Learning Behavior-Grounded Event Segmentations

Christian Gumbsch (christian.gumbsch@student.uni-tuebingen.de)

Jan Kneissler (jan.kneissler@uni-tuebingen.de)

Martin V. Butz (martin.butz@uni-tuebingen.de)
Chair of Cognitive Modeling, Department of Computer Science and Department of Psychology, Faculty of Science,
Eberhard Karls University of Tübingen, Tübingen, Germany

Abstract

The event segmentation theory (EST) postulates that humans systematically segment the continuous sensorimotor information flow into events and event boundaries. The basis for the observed segmentation tendencies, however, remains largely unknown. We introduce a computational model that grounds EST in the interaction abilities of a system. The model learns events and event boundaries based on actively gathered sensorimotor signals. It segments the signals based on principles of probabilistic predictive coding and surprise. The implemented model essentially simulates, anticipates, and learns event progressions and event transitions online while interacting with the environment by means of dynamic, predictive Bayesian models. Besides the model's event segmentation capabilities, we show that the learned encodings can be used for higher-order planning. Moreover, the encodings systematically conceptualize environmental interactions and they help to identify the factors that are critical for ensuring interaction success.

Keywords: event models; object interaction; predictive encoding; event segmentation; higher order planning; factorization; conceptualization

Introduction

The embodiment turn in cognitive science has emphasized the importance of simulating relevant aspects of the outside environment by means of perceptual symbol systems (Barsalou, 1999). To enable motor-grounded simulations, the inclusion of actions was emphasized (Engel, Maye, Kurthen, & König, 2013). Moreover, the importance of explicit forms of predictions and anticipations has been emphasized, supporting both, cognitive development (Barsalou, Breazeal, & Smith, 2007) and adaptive, goal-directed behavior (M. Botvinick & Weinstein, 2014; Butz, Sigaud, & Gérard, 2003; Sigaud, Butz, Pezzulo, & Herbort, 2013). In fact, recent treatises suggest that predictive coding and anticipations may form the foundations that bring about embodied cognition (Clark, 2013; Friston, 2009; Hohwy, 2013). In this paper, we present an algorithm that models an anticipatory learning system, which develops suitable compositional structures to interact with the environment adaptively and goal-directedly.

The *event segmentation theory* (EST) (Zacks & Tversky, 2001; Zacks, Speer, Swallow, Braver, & Reynolds, 2007) suggests that humans tend to structure the stream of sensory perceptions into events and event transitions. Events were characterized as "a segment of time at a given location that is conceived by an observer to have a beginning and an end" (Zacks & Tversky, 2001, p. 3). In various studies that focused on event structure perception, it was shown that events are characterizable as relatively uniformly unfolding interactions, whereas event boundaries are characterized by sudden, strongly non-linear changes in the unfolding events. While some of these changes seem to be strongly related to movement variables, movement variables alone could not account for all the segmentations that humans indicated (Zacks, Kumar, Abrams, & Mehta, 2009). We propose that event segmentations may be grounded in, and develop from, own sensorimotor experiences.

To investigate this proposition, we introduce a computational cognitive model implementation, which is based on Zacks et al. (2007)'s schematic EST model. The implemented system learns how it is able to manipulate objects solely by actively processing sensorimotor interactions. The system essentially develops a predictive world model, which segments the gathered sensorimotor experiences into events and event transitions from scratch. Events are sets of forward models that are active over an extended period of time while interacting with the environment. Event boundaries mark the beginning and ending of particular events. As a result, the individual events characterize particular object manipulations or simple hand movements, while event boundaries identify types of contact onset and offset events. We show that the developing structures are highly suitable (i) to predict the future sensorimotor progression, including when the next event boundary is probably reached and which event can be expected next, and (ii) to execute higher-order, goal-directed planning. We particularly show that the developing hierarchically organized, event-oriented, behaviorally-grounded structures are highly suitable for executing factorized, hierarchical reinforcement learning (RL) according to the options framework (M. M. Botvinick, Niv, & Barto, 2009; Sigaud, Butz, Kozlova, & Meyer, 2009; Sutton, Precup, & Singh, 1999).

Architecture

According to EST, the processing of sensory inputs is influenced by a set of event models, which predict future sensory input. Information about errors in these predictions is used to adapt and switch between the available event models. We implement this approach by using forward models as event models to generate sensory predictions. Additionally, to form representations of events as a set of forward models, our system builds representation of event boundaries, marking the transition from a forward model to another.

The system consists of four main components. It is schematically shown in Figure 1. The **Predictive System** consists of a set of currently active event models, which pre-

dict the next sensory information $\vec{s}_{t+1}$ based on the current sensory input $\vec{s}_t$ and the executed motor action $\vec{x}_t$. The active event models are updated while interacting with the environment using the error $\vec{e}_{t+1}$ between predicted $\vec{s}'_{t+1}$ and actually encountered next sensory perceptions $\vec{s}_{t+1}$. **Error Detection** mechanisms are used to keep track of the accuracy of the predictions. If a significant prediction error is detected, which may be related to notions of *surprise*, event transitions might be detected, leading to a change in the set of active event models. The **Event Models**-component contains all currently active event models and determines which ones are currently applicable. Moreover, the component learns new event models when necessary. Finally, the **Event Boundary Model**-components contain models that characterize event transitions dependent on the currently active event models, the sensory information, and the surprise signal.

These four components essentially constitute the learning and control architecture. The architecture encodes events in the form of sets of temporal forward models. Event boundaries are characterized by event boundary models, which additionally predict the state of the environment at particular event transitions. While interacting with the environment, the system internally simulates the interaction with the environment and verifies these simulations given sensory observations. Moreover, the system can be used to actively infer actions given desired goal states. Due to the event-based architecture, higher-level inference-based planning is possible.

Predictive system

While simulating the current changes in the environment, the predictive system holds N forward models $M(t) = (M_{1,i_1}(t), ..., M_{N,i_N}(t))$ at a certain point in time t. Each forward model predicts sensory changes given motor commands, that is, $M_{n,i} : \vec{x} \rightarrow \Delta s$. At a certain point in time t, each active forward model $M_{n,i_n}(t)$ receives the motor command $\vec{x}_t$ as an input and predicts the sensory consequences $\Delta s'_{n,t+1}$. The predictions of all N active forward models form the predicted sensory change vector $\Delta\vec{s}'_{t+1}$. The predicted sensory input of the next time step is thus:

$$\vec{s}'_{t+1} \leftarrow \vec{s}_t + \Delta\vec{s}'_{t+1}. \quad (1)$$

After executing $\vec{x}_t$ the real sensory input $\vec{s}_{t+1}$ is used to update the active forward models to improve the respective model predictions. Additionally, each forward model $M_{n,i}$ stores the moving average (over the 100 last steps) of its prediction error $\bar{e}_{M_{n,i}}$ and the variance of that prediction error $\sigma^2_{M_{n,i}}$, estimating the current accuracy of the model predictions.

Choosing sufficiently small step sizes, the velocity kinematics of any dynamic system can be approximated arbitrarily well by a linear model. In the general case, the current velocity kinematics depend on the system state. Seeing that in our simple test scenario this was not the case, though, we assumed non-changing, linear velocity kinematics, which we learned by means of Recursive Least Squares.

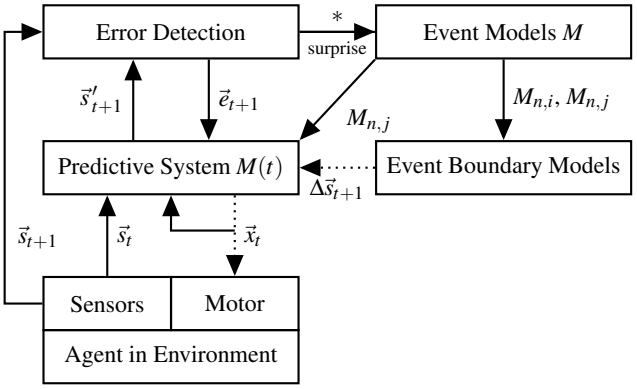


Figure 1: Illustration of the system. Solid arrows symbolize the information flow during forward modeling. In this pathway active forward models predict the next sensory input, are improved by learning and exchanged if necessary. Dashed arrows symbolize the information flow during planning. Here the required sensory change to trigger a desired event is computed and the forward models are used inversely to generate a suitable motor command.

Error detection and switching event models

To this point our architecture uses a set of active forward models to predict the sensory changes during the course of one event. However, if the system is confronted with an event boundary, marking the end of the current event and the beginning of a new one, it should autonomously decide to switch its set of forward models.

According to EST an event boundary is accompanied by a rising error in prediction (Zacks et al., 2007). It has been shown that computational models can use prediction errors to segment a stream of video sequences into separate events (Reynolds, Zacks, & Braver, 2007) or to autonomously identify useful subgoals for higher level motor planning (Butz, Swarup, & Goldberg, 2004). Our architecture uses the error in prediction as a criterion to detect a possible event boundary. The prediction error $e_{n,t}$ for the sensory dimension n at time step t is considered 'surprising' (Butz et al., 2004) if

$$e_{n,t} > \bar{e}_{M_{n,i}} + 2\sigma_{M_{n,i}}, \quad (2)$$

with $M_{n,i}$ being the currently active forward model of dimension n.

If a significant error signal is detected the architecture enters a *searching period*, during which the next active event models are determined. All existing forward models M_n of dimension n and one newly generated forward model are considered. For a fixed number of time steps (10 time steps in our simulations) each model predicts the next sensory input, is updated, and the prediction error is recorded. Afterwards, the mean prediction errors of all models during this searching period are compared and the model $M_{n,j}$ with the smallest mean error is chosen as the new forward model for dimension n in the predictive system. If the winning model $M_{n,j}$

already existed prior to searching, the newly generated model is discarded. If $M_{n,j}$ is new, it is added to the set of possible models M_n for dimension n. All forward model updates on existing models during a searching period are discarded.

Learning event boundary models

The introduced components form a mechanism to detect event boundaries, which can be characterized by an exchange of at least one of the active forward models in the predictive system. For our architecture to be able to act goal-directedly, a representation is necessary that describes at which situation one event boundary occurs. Assuming that an event boundary can be characterized by particular constellations of event-boundary-relevant sensory inputs, we approximate an event boundary by the probability density of sensory constellations that are experienced when a transition occurred. In other words, we are modeling the conditional probability $P(\vec{s} \mid M_{n,i} \to M_{n,j})$, making the assumption that this probability distribution can be reasonably well approximated by a multidimensional normalized Gaussian function $G_{\vec{\mu}_{n,i\to j},\Sigma_{n,i\to j}}(\vec{s}_t)$. This is equivalent to requiring that event boundaries occur close to specific points in sensory space. This assumption holds well in the simple scenario considered. In the general case, other densities may be used such as Gaussian mixture models.

Planning

To be able to trigger desired events, our system can be used in a backwards fashion to plan goal-directed behavior to reach specific event boundaries. To do so, we approximate active inference (Friston et al., 2013) by means of the developing event and event boundary models. As a result, planning consists of two inference stages. First, a target event boundary, or a sequence of event boundaries, is chosen. Next, the necessary motor commands are inferred to reach the next desired event boundary.

Selection of a target event boundary We assume that some of the event boundaries are coupled with positive reward. In our model, event boundaries are characterized by event transitions, such that a particular event transition $M_{n,i} \to M_{n,j}$ is chosen as the goal transition. As a result, the system strives to achieve this transition by attempting to maximize $P(M_{n,i} \to M_{n,j}, \vec{s}_t)$. Higher-level, inference-based planning is used to determine a sequence of event boundaries, which is expected to lead from the current event to the desired event transition.

When $M_{n,i}$ to $M_{n,j}$ is the only desired transition and $M_{n,i}$ is the currently active model, then the system strives to maximize $P(M_{n,i} \to M_{n,j}, \vec{s}_t)$. When multiple event transitions are considered desirable, that is, when transitions from the currently active model $M_{n,i}$ to a set J of potential target models $(M_{n,j})_{j\in J}$ are rewarding, then the transition to the closest mean $\vec{\mu}_{n,i\to j}$ of the associated Gaussian is chosen. When $M_{n,i}$ is currently active, but only transitions $M_{n,k} \to M_{n,j}$ are expected to be rewarding (with $k \neq i$), the system chooses a reachable intermediate transition $M_{n,i} \to M_{n,k}$. Although in our simulation one intermediate transition always suffices, the principle can generally be applied for generating larger sequences of transitions. The approach is also closely related to model-based, hierarchical RL, where extended actions are described as 'options' (Sutton et al., 1999; M. M. Botvinick et al., 2009).

Deriving the motor commands Let us assume that the nth coordinate prediction is based on model i and the transition $M_{n,i} \to M_{n,j}$ is currently desired. The system is supposed to reach a place $\vec{s}'_n$ in sensor space that maximizes the conditional probability of the desired transition:

$$\vec{s}'_n := \arg\max_{\vec{s}} P(M_{n,i} \to M_{n,j} | \vec{s}). \tag{3}$$

This can be reformulated and solved using the Bayes theorem, assuming that also the prior of $\vec{s}_n$ is a Gaussian distribution $P(\vec{s}_n) = G_{\vec{\mu}_{n,i},\Sigma_{n,i}}(\vec{s}_n)$:

$$\vec{s}'_n = (\Sigma_{n,i}\,\vec{\mu}_{n,i\to j} - \Sigma_{n,i\to j}\,\vec{\mu}_{n,i})\,(\Sigma_{n,i} - \Sigma_{n,i\to j})^{-1}. \tag{4}$$

Under the assumption that the prior is approximately uniform (compared to the transition distribution), $\Sigma_{n,i\to j} \ll \Sigma_{n,i}$ this corresponds to the maximum of $P(\vec{s}_t \mid M_{n,i} \to M_{n,j})$. We thus are making small steps in sensory space following the gradient of the transition model:

$$\Delta\vec{s}'_t = \eta\, W\, \nabla P(\vec{s}_t \mid M_{n,i} \to M_{n,j}). \tag{5}$$

If the matrix W is the identity matrix this performs an exact gradient ascend with step width constant η. However, we found better performance when choosing W such that dimensions with large variance are effectively suppressed. The suppression essentially focuses system behavior on the behaviorally relevant input dimensions.

Finally, the desired displacement $\Delta\vec{s}'_t$ in sensory space can be translated directly into a motor command using the inverse of the prediction model:

$$\vec{x}'_t := M^{-1}_{n,i}(\Delta\vec{s}'_t), \tag{6}$$

effectively making the system move towards an area where a desired event boundary is believed to be situated.

Evaluation

Our system was tested in a scenario, where multiple events occur and thus the acquisition of different forward models is necessary to predict the sensory changes. We have therefore chosen a scenario in which a simulated agent interacts with different objects in continuous space. Figure 2 shows the hidden, conceptual structure of the environment, which the model uncovers by the detailed principles.

The agent consists of a hand, able to move freely through a limited workspace, and a stationary mouth area. Three types of differently colored objects (1 type of 'foe' and 2 types of 'food') occur in the simulation. Foe objects have no friction

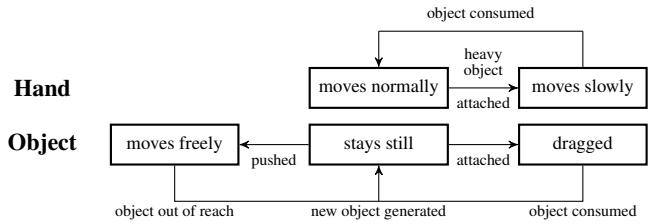


Figure 2: Illustration of the different events (boxes) and event boundaries (arrows), which the introduced system uncovers. The system's 'hand' is able to attach to objects or to push objects, dependent on the object type. A pushed object moves away from the hand until it is out of reach. An attached object moves with the hand until it is consumed. Hand movements are slower when a heavy object is attached.

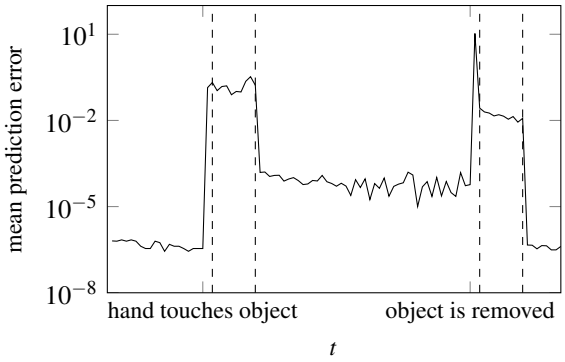


Figure 3: Mean prediction error of all active forward models for one exemplary interaction with a food object. The x-axis displays time with event boundaries highlighted. Dashed lines mark the beginning and end of a searching phase

and slide away without friction when pushed by the hand. They vanish when the distance of the object to the center of the agent's workspace exceeds a threshold. Food objects stick to the hand upon contact and afterwards move along with it. They vanish when they are dragged into the mouth. We use two types of food objects: Light food does not alter the hand movement when attached to it, whereas heavy food slows the hand movement down by a factor of $\frac{1}{2}$. If an object vanishes, a new one is immediately generated at a different position.

In every simulation step t one elementary movement of the hand, described by $\vec{x}_t$, is performed and a sensory input $\vec{s}_t$ is received, which contains all information necessary to predict event boundary occurrences. In particular, $\vec{s}_t$ consists of the position of the hand ($s_{1,t}, s_{2,t} \in [0,100]$), the position of the object ($s_{3,t}, s_{4,t} \in [0,100]$), the position of the object in a hand-centered frame of reference ($s_{5,t}, s_{6,t} \in [-100,100]$), the distance of the object to the center of the workspace ($s_{7,t} \in [0,50]$) and the object's color ($s_{8,t} \in \{0,100,200\}$). $\vec{x}_t$ contains the motor command, which determines the change in hand position (with $\Delta s_{1,t}, \Delta s_{2,t} \in [-0.5,0.5]$). Since the forward models of our architecture must be able to linearly compute the change in sensor information based on $\vec{x}_t$, the vector additionally contains sensory information describing the velocity of the hand during the last object contact (to predict the position of the foe after pushing it) and the velocity of the object in reference to the center of the workspace (to predict changes in the object's distance to the workspace center). A small amount of Gaussian distributed motor noise was added ($\sigma = 0.05$), such that an elementary movement was not completely deterministic.

In our scenario the event boundaries leading to the disappearance of an object and the creation of a new one are rewarded. Therefore the system strives to drag food objects in the agent's mouth and 'kill' foes by pushing them out of the agent's workspace. We chose to reward the event boundaries of the sensory dimension s_3, since the forward models for the object's position need to change at every event boundary and is therefore considered most reliable.

Results

In a first test we evaluated the improvement of the forward models over time by monitoring the prediction error in ten independent simulations. The motor command $\vec{x}_t$ was determined by an informed, hard-coded algorithm, which made the system touch an object with the hand, pushing it away or subsequently dragging it into its 'mouth'. The average prediction error of all active forward models for one exemplary object interaction is plotted in Figure 3. In this example, the hand first moves to the food. After contact, the food sticks to the hand and is moved alongside the hand into the mouth, which results in 'food consumption' and thus food removal. After that, a new object is generated randomly. At the event boundaries (hand touches object, object is removed) the prediction error drastically increases. This ascent is particularly big when the object is removed, since a lot of sensory information changes in this single time step. For the following ten time steps, the system searches for new forward models, such that the prediction error remains large. After that, the best adapted set of forward models is active. The prediction error for all forward models decreases over time. Figure 4 shows the prediction error for some of the forward models over their time of activation. While the prediction errors strongly fluctuate, they all logarithmically converge to 0. All forward models that correctly predict no change in sensory information immediately reach a prediction error of 0 (not shown).

In a second test we evaluated the planning capabilities of our system to use its event and event boundary models to perform goal-directed behavior. The system's goal was to trigger events resulting in the removal of the currently present object. We ran ten simulations, whereas one simulation run consisted of 25 epochs, each consisting of a training and a testing phase. During training, five objects of each type were presented consecutively at random positions. The system was given a time interval of 500 simulation steps to interact with the object. If the system failed to remove the object in the given time period, the hard-coded algorithm used above performed the

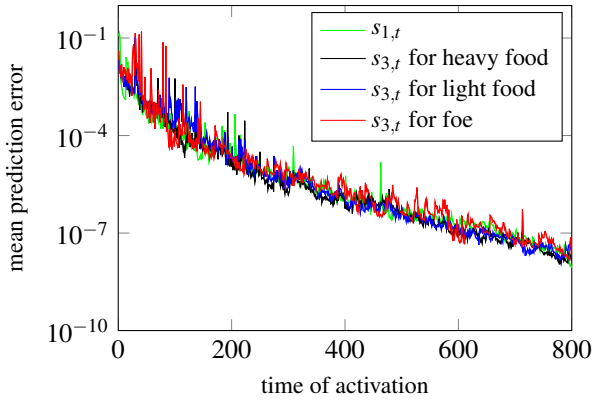
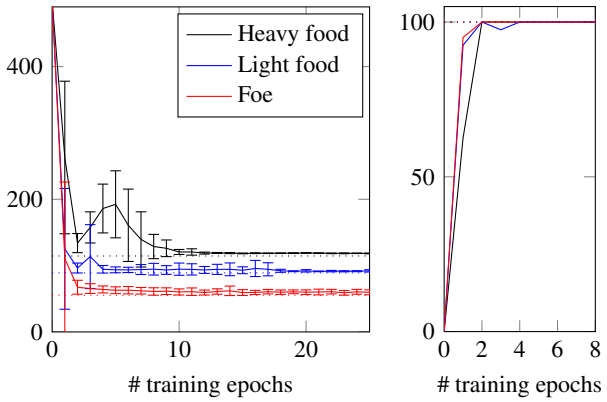


Figure 4: Mean prediction error of one forward model over the time this model is active. Colors indicate the type of the present object and the sensory dimension this forward model predicts. Only the non-trivial cases, in which the object moves, are shown.



(a) Mean number of time steps (b) % successful

Figure 5: Goal-directed behavior during testing epochs; solid lines show system performance; dotted lines show optimal performance; a) mean and standard deviation of time steps required to successfully interact with an object; b) mean percentage of successfully completed object interactions.

required movements. During testing, each object consecutively appeared at four fixed positions. When removing an object during testing the hand was reset to a starting position. Figure 5a shows the mean number of time steps the system needed to remove an object for the different testing epochs. In the first two testing epochs the time required for the interactions drastically decreases. After ten training epochs, the system performs the interactions in nearly minimum time (dashed lines indicate the optimum). Figure 5b shows the percentage of objects removed by the system's hierarchical, goal-directed behavior. Already after the first epoch, the system successfully removes nearly all foe and light food objects. From the fourth epoch onwards, the system removes all objects reliably within the allowed time frame.

To analyze if the system was able to differentiate between relevant and irrelevant sensory dimensions for the prediction of an event boundary, we analyzed the variances of the covariance matrices of each event boundary model after one exemplary run. The mean difference over all Gaussian distributions between biggest and smallest variance in between each Gaussian distribution is 697 – implying that there are drastic differences in relevance for the different sensory dimensions. A more detailed analysis shows that the largest and smallest variance indeed depended on the event boundary. For example, a model describing the event boundary 'hand touches foe' contains the biggest variances for the global position of the object (x-wise 296, y-wise 249) and small variances for the object's color (0.007) and the position of the object in the hand-centered frame of reference (x-wise 19.9, y-wise 21.6). This implies that the object's type and the distance between hand and object are considered relevant for this event boundary, while the exact position of the object is not. In contrast the event boundary 'object is consumed' has the biggest variance for object color (3095) and small variances for the object's position in the hand-centered frame (x-wise 22.9, y-wise 19.3) and for global object position (x-wise 48.7, y-wise 36.5). Here the color of the object is irrelevant because both food objects can be consumed and they differ strongly in color (color difference = 100 in our simulation). Instead, the exact object position is relevant.

Conclusion

Inspired by the event segmentation theory and its schematic model put forward in Zacks et al. (2007), we have developed a computational, motor-grounded event segmentation model. Previous work has shown that statistical analyses of visual changes can be used to categorize segments of video sequences into distinct events (Buchsbaum, Canini, & Griffiths, 2011; Shi, Wang, Cheng, & Smola, 2008; Niebles, Wang, & Fei-Fei, 2008). Additionally and partially in contrast, our model has analyzed spatial, motor-dependent changes by learning predictive forward models and by using the learned forward models to detect event transitions based on a rigorous statistical measure of 'surprise'. Moreover, our system has shown that the learned predictive model cannot only be used to segment sensorimotor time series, but also to plan hierarchically goal-directedly. In the still rather restricted but continuous noisy environmental simulation, our system was able to identify events, which characterized particular object manipulations including 'moving without object contact', 'dragging a light object', 'dragging a heavy object', and 'moving while an object is moving'. Identified event transitions characterized boundary conditions including 'attaching to an object', 'kicking an object', and 'consuming an object'. Event boundary encodings identified those environmental factors that were critical for causing particular event transitions, such that the encodings can be thought of as conceptualizations of environmental interaction options, yielding object concepts, such as 'kickable', 'attachable', or 'draggable'.

Our model is closely related to advances in artificial intelligence and cognitive robotics. Calinon, Guenter, and Billard (2007) have put forward a system that learns a temporal Gaussian Mixture Model from behavioral demonstrations. Imitations of observed environmental interactions were executed using Gaussian mixture regression, focusing control on the relevant interaction aspects. Segmentation and higher level planning, however, were not addressed. Other work has predefined partitions over continuous subspaces during which a particular motor skill could be activated (Konidaris, Kaelbling, & Lozano-Perez, 2014). Partitions were computed by a global clustering algorithm. In contrast, our system learns to partition its environment by means of local measures of surprise based on developing forward models. Moreover, our system factorizes its developing model such that is becomes able to identify those environmental properties that are critical to bring a particular event about.

In sum, the proposed computational model offers an algorithm that can develop suitable event segmentations online from sensorimotor experiences with the environment. The model suggests that EST may be applied to structure own motor behavior and to identify those sensorimotor signals that are critical to accomplish particular environmental manipulations. We are currently working on extending the framework to be able to also solve non-linear control challenges in more complex scenarios in virtual realities.

References

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences*, *22*, 577–600.

Barsalou, L. W., Breazeal, C., & Smith, L. B. (2007). Cognition as coordinated non-cognition. *Cognitive Processing*, *8*, 79-91.

Botvinick, M., & Weinstein, A. (2014). Model-based hierarchical reinforcement learning and human action control. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, *369*(1655). doi: 10.1098/rstb.2013.0480

Botvinick, M. M., Niv, Y., & Barto, A. C. (2009). Hierarchically organized behavior and its neural foundations: A reinforcement learning perspective. *Cognition*, *113*, 262 - 280.

Buchsbaum, D., Canini, K. R., & Griffiths, T. L. (2011). Segmenting and recognizing human action using low-level video features. In *Annual conference of the cognitive science society* (p. 3162-3167).

Butz, M. V., Sigaud, O., & Gérard, P. (2003). Internal models and anticipations in adaptive learning systems. In M. V. Butz, O. Sigaud, & P. Gérard (Eds.), *Anticipatory behavior in adaptive learning systems: Foundations, theories, and systems* (pp. 86–109). Berlin: Springer.

Butz, M. V., Swarup, S., & Goldberg, D. E. (2004). Effective online detection of task-independent landmarks. In R. S. Sutton & S. Singh (Eds.), *ICML'04 workshop on predictive representations of world knowledge.*

Calinon, S., Guenter, F., & Billard, A. (2007). On learning, representing, and generalizing a task in a humanoid robot. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, *37*, 286–298.

Clark, A. (2013). Whatever next? predictive brains, situated agents, and the future of cognitive science. *Behavioral and Brain Science*, *36*, 181-253.

Engel, A. K., Maye, A., Kurthen, M., & König, P. (2013). Where's the action? the pragmatic turn in cognitive science. *Trends in Cognitive Sciences*, *17*, 202 - 209.

Friston, K. (2009). The free-energy principle: a rough guide to the brain? *Trends in Cognitive Sciences*, *13*, 293 - 301.

Friston, K., Schwartenbeck, P., Fitzgerald, T., Moutoussis, M., Behrens, T., & Dolan, R. J. (2013). The anatomy of choice: active inference and agency. *Frontiers in Human Neuroscience*, *7*(598). doi: 10.3389/fnhum.2013.00598

Hohwy, J. (2013). *The predictive mind*. Oxford, UK: Oxford University Press.

Konidaris, G., Kaelbling, L., & Lozano-Perez, T. (2014). Constructing symbolic representations for high-level planning. *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 1273-1280.

Niebles, J. C., Wang, H., & Fei-Fei, L. (2008). Unsupervised learning of human action categories using spatial-temporal words. *International journal of computer vision*, *79*, 299–318.

Reynolds, J. R., Zacks, J. M., & Braver, T. S. (2007). A computational model of event segmentation from perceptual prediction. *Cognitive Science*, *31*, 613–643.

Shi, Q., Wang, L., Cheng, L., & Smola, A. (2008). Discriminative human action segmentation and recognition using semi-markov model. In *Computer vision and pattern recognition. cvpr 2008. ieee conference on* (pp. 1–8).

Sigaud, O., Butz, M., Pezzulo, G., & Herbort, O. (2013). The anticipatory construction of reality as a central concern for psychology and robotics. *New Ideas in Psychology*, *31*, 217 - 220.

Sigaud, O., Butz, M. V., Kozlova, O., & Meyer, C. (2009). Anticipatory learning classifier systems and factored reinforcement learning. In G. Pezzulo, M. V. Butz, O. Sigaud, & G. Baldassarre (Eds.), *Anticipatory behavior in adaptive learning systems* (p. 321-333). Berlin: Springer.

Sutton, R. S., Precup, D., & Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, *112*, 181-211.

Zacks, J. M., Kumar, S., Abrams, R. A., & Mehta, R. (2009). Using movement and intentions to understand human activity. *Cognition*, *112*, 201–216.

Zacks, J. M., Speer, N. K., Swallow, K. M., Braver, T. S., & Reynolds, J. R. (2007). Event perception: A mind-brain perspective. *Psychological Bulletin*, *133*, 273–293.

Zacks, J. M., & Tversky, B. (2001). Event structure in perception and conception. *Psychological Bulletin*, *127*, 3–21.