different styles in how they went about achieving success in the game and this is why using subject-specific statistics led to better results in the HSMM reconstructions. Also, using subject-specific EEG patterns led to better results. Some of the EEG differences might just reflect differences in electrode placement and conductance, but they also reflected differences in how the subject processed the game such as when subjects responded to changes in game state. It should be noted that there were also considerable differences among individual model runs, reflecting both differences in parameter settings but also differences in what the model learned in that run because of the random differences in game play. The model variation matched the subject variation at the unit-task level, but it did not extend down to capture all the detailed differences in how subjects performed the unit tasks.

Both from behavioral comparison and its use to reconstruct events from EEG, the model of Dimov et al. (2023) is a good model of average subject behavior at the unit task level but not lower levels. Lower-level details such as the use of overlapped keys to fly curved paths show considerable individual differences. The model could be elaborated to have this behavior, but it would have to be an option that some models did, and others did not. The model was originally designed to capture the higher level of subject behavior (data on the scale of Fig. 2) in a complex game and was focused on the emergence of cooperation between two human subjects. Extending the model to capture lower-level detail and individual differences would not further these goals. Like all modeling efforts, there is a level of analysis appropriate to the goals of that effort and for this effort that level is the unit-task level. It is conceivable that for other purposes, one would want a model that matched the detailed behavior of subjects even if it ignored how the pieces came together in a way to achieve success in the game.

The unit-task level is most critical for the research questions that focus on the organization of behavior in open-ended tasks. As seen in this study, individuals can vary in how they achieve unit tasks without consequence for their overall success. The unit-task level is also the right level of aspiration for detailed reconstruction of open-ended tasks from EEG. Completion of one unit-task frees the performance of the next unit-task from the details of past history. We were able to use semi-Markov methods for interpreting the EEG signal, which require independence from past history. In addition, the EEG signal associated with such tasks tends to be sufficiently robust to stand out against the noise. It is worth noting that while unit-tasks occupy seconds of task, the HSMM analysis allowed us to identify their location with an accuracy that matched the temporal grain size of deliberate acts.

While they are not the best-performing possibilities in Table 3a for unit tasks, there is special potential in the two cases in Table 3a (row 5, columns 2 and 3) where a computational model is providing the distributional information, and we are using average signal patterns (separately for different events or just a generic P300). In many situations, it would not be feasible to perform the per-subject training to obtain subject-specific estimates of the signals. In an applied context, one might also not have the data to estimate per-subject distributional information. In a theoretical context, one might want to evaluate a model over many subjects without running any subject enough to get subject-specific signal estimates.

In summary, understanding an open-ended task like the video game depends on understanding how its unit tasks are achieved. Dimov et al. developed a model that learned to fly so that it quickly achieved kills and avoided deaths like the successful subjects. However, within the constraints of efficient and safe flight, there is room for variation and subjects showed more variations than the model did. Capturing this variation at the operation and deliberate act level is not important to understanding the open-ended task as a whole. It is fortunate from this perspective that the strong EEG signals were associated with the unit tasks. As a consequence, it is possible to use EEG to track the key aspects of performance in an open-ended task.

Acknowledgments

This work was supported by the Office of Naval Research Grant N00014-21-1-2586. We thank Henry Liu for his work in data collection. The data and analyses which were used to create the figures in the paper are available at https://osf.io/kuyrz/?view_only=84139448d94549408eb84b109c7128dd

Notes

[1] Available for demo at https://osf.io/kuyrz/?view_only=84139448d94549408eb84b109c7128dd

[2] Available at https://osf.io/kuyrz/?view_only=84139448d94549408eb84b109c7128dd

3 This is calculated by collapsing the *n* x *n* matrix into *n* 2x2 matrices representing presence versus absence of the category and averaging the d-primes for the *n* matrices.

4 Using the Matlab function ksdensity.

5 The HSMM only considered intervals of at least 30 ticks and no more than 1800.

6 Turns often take some time, but typically involve deceleration and so not much distance is covered.

7 Defined as holding thrust and turn down at least 30 ticks, the length of the shortest linear leg.

References

Abiri, R., Borhani, S., Sellers, E. W., Jiang, Y., & Zhao, X. (2019). A comprehensive review of EEG-based brain–computer interface paradigms. *Journal of Neural Engineering*, *16*(1), 011001.

Altarelli, I., Green, C. S., & Bavelier, D. (2020). Action video games: From effects on cognition and the brain to potential educational applications. In *Educational neuroscience* (pp. 273–297). Routledge.

Anderson, J. R. (2002). Spanning seven orders of magnitude: A challenge for cognitive modeling. *Cognitive Science*, *26*, 85–112.

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.

Anderson, J. R., Betts, S., Bothell, D., Hope, R., & Lebiere, C. (2019). Learning rapid and precise skills. *Psychological Review*, *126*, 727–760.

Anderson, J. R., Betts, S., Bothell, D., & Lebiere, C. (2021). Discovering skill. *Cognitive Psychology*, *129*, 101410.

Anderson, J. R., Betts, S., Fincham, J. M., Hope, R., & Walsh, M. W. (2020). Reconstructing fine-grained cognition from brain activity. *Neuroimage*, *221*, 116999.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of mind. *Psychological Review*, *111*, 1036–1060.

Anderson, J. R., Fincham, J. M., Fox, E. L., Stevens, C. A., & Swan, G. (2023). Using EEG to understand multitasking performance. In *Annual Meeting of the Society for Mathematical Psychology*.

Anderson, J. R., Zhang, Q., Borst, J. P., & Walsh, M. M. (2016). The discovery of processing stages: Extension of Sternberg's method. *Psychological Review*, *123*, 481–509.

Baldwin, C. L., & Penaranda, B. N. (2012). Adaptive training using an artificial neural network and EEG metrics for within- and cross-task workload classification. *Neuroimage*, *59*(1), 48–56.

Boot, W. R. (2015). Video games as tools to achieve insight into cognitive processes. *Frontiers in Psychology*, *6*, 3.

Borst, J. P., & Anderson, J. R. (2021). Discovering cognitive stages in M/EEG data to inform cognitive models. In *An introduction to model-based cognitive neuroscience*.

Buckland, M., & Gey, F. (1994). The relationship between recall and precision. *Journal of the American Society for Information Science*, *45*(1), 12–19.

Card, S., Moran, T., & Newell, A. (1983). *The psychology of human−computer interaction*. Hillsdale, NJ: Erlbaum.

Delorme, A., & Makeig, S. (2004). EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, *134*(1), 9–21.

Dimov, C. M., Anderson, J. R., Betts, S. A., & Bothell, D. (2023). An integrated model of collaborative skill acquisition: Anticipation, control tuning, and role adoption. *Cognitive Science*, *47*(7), e13303.

Fazel-Rezai, R., Allison, B. Z., Guger, C., Sellers, E. W., Kleih, S. C., & Kübler, A. (2012). P300 brain computer interface: Current challenges and emerging trends. *Frontiers in Neuroengineering*, *5*, 28055.

Fincham, J. M., Lee, H. S., & Anderson, J. R. (2020). Spatiotemporal analysis of event-related fMRI to reveal cognitive states. *Human Brain Mapping*, *41*, 667–683.

Fine, S., Singer, Y., & Tishby, N. (1998). The hierarchical hidden Markov model: Analysis and applications. *Machine Learning*, *32*, 41–62.

Gianferrara, P. G., Betts, S., Bothell, D., & Anderson, J. R. (2021). Simulating human periodic tapping and implications for cognitive models. In *Proceedings of the 19th International Conference on Cognitive Modelling*.

Gray, W. D. (2017). Game-XP: Action games as experimental paradigms for cognitive science. *Topics in Cognitive Science*, *9*(2), 289–307.

Holroyd, C. B., & Coles, M. G. (2002). The neural basis of human error processing: Reinforcement learning, dopamine, and the error-related negativity. *Psychological Review*, *109*(4), 679.

Jap, B. T., Lal, S., Fischer, P., & Bekiaris, E. (2009). Using EEG spectral components to assess algorithms for detecting fatigue. *Expert Systems with Applications*, *36*(2), 2352–2359.

Johnson, M. J., & Willsky, A. S. (2013). Bayesian nonparametric hidden semi-Markov models. *Journal of Machine Learning Research*, *14*, 673–701.

Kotseruba, I., & Tsotsos, J. K. (2020). 40 Years of cognitive architectures: Core cognitive abilities and practical applications. *Artificial Intelligence Review*, *53*(1), 17–94.

Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A., & Yger, F. (2018). A review of classification algorithms for EEG-based brain–computer interfaces: A 10 year update. *Journal of Neural Engineering*, *15*(3), 031005.

Mané, A., & Donchin, E. (1989). The space fortress game. *Acta Psychologica*, *71*(1–3), 17–22.

Miller, Jr, W. D., Schmidt, K. D., Estepp, J. R., Bowers, M., & Davis, I. (2014). *An updated version of the US Air Force Multi-Attribute Task Battery (AF-MATB)*. Technical report. Air Force Research Lab Wright-Patterson AFB OH Human Performance Wing (711th) - Human Effectiveness Directorate/Applied Neuroscience Branch.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Cambridge University Press.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286.

Rahman, R., & Gray, W. D. (2020). SpotLight on dynamics of individual learning. *Topics in Cognitive Science*, *12*(3), 975–991.

Verleger, R. (2020). Effects of relevance and response frequency on P3b amplitudes: Review of findings and comparison of hypotheses about the process reflected by P3b. *Psychophysiology*, *57*(7), e13542.

Walsh, M. M., & Anderson, J. R. (2012). Learning from experience: Event-related potential correlates of reward processing, neural adaptation, and behavioral choice. *Neuroscience & Biobehavioral Reviews*, *36*(8), 1870–1884.

Wickens, T. D. (2002). *Elementary signal detection theory*. New York: Oxford University Press.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Supporting Material

Appendix

Bot variability

Bot variability comes from uniform noise added to its actions (key presses). The mean durations of the actions are based on their durations in the ACT-R model. All actions involve a cognitive phase that can vary from 4 to 8 ticks and a motor execution time which varies from 80% to 120% of the intended time. In addition, most turns are followed by thrust, and in these cases, a variable interval of 6–12 ticks is inserted between the turn and thrust. While all sources of variability impact when the bot takes an action, variability in the duration of a keypress will also affect the direction and speed of the bot.

Probability maximized by the Viterbi algorithm

Any sequence of events (kills, deaths, misses) can be denoted as events a_1, a_2, …, a_n occurring at game ticks t_1, t_2, …, t_n, where a_1 is the start of the game (hence t_1 is the first game tick 1), and a_n is the last event before the last game tick 10,800. The following gives the probability of the sequence assuming it satisfies the semi-Markov assumption:

$$Prob = \left(\prod_{i=1}^{n-1} tr\left(a_i.a_{i+1}\right) * f\left(t_{i+1} - t_i | a_i, a_{i+1}\right) * P\left(EEG(t_i + 1 : t_{i+1} | a_{i+1}\right) \right)$$
$$*S\left(10800 - t_n | a_n\right) * P\left(EEG(t_n + 1 : 10800 | Null\right) \qquad \text{(A1)}$$

where $tr(a_i,a_{i+1})$ is the probability of transition between the events a_i and a_{i+1}, $f(t_{i+1} - t_i| a_i, a_{i+1})$ is the probability of the $t_{i+1} - t_i$ game ticks between the events a_i and a_{i+1}, $P(EEG(t_i+1,:t_{i+1}) \mid a_{i+1})$ is the conditional probability of the EEG signal for this period if it ends in a_{i+1}, and $S(10800\text{-}t_n|a_n)$ is the probability that there are no critical events in the remaining game ticks if the last event is a_n. While calculating this quantity for all possible sequences is prohibitive, the Viterbi algorithm uses dynamic programming to efficiently identify the most probable sequence. Anderson et al. (2020) describe a further efficiency that allows us to ignore the probability of the EEG signal for Null ticks and deals with the nonindependence of the EEG signal on adjacent ticks.

Legs of flight

A thrust will send the ship on a new path. Therefore, legs of flight begin when the thrust key is lifted up, leaving flight direction unchanged, and they end when the thrust key is depressed, changing flight direction. Turns are often achieved by thrusting, then turning the ship some more, and then thrusting again to achieve a new path. This can result in a short period of constant flight direction between the two thrusts. To avoid treating these as true legs, we required that legs be at least 30 ticks (half a second long) and considered the shorter legs part of a turn.