BehAVE: Behaviour Alignment of Video Game Encodings

Nemanja Rašajski[1⋆], Chintan Trivedi[1⋆], Konstantinos Makantasis[2], Antonios Liapis[1], and Georgios N. Yannakakis[1]

[1] Institute of Digital Games, University of Malta, Malta
{nemanja.rasajski,ctriv01,antonios.liapis,georgios.yannakakis}@um.edu.mt
[2] AI Department, University of Malta, Malta
konstantinos.makantasis@um.edu.mt

Abstract. Domain randomisation enhances the transferability of vision models across visually distinct domains with similar content. However, current methods heavily depend on intricate simulation engines, hampering feasibility and scalability. This paper introduces BehAVE[3], a video understanding framework that utilises existing commercial video games for domain randomisation without accessing their simulation engines. BehAVE taps into the visual diversity of video games for randomisation and uses textual descriptions of player actions to *align* videos with similar content. We evaluate BehAVE across 25 first-person shooter (FPS) games using various video and text foundation models, demonstrating its robustness in domain randomisation. BehAVE effectively aligns player behavioural patterns and achieves zero-shot transfer to multiple unseen FPS games when trained on just one game. In a more challenging scenario, BehAVE enhances the zero-shot transferability of foundation models to unseen FPS games, even when trained on a game of a different genre, with improvements of up to 22%. BehAVE is available online[4].

1 Introduction

Video game engines uphold an internal representation of the game environment [12, 17], encompassing essential variables such as player position and map layout. Upon undergoing processing by the game graphics *renderer*, this data becomes intricately entwined with the game's visual style, resulting in the images presented to the player on screen. Securing access to game engine data, however, proves challenging, if not impossible in practice, particularly for commercial video games. Consequently, this circumstance directs the trajectory of game artificial intelligence (AI) research towards the utilisation of more accessible game representations such as pixels [14, 39]. Unfortunately even with state of the art pre-trained computer vision (CV) models the resulting game pixel encodings do not generalise well, even between games of the same genre, and suffer from what is known as the *domain gap* problem [35] (see Fig. 1a).

⋆ Equal contribution
[3] https://sites.google.com/view/behavefw/home
[4] https://github.com/nrasajski/BehAVE

(a) Foundation Encodings **(b)** BehAVE Encodings

Fig. 1: High level overview of the **BehAVE** framework. The t-SNE plots show encodings of short video sequences from 5 distinct FPS games: (a) indicates the *domain gap* between encodings of different games from a video foundation model, while (b) shows encodings aligned by BehAVE. The framework positions similar player behaviour encodings (*e.g.*, aim gun) closely across visually diverse games like *PUBG* (left) and *Apex Legends* (right).

Improving model generalisation stands as a pivotal problem within machine learning (ML) research, encompassing various research topics such as domain adaptation, meta-learning, and transfer learning. Domain generalisation [33] has emerged as a focal point of interest aiming to develop models capable of effectively generalising across unseen testing domains by leveraging training data from diverse but related domains. One highly promising technique for enhancing the transferability of CV models in games is **domain randomisation** [28], a simple technique that improves the robustness of a CV model by training it on visuals derived from randomising rendering parameters of a simulator engine. Building accurate large-scale simulators, however, is a formidable challenge which requires significant time, expertise, and effort [6]. Identifying and randomising relevant simulation parameters adds further to the complexity, making the entire process a resource-intensive endeavour. Additionally the randomisation process can occasionally lead to inaccurate and infeasible results.

To address the above mentioned issues, in this paper we introduce a novel approach to domain randomisation that leverages the rich visual diversity inherent in **video games**; see Fig. 1. Specifically our framework distinguishes itself by not relying on game engine access for the randomisation process. This unlocks the potential of CV to be trained and tested on videos from commercial-grade video game titles, a capability that has, until now, remained elusive. Our Behaviour Alignment of Video Game Encodings (**BehAVE**) framework harnesses the unique characteristic of gameplay videos opposed to any other videos available. Gameplay video footage is generated by sequential player actions (*i.e.,* controller inputs) that control on-screen animations characterised as *player behaviour*. Using player actions, BehAVE is able to align video encodings of similar player behaviours across visually diverse games (*i.e.*, different *domains*). Crucially, it employs *semantic action encoding*, a method that infuses semantic information about behaviour through textual descriptions of actions, which are encoded using pre-trained text encoders [15]. As a result, the text encodings of player actions guide the behavioural alignment of video encodings.

We train BehAVE's alignment module, implemented on top of foundation video encoders [30], across a diverse array of games from the first person shooter (FPS) game genre, namely our introduced *SMG-25* dataset. Our findings suggest that BehAVE is able to uncover similar behavioural patterns—despite visual distinctions, such as variations in game style or aesthetics—across unseen games of the SMG-25 dataset (see Figure 1 for an illustrative example). The alignment training proves efficient even with small datasets comprising only a few games, and it demonstrates robustness across various tested video and text encoders.

To assess the *transferability* of BehAVE we test it on a video understanding task across various FPS games, while solely being trained on the FPS game *Counter Strike: GO* (Valve, 2012). Further, we test a more challenging scenario evaluating the transfer performance to the FPS genre from *Minecraft* (Mojang, 2011) a first person game from a different genre (non-FPS). Our findings indicate higher transferability when learning to classify behaviour from our aligned representation space as compared to without, showcasing up to 22% higher classification accuracies across the different behaviour categories tested. We view this as a potential avenue for annotating extensive datasets of online gameplay videos with behaviour labels, thereby serving as a stepping stone towards learning generalised representations of behaviour in videos. Our contributions can be summarised as follows: (1) We introduce the **BehAVE** framework for domain randomisation via commercial video games; (2) We propose **Semantic Action Encoding** for representing player actions as textual descriptions processed through a pretrained text encoder; (3) We introduce the **SMG-25** dataset of synchronised gameplay and actions.

2 Background

Video Understanding in CV. Video understanding methods seek to interpret visual information embedded within temporal image-sequences. Recent strides in deep learning have led to attaining remarkable performance in diverse video understanding tasks, including but not limited to *video classification* [4], *video summarisation* [1], *short and long-form video understanding* [38], and *object tracking* [44]. Current endeavours focus on training strategies that are independent of any specific downstream task. The resulting *video foundation models* [34, 36] yield powerful video representations, readily applicable across a diverse range of tasks. We use such foundation models in our study courtesy of their out-of-the-box performance and employ them as is (*i.e.* frozen) bounded by limited computational resources [29]. This underscores the computational efficiency of our video understanding framework, ultimately enhancing its accessibility.

Transferable CV and Domain Randomisation. Despite their impressive out-of-the-box performance, foundation models showcase limited capacities on transferring knowledge from one domain to another *visually distinct* domain, primarily due to the "domain gap" challenge [31, 32]. Tobin et al. [28] introduced the technique of *domain randomisation* to train transferable vision models by injecting variability during learning. This is achieved by randomising the ren-

dering of a simulator that generates training data. Leiprecht [11] showcases the efficacy of domain randomisation in CARLA [6], a large-scale driving simulator. Mishra et al. [13], however, bring to light the numerous complexities associated with identifying and tweaking relevant parameters of such simulators. Furthermore, Kim et al., [8] emphasise the *limited* variability that can be attained from a single simulator, impacting the transfer capacity [42]. Hence, in this work, we adopt a simulator-free approach for visual domain randomisation.

Video Games for CV. Inspired by insights from [23] and [20] suggesting that procedurally generated sets of diverse games enhance generality in machine learning, we explore the use of existing video games in CV. Several recent studies investigate the use of commercial-standard games as an alternative to dedicated in-lab simulators or procedural game level generation approaches, in an attempt to circumvent limitations related to inaccessible game engines. Notably, *Grand Theft Auto 5* (Rockstar, 2013) serves as a popular video game for collecting annotated data, achieved by intercepting rendering communication between graphics hardware and the screen buffer [10,22,27] or employing a game modification such as "infrared vision mod" [7]. Alternatively, Pearce and Zhu [16] gather internal game state information from *CS:GO* by probing the machine's memory. In contrast to such prior works involving the "reverse-engineering" of game engines, our approach captures high-level game information such as player actions using raw inputs from the machine's I/O devices, thereby simplifying the collection of annotated gameplay and boosting the scalability of our method across numerous commercial video games.

Multimodal Alignment of CV Models. Given that BehAVE considers different modalities of input such as videos of gameplay and corresponding player actions, we draw inspiration from contemporary work in *video action recognition* [43]. In their work with a paired video-text caption dataset, Song et al. [26] extract *verbs* from captions and use them as *action labels*. Our framework builds on similar principles utilising language models [15], but instead, encodes player actions; BehAVE then uses these action encodings for alignment with another modality, namely gameplay videos. To achieve this, we rely on *multimodal alignment frameworks* that operate with and align vision and language such as CLIP [18] and VideoCLIP [9,40]. Drawing upon insights from the aforementioned studies, we propose a novel method for performing visual domain randomisation with commercial games by aligning gameplay videos with semantically represented player actions.

3 The BehAVE Framework

As introduced earlier, we present *BehAVE*, a framework operating on paired visuals-and-actions datasets derived from commercial games, with the aim of aligning video encodings based on similar player behaviour. The BehAVE method is presented in Algorithm 1 and visually depicted in Figure 2a. In Section 3.1, we explain the special structured dataset of games imperative for our framework,

Algorithm 1 Behaviour-Alignment Training with BehAVE

Inputs: Games Dataset $\mathbb{D}$, semantic action mapper m, pre-trained text encoder h, pre-trained video encoder f and trainable alignment projector p.
for (video V, actions A) in $\mathbb{D}$ **do**
  Compute video encoding $z^{\text{video}} = f(V)$
  Compute action encoding $z^{\text{caption}} = h(m(A))$
  Project to aligned encoding $z^{\text{align}} = p(z^{\text{video}})$
  Calculate loss $\mathcal{L}_{\cos} = 1 - cosine(z^{\text{align}}, z^{\text{caption}})$
  Update projector network parameters p_θ
end for
Output: Trained alignment projector p.

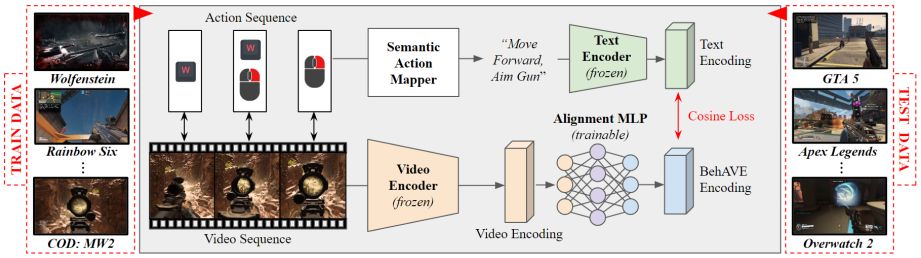
followed by Section 3.2 covering the encoder models used for both modalities. Finally, Section 3.3 details the training method employed.

3.1 Games Dataset for Training BehAVE

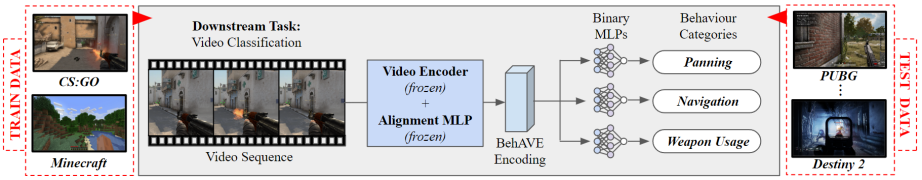
A crucial component of our domain randomisation framework involves the meticulous preparation of a dataset adhering to a specific *structure* that accommodates semantically similar visual content represented across diverse visual styles. We enforce this structure via the selection criteria of the various commercial games in the training dataset. Note that since BehAVE is trained upon player-game interaction data, we do not require access to the game engines, making it a viable strategy to use commercial games.

Game Selection for Domain Randomisation. Let $\mathcal{G}$ represent a game, with a frame-renderer g, and $\mathcal{G} \in \mathcal{C}$, where $\mathcal{C}$ denotes the family of games of a certain game genre category. Given that domain randomisation with customisable simulators involves the adjustment of simulator render parameters, we proceed to identify and formally define comparable parameters ξ within the context of video games. Each game's renderer encompasses certain game-specific parameters denoted as "game style parameters" ($\xi^{\mathcal{G}}$), associated with either the visual aesthetics of the game, such as *textures and colours of objects*, or the underlying rules governing the game, such as *game physics*. These parameters are considered invariant throughout the game, reflecting game design choices made during development, and are less likely to be shared across all games of this genre category. In the context of our analysis involving multiple commercial games, we observe $\xi^{\mathcal{G}} \in \Xi$ where Ξ represents the diverse *global game-design space*, introducing implicit "randomisation" into our framework. This unique characteristic of games makes them ideal for the purpose of visual domain randomisation.

Additionally, we also characterise all game-state-specific parameters of the renderer, including the *player's spatial coordinates*, *health or ammunition status*, and *camera perspective*, as "game content parameters" ($\xi_t^{\mathcal{C}}$), which dynamically evolve at each timestep t in response to player interactions with the game environment. Note that these parameters remain largely consistent across different games that are categorised under the same genre.



(a) Behaviour Alignment



(b) Behaviour Classification

Fig. 2: Overview of experiments and datasets used: **(a) Behaviour Alignment:** BehAVE is trained on synchronised gameplay video and player actions from the SMG-25 train dataset, and evaluated on unseen games from the SMG-25 test dataset. **(b) Behavior Classification:** We test the transferability of a *video classification* task. BehAVE is trained independently on CS:GO and Minecraft, and transferred to the SMG-25 test dataset.

Synchronised Gameplay Recording. At each timestep t, we obtain two synchronised information streams—visuals and actions. Player inputs or actions are selected from the shared action space of the game genre $\mathcal{C}$ and are represented by a set of $N^{\mathcal{C}}$ unique keypresses as $A_t = \{a_n\}_{n=1}^{N^{\mathcal{C}}}$, where $a_n \in \{0, 1\}$. Visuals are recorded in the form of RGB frames $F_t \subset \mathbb{R}^{h \times w \times 3}$ where h is height and w is width. The visuals of a game can be regarded as dynamic sequences of frames that arise from the interactions between the player and the game, as follows: $F_{t+1} = g(F_t, A_t, \xi_t^{\mathcal{C}} \mid \xi^{\mathcal{G}})$. Thus, the video frames are generated sequentially by the game renderer g processing the game content and player action information at every timestep for the given predefined game style. This inherent characteristic of gameplay visuals, derived from player interactions, enables us to employ actions for effectively discerning visual content.

Data Pre-processing. Although we collect data at the timestep level, our framework operates on videos for identifying behaviour. To this end, we aggregate data over consecutive timesteps, forming a window of length T to obtain video sequences $V = (F_1, F_2, ..., F_T)$ and action sequences $A = (A_1, A_2, ..., A_T)$. Consequently, for each game, the synchronised gameplay-actions dataset is denoted as $D^{\mathcal{G}} = \{(V_i, A_i)\}_{i=1}^{I}$ where $|D^{\mathcal{G}}| = I$. We construct the *overall* dataset $\mathbb{D} = \bigcup_{\mathcal{G} \in \mathcal{C}} D^{\mathcal{G}}$ comprising of $k = |\mathbb{D}|$ "distinct" games from the game genre $\mathcal{C}$, adhering to the previously outlined selection strategy. This dataset forms the basis for behaviour alignment training.

3.2 Encoding the Modalities

Video Encoding and Alignment. As previously stated, we harness the capabilities of a pre-trained video foundation model in our study for video understanding. Let f denote the backbone model of a *video encoder*; thus, the latent representation of the backbone's video encoding can be given by $z^{\text{video}} = f(V)$ (Algorithm 1, line 3), where V denotes a 1 second video consisting of 16 consecutive frames. Note that we employ f within our training in a frozen state, a decision influenced by our evaluation of foundational models as well as other computational constraints. To facilitate the alignment of representation spaces of different modalities within our framework, we employ a trainable MLP model p that acts as an *alignment projector* and operates on top of the video encoder, yielding the aligned projection encoding $z^{\text{align}} = p(z^{\text{video}})$ (Algorithm 1, line 5).

Semantic Action Encoding. As previously explained, each video is associated with a sequence of binary actions A, indicating the presence or absence of a key-press at every timestep. Binary labels for actions, however, offer limited insights into the inter-relationships among various sub-actions. For instance, in FPS games, the binary encodings of the four actions—*move left*, *move right*, *shoot gun*, and *aim gun*—are equidistant from one another. This encoding type, however, fails to capture the underlying semantic similarity between the first two actions (*i.e.*, related to movement), the last two actions (*i.e.*, related to weapon use) and also the semantic difference between these two behavioural categories.

To address the above limitation, we propose to equip BehAVE with a handcrafted *semantic action mapper* function m which injects semantic information into the action encodings via text. It maps the binary sequence of keypresses to a behaviour text caption. Then, we use a pre-trained text foundation model in the form of a *text encoder* h that gives the caption's text encoding $z^{\text{caption}} = h(m(A))$ (Algorithm 1, line 4). We argue that such pre-trained encoders will be able to better capture the inter-relationships among the joint distribution of actions that are otherwise difficult to represent with binary action encodings.

3.3 Alignment Training

Upon obtaining the encodings of videos and actions, we initiate the training phase of the framework using the specified dataset of games. We observe different video sequences exhibiting similar behaviour across different games. Consequently, to align the representation space of the video encoder to match that of the text encoding of behaviour, we choose to train our projector head as attached to the video encoder. Subsequently, we use a loss $\mathcal{L}_{\cos}$ (Algorithm 1, line 6) based on the cosine similarity between the video projector encoding and the behaviour text encoding so that the former aligns with the latter on the same (shared) representation space. The loss is defined as follows:

$$\mathcal{L}_{\cos}(z^{\text{align}}, z^{\text{caption}}) = 1 - \frac{z^{\text{align}} \cdot z^{\text{caption}}}{\|z^{\text{align}}\|_2 \, \|z^{\text{caption}}\|_2} \tag{1}$$

Upon completion of the alignment training, the video encoder equipped with the trained alignment projector can be utilised on any other visual content for video understanding, without requiring access to any other modalities—such as player actions—that only pertain to games.

In summary, the introduced BehAVE framework operates as follows. The structurally enriched dataset of the framework facilitates domain randomisation, the pre-trained video encoder enables video understanding, the semantic action encoding introduces the semantic notion of behaviour, and the alignment training module ensures enhanced transferability in video understanding.

4 Experiments

Figure 2 outlines the two primary experiments conducted in our study: 2a *Behaviour Alignment*, where we perform alignment training (Section 4.2), and 2b *Behaviour Classification*, where we assess transferability of the aligned models in a downstream classification task (Section 4.3). Before delving into the experiments, in Section 4.1 we introduce the datasets and evaluation metrics employed. Note that all experiments have been carried out on a single GTX 1070 (8GB) GPU, highlighting the cost-effective nature of our method.

4.1 Dataset and Metrics

We test BehAVE on three datasets, namely *SMG-25*, *CS:GO*, and *Minecraft*, across the two experiments of our study. The **SMG-25** (Synchronised Multi-Game FPS Dataset) is our newly introduced dataset illustrated in Figure 3 that encompasses synchronised gameplay visuals and player action data from multiple commercial First Person Shooter (FPS) games, gathered following the structure outlined in Section 3.1. It comprises over ∼250K data points spanning 25 visually diverse FPS games, encompassing actions related to player behaviour categories such as *panning* (player looking around), *navigation* (player moving in the environment) and *weapon usage* (player engaging the equipped weapon). We partition it into an SMG-25 *train set* for use in Experiment (a) and an SMG-25 *test set* used for evaluations in both Experiments (a) and (b). The train-test splits consist of disjoint sets of games, enabling evaluation of zero-shot performance on unseen games. Further details about this dataset are available in supplementary material. Additionally, we source similar gameplay-actions data from other commercial games, namely **CS:GO** (180K data points of *cs-dust* level from [16]) and **Minecraft** (50K data points of *contractor demonstrations* from [3]), which serve as training datasets for Experiment (b).

Evaluation Metrics. To comprehensively assess the alignment quality in Experiment (a) and the transferability of aligned models in Experiment (b), we employ several metrics. The *Silhouette Score* [24], ranging from -1 to 1, quantifies the cluster quality of embeddings. In Experiment (a), it is used to gauge the effectiveness of the alignment projection based on behaviour categories as cluster labels, with a higher score indicating better-defined clusters. Additionally,

Fig. 3: Screenshots from all games of the SMG-25 dataset: **1)** PUBG: Battlegrounds (*PUBG Studios, 2017*); **2)** Payday 3 (*Starbreeze Studios, 2023*); **3)** Insurgency: Sandstorm (*New World Interactive, 2021*); **4)** Call of Duty: MW2 (*Infinity Ward, 2022*); **5)** Far Cry 5 (*Ubisoft, 2018*); **6)** Bioshock Infinite (*Irrational Games, 2013*); **7)** Grand Theft Auto 5 (*Rockstar, 2013*); **8)** Rainbow Six: Siege (*Ubisoft, 2015*); **9)** Team Fortress 2 (*Valve, 2007*); **10)** Wolfenstein (*Machine Games, 2014*); **11)** Apex Legends (*Respawn Entertainment, 2019*); **12)** Atomic Heart (*Mundfish, 2023*); **13)** Warhammer: Vermintide 2 (*Fatshark, 2018*); **14)** Back 4 Blood (*Turtle Rock Studios, 2021*); **15)** Halo 4 (*343 Industries, 2012*); **16)** Crysis 2 (*Crytek, 2011*); **17)** Overwatch 2 (*Blizzard Entertainment, 2022*); **18)** Deathloop (*Arkane Lyon, 2021*); **19)** Valorant (*Riot Games, 2020*); **20)** Generation Zero (*Systemic Reaction, 2019*); **21)** Polygon (*Readaster Studio, 2020*); **22)** Titanfall 2 (*Respawn Entertainment, 2016*); **23)** Destiny 2 (*Bungie, 2017*); **24)** Shatterline (*Frag Lab, 2022*); **25)** Operation Harsh Doorstep (*Drakeling Labs, 2023*).

the *Transferability Score* evaluates results in Experiment (b) by considering the percentage difference in classification test accuracy between models trained on BehAVE encodings and those trained on foundation video encodings. A positive difference signifies better transferability for the alignment method relative to the corresponding foundation method, and vice versa.

4.2 Behaviour Alignment

In this experiment, depicted in Figure 2b, we test the BehAVE framework on the SMG-25 dataset. Table 1 provides details on various pre-trained video and text encoders analysed, while a comprehensive report on the configurations tested is available in supplementary material. For the alignment projector, we opt for a 4-layer MLP with ReLU activations and 50% dropout. The size of the final layer of the MLP is adjusted to match the encoding size of the selected text encoder.

As elaborated next, we perform a thorough analysis of various design choices incorporated into our framework.

Table 1: The video and text foundation models used in our experiments. (†) For video and text models, respectively, input size indicates timesteps of RGB frames and maximum token length. The number of parameters and the encoding size of the models are also listed.

Input	PreTrain Method	Model	Input Size†	#Params	Encoding Size
Video	I3D [5]	3D-ConvNet	$16 \times 3 \times 224 \times 224$	79M	512
	VideoMAEv2 [34]	ViT-Base	$16 \times 3 \times 224 \times 224$	87M	768
	MVD [36]				
Text	GPT-2 [19]	Transformer	512 tokens	110M	768
	CLIP [18]		77 tokens	63M	512
	BERT [21]		256 tokens	33M	384

Action Encoding Schemes. We conduct a comparative study on our semantic action encodings, focusing on actions alone, without videos, in our dataset. We compare the cluster quality of actions encoded traditionally in binary form (*i.e., keypress labels*) to text encodings (*i.e., behaviour captions*), with the aim of highlighting benefits of infusing semantics into our BehAVE framework. Results using only the unique set of actions from SMG-25 are presented in Section 5.1.

Impact of Alignment Training. Subsequently, we train BehAVE with the previously mentioned pre-trained video and text encoders and analyse the benefits of aligning the representation space of the video encoder with that of the text encoder. The alignment projector is trained for 10 epochs using the *adam* optimiser with a learning rate of $1e^{-3}$. The training data comprises a set of 15 games from SMG-25, processed in batch sizes of 128. The resulting representation space post-alignment is evaluated using the *silhouette score* metric on 10 games from the SMG-25 test set, allowing us to assess the "zero-shot" performance of our methods on unseen games. In Section 5.1 we present results from 5 independent runs, with randomised train-test splits for each run.

Sensitivity Analysis of *k*. For practical applications it is important to analyse how the number of games (k) used during training affects alignment performance. Thus, we explore the impact of varying the number of games included in the training set, ranging from 1 to 15 and compare them on a fixed test set of 10 games. We randomise the selection of games in the train and test sets across the 5 runs reported in Section 5.1.

4.3 Behaviour Classification

Following alignment training, we emphasise the benefits of BehAVE encodings for transferring a downstream task across visually distinct domains. For this purpose, we select video classification as a representative task in video understanding, with behaviour categories serving as the class labels. As depicted in Figure

Table 2: Impact of Alignment Training. Average silhouette scores (with standard deviations) over 5 runs for behaviour and game labels on the SMG-25 test set. Higher scores indicate better performance for behaviour categories, while lower scores indicate better performance for game labels. Best performing models are highlighted in bold.

Alignment (Approach)	Encoder Methods (Video - Action)	Alignment Dimension	Behaviour Labels Panning ↑	Navigation ↑	Weapon ↑	Game Label ↓
Foundation (Baseline)	I3D - None	512	$0.03_{\pm 0.04}$	$0.04_{\pm 0.04}$	$0.00_{\pm 0.01}$	$0.07_{\pm 0.07}$
	VideoMAEv2 - None	768	$0.08_{\pm 0.00}$	$0.13_{\pm 0.00}$	$0.03_{\pm 0.00}$	$0.10_{\pm 0.00}$
	MVD - None	768	$0.14_{\pm 0.02}$	$0.09_{\pm 0.04}$	$0.01_{\pm 0.01}$	$-0.05_{\pm 0.02}$
Keypress (Naive)	I3D - Binary	16	$0.36_{\pm 0.01}$	$0.45_{\pm 0.01}$	$0.30_{\pm 0.02}$	$-0.21_{\pm 0.01}$
	VideoMAEv2 - Binary	16	$0.35_{\pm 0.00}$	$0.48_{\pm 0.01}$	$0.32_{\pm 0.01}$	$-0.16_{\pm 0.00}$
	MVD - Binary	16	$0.43_{\pm 0.02}$	$0.44_{\pm 0.03}$	$0.09_{\pm 0.01}$	$-0.24_{\pm 0.02}$
BehAVE (Ours)	VideoMAEv2 - GPT-2	768	$\mathbf{0.51_{\pm 0.02}}$	$\mathbf{0.58_{\pm 0.05}}$	$0.20_{\pm 0.05}$	$\mathbf{-0.29_{\pm 0.06}}$
	VideoMAEv2 - CLIP	512	$0.40_{\pm 0.01}$	$0.49_{\pm 0.01}$	$\mathbf{0.35_{\pm 0.00}}$	$-0.20_{\pm 0.01}$
	VideoMAEv2 - BERT	384	$0.41_{\pm 0.03}$	$0.48_{\pm 0.04}$	$\mathbf{0.35_{\pm 0.02}}$	$-0.17_{\pm 0.03}$

2b, for each behaviour category, we train a classifier (3-layer MLP with binary output) on video encodings of a single game (source domain) not included in our SMG-25 dataset, and evaluate the performance of this classifier on multiple FPS games from the SMG-25 test set (target domains). We first use foundation video encodings as input to this classifier, and then BehAVE encodings are employed in the same fashion. We ultimately compare the classifiers' performance on FPS games between the two methods using the *transferability score* metric. We first perform this experiment with our source domain being CS:GO, a game of the *same* genre, followed by a more challenging experiment with our source domain being Minecraft, a *similar* first-person perspective game but from non-shooter genre. In both cases, we evaluate classification performance on the SMG-25 test set of games unseen in alignment training to gauge the zero-shot transfer capacity of our classifiers; results are reported in Section 5.2.

5 Results

5.1 Behaviour Alignment

Comparing Action Encoding Schemes. Figure 4a presents the comparison between binary action encoding and semantic text encoding from the pre-trained CLIP text encoder [18]. The silhouette score for binary actions (0.11) is significantly lower than that for text encodings (0.41), showing that text encodings produce better cluster quality. Findings are further supported by t-SNE projections, where actions in the same behaviour categories appear to form more compact and distinct clusters. These results highlight the advantage of encoding actions as text based on semantics, rather than just as binary keypresses.

Impact of Alignment Training. Table 2 presents the analysis of the representation spaces before and after running BehAVE's alignment training. First, we observe that for all three behaviour categories, even alignment based on a naive

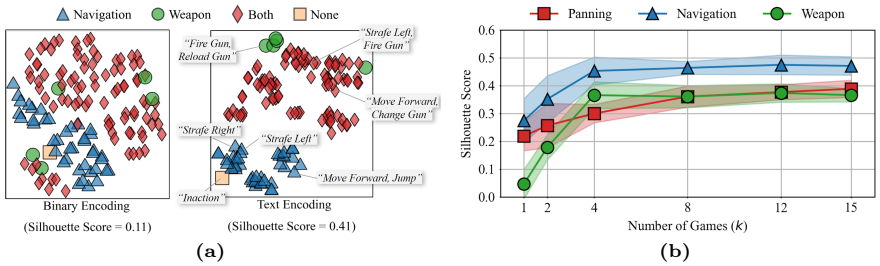


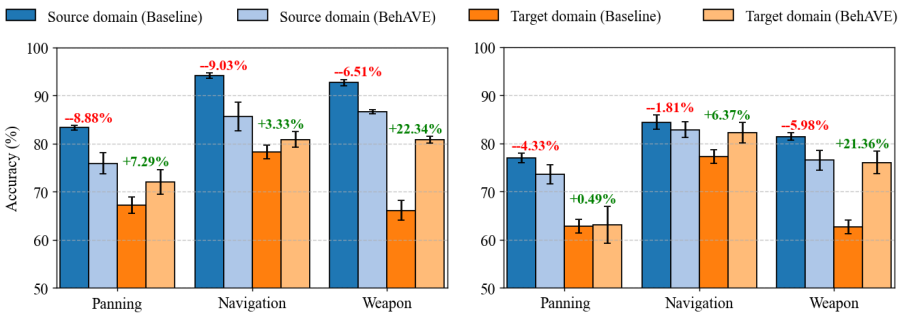
Fig. 4: Behaviour alignment experiments: (a) t-SNE embeddings and corresponding silhouette scores of actions encoded as binary labels (left) compared to pretrained text encoders (right). (b) Effect of varying the number of games in alignment training on behaviour category clustering across 10 test games.

approach of binary encoding of actions (see middle block of table) improves clustering quality across all video foundation models tested (upper block). Second, more interestingly, we observe even bigger improvements when using BehAVE's text encoding scheme (lower block). As a consequence of this alignment, we also observe that the domain gap, indicated by clustering on game labels, reduces for all alignment approaches compared to the foundation approach. The benefits of BehAVE are apparent across all tested configurations; a comprehensive list of all video and text encoders are provided in supplementary material.

Sensitivity Analysis of *k*. Figure 4b illustrates the sensitivity of the domain randomisation process to the number of games (k) used in training. Performance varies across different behaviour categories. For *weapon usage* and *navigation*, we observe convergence between 4 to 6 games, indicating that a relatively small number of games is sufficient for identifying these categories. Surprisingly, for *panning*, alignment continues to improve beyond 10 games. We argue that this phenomenon is attributed to the substantial variability in panning actions across different games, particularly due to different mouse sensitivity presets in SMG-25 that were approximated through visual inspection rather than precise extraction from the game engine (additional details in supplementary material). As a result, in our experiments with SMG-25, we chose 15 games for training and 10 for testing to balance the effectiveness of domain randomisation while maintaining a sufficiently large test set for zero-shot performance evaluation.

5.2 Behaviour Classification

Transfer from CS:GO (*Same* Genre). Figure 5a shows transferability scores across all three behaviour categories, with CS:GO as the source domain and multiple unseen FPS games from the SMG-25 test set as target domains. Notably, we notice the poor transfer capacity of classifiers based on foundation encodings. In the case of BehAVE, while the absolute performance of the classifiers on source domains declines (highlighted in red above the respective bars), we observe a 3% to 22% improvement in transfer to target domains (highlighted



(a) CS:GO (Source) Unseen games (Target) **(b)** Minecraft (Source) Unseen games (Target)

Fig. 5: Behaviour classification accuracy across 3 behaviour categories when transferring from (a) CS:GO (FPS game) and (b) Minecraft (non-FPS game) to unseen FPS games. Although BehAVE (aligned) encodings perform slightly worse on source domain test sets than foundation (unaligned) encodings, they show significant improvements in generalisation to target domains, highlighting BehAVE's enhanced transfer capacities.

in green). These findings are in line with contemporary studies, such as those in [2, 37], suggesting that minimising training error leads machines to absorb all correlations, including spurious ones from data biases. As shown in Figure 1a, foundation models retain game-specific style information, leading to the exploitation of spurious correlations and poor generalisation. Conversely, BehAVE reduces data bias impact, enabling generalisation to new test distributions by discarding game-specific information, as seen in Figure 1b.

Transfer from Minecraft (*Similar* Genre). Figure 5b showcases similar benefits when using Minecraft as the source domain, despite its non-shooter nature. While *panning* category obtains minimal improvement, we see a 6% to 22% improvement in classifying *navigation* and *weapon* categories. Although the weapon equipped by the Minecraft player is a pick-axe, the classifier is still able to transfer this knowledge to the gun-related behaviours of SMG-25 games. This indicates that our model, trained on a diverse dataset covering multiple game genres with aligned semantic behaviour patterns, can generalise to new genres with overlapping characteristics. However, while the results demonstrate BehAVE's transferability across genres, this exploration is preliminary; further investigation into cross-genre transferability is left for future work.

6 Discussion

Downstream Applications. BehAVE's zero-shot performance shows promising applications in imitation learning across both first-person shooter and non-shooter game genres. While our focus has been on transferable video understanding, a complementary study could explore representation learning optimised for specific downstream tasks. Supplementary material includes initial experiments on BehAVE's potential in learning inverse dynamics models.

Future Work. Future experiments could employ BehAVE to various game genres, such as fighting, platformer, and driving games, with some, like driving games, requiring additional action preprocessing like discretising steering [41]. Experiments could also assess training across multiple genres and evaluate transferability to unseen ones. Another direction is to keep pace with advancements in foundation models like large language and vision-language Models, deepening action understanding. With more computational resources and a larger team [29], fine-tuning these models could enhance performance and transferability, making BehAVE more adaptable to specific applications.

Scalability. As outlined in Section 4, we operate within a limited compute and with a small team of annotators, hindering data collection expansion and comparisons with end-to-end trained models. BehAVE could potentially perform better with increased data and computational resources; however, our focus is on introducing an accessible method that proves effective even at a smaller scale [29].

Ethical impact. This paper introduces a dataset of fully anonymised annotated FPS gameplay videos. Both the gameplay videos and the annotations were collected in-house from participants in a laboratory setting under a data collection protocol approved by the University Research Ethics Committee of the University of Malta. The protocol was inspired by previously peer-reviewed work cited in Section 2. We acknowledge the emerging trends in machine learning for autonomous weapons research [25] and wish to clarify that this project has no military applications and was not funded by military sources. Additionally, the data collection protocol is exclusively applicable to video games, and the dataset aims to support research in video games and artificial intelligence.

Dataset The dataset utilised in this research was derived from various commercial games. All game data consisting of screen captured game graphics, is the intellectual property of their respective game developers and publishers (see Figure 3). To the best of our knowledge the inclusion of the data in this study falls under the provisions of fair use for the purpose of academic research, analysis, and non-commercial study under US law, and text and data mining (TDM) for research purposes under EU law. The dataset will be publicly available for future studies under a non-commercial license to ensure scientific reproducibility.

7 Conclusion

In this paper, we introduced BehAVE, a video understanding framework that uses a simulator-free domain randomization method, leveraging the inherent variations in graphics and animations in commercial video games. Tested on first-person shooter games, BehAVE effectively aligns video encodings of similar player behaviours across different games. It outperforms foundation models in zero-shot transferability to unseen FPS games during a behaviour classification task, even when trained on games from different genres like Minecraft. BehAVE offers a strong and efficient method for generalising perception across visually diverse environments.

Acknowledgements

We would like to express our gratitude to Tim Pearce for insightful discussions that contributed to the development of this work. Special thanks also go to Roberto Gallotta and Marvin Zammit for their invaluable assistance in collecting the necessary data for our research. Makantasis was supported by Project ERICA (GA: REP-2023-36) financed by the Malta Council for Science & Technology (MCST), for and on behalf of the Foundation for Science and Technology, through the FUSION: R&I Research Excellence Programme. Rasajski, Liapis and Yannakakis were supported by Project OPtiMaL funded by MCST through the SINO-MALTA Fund 2022.

References

1. Apostolidis, E., Balaouras, G., Mezaris, V., Patras, I.: Combining global and local attention with positional encoding for video summarization. In: 2021 IEEE international symposium on multimedia (ISM). pp. 226–234. IEEE (2021)
2. Arjovsky, M., Bottou, L., Gulrajani, I., Lopez-Paz, D.: Invariant risk minimization. arXiv preprint arXiv:1907.02893 (2019)
3. Baker, B., Akkaya, I., Zhokov, P., Huizinga, J., Tang, J., Ecoffet, A., Houghton, B., Sampedro, R., Clune, J.: Video pretraining (vpt): Learning to act by watching unlabeled online videos. Advances in Neural Information Processing Systems **35**, 24639–24654 (2022)
4. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? In: ICML. vol. 2, p. 4 (2021)
5. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
6. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: Carla: An open urban driving simulator. In: Conference on robot learning. pp. 1–16. PMLR (2017)
7. Gu, X., Liu, G., Zhang, X., Tang, L., Zhou, X., Qiu, W.: Infrared-visible synthetic data from game engine for image fusion improvement. IEEE Transactions on Games (2023)
8. Kim, Y.w., Mishra, S., Jin, S., Panda, R., Kuehne, H., Karlinsky, L., Saligrama, V., Saenko, K., Oliva, A., Feris, R.: How transferable are video representations based on synthetic data? Advances in Neural Information Processing Systems **35**, 35710–35723 (2022)
9. Ko, D., Choi, J., Ko, J., Noh, S., On, K.W., Kim, E.S., Kim, H.J.: Video-text representation learning via differentiable weak temporal alignment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5016–5025 (2022)
10. Krähenbühl, P.: Free supervision from video games. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2955–2964 (2018)
11. Leiprecht, S.: Using simulations and domain randomization for autonomous driving. Technical Reports in Computing Science (July 2020), 1–4 (2020)
12. Lewis, M., Jacobson, J.: Game engines. Communications of the ACM **45**(1), 27 (2002)

13. Mishra, S., Panda, R., Phoo, C.P., Chen, C.F.R., Karlinsky, L., Saenko, K., Saligrama, V., Feris, R.S.: Task2sim: Towards effective pre-training and transfer from synthetic data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9194–9204 (2022)
14. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602 (2013)
15. Patil, R., Boit, S., Gudivada, V., Nandigam, J.: A survey of text representation and embedding techniques in nlp. IEEE Access (2023)
16. Pearce, T., Zhu, J.: Counter-strike deathmatch with large-scale behavioural cloning. In: 2022 IEEE Conference on Games (CoG). pp. 104–111. IEEE (2022)
17. Rabin, S.: Introduction To Game Development (Game Development). Charles River Media, Inc. (2005)
18. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
19. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019)
20. Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S.G., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J.T., et al.: A generalist agent. arXiv preprint arXiv:2205.06175 (2022)
21. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (11 2019), `https://arxiv.org/abs/1908.10084`
22. Richter, S.R., Vineet, V., Roth, S., Koltun, V.: Playing for data: Ground truth from computer games. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14. pp. 102–118. Springer (2016)
23. Risi, S., Togelius, J.: Increasing generality in machine learning through procedural content generation. Nature Machine Intelligence **2**(8), 428–436 (2020)
24. Rousseeuw, P.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics **20**, 53–65 (1987)
25. Simmons-Edler, R., Badman, R., Longpre, S., Rajan, K.: Ai-powered autonomous weapons risk geopolitical instability and threaten ai research. arXiv preprint arXiv:2405.01859 (2024)
26. Song, Y.C., Naim, I., Al Mamun, A., Kulkarni, K., Singla, P., Luo, J., Gildea, D., Kautz, H.A.: Unsupervised alignment of actions in video with text descriptions. In: IJCAI. pp. 2025–2031 (2016)
27. Taesiri, M.R., Macklon, F., Bezemer, C.P.: Clip meets gamephysics: Towards bug identification in gameplay videos using zero-shot transfer learning. In: Proceedings of the 19th International Conference on Mining Software Repositories. pp. 270–281 (2022)
28. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world. In: 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 23–30. IEEE (2017)
29. Togelius, J., Yannakakis, G.N.: Choose your weapon: Survival strategies for depressed ai academics [point of view]. Proceedings of the IEEE **112**(1), 4–11 (2024)

30. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. Advances in neural information processing systems **35**, 10078–10093 (2022)
31. Trivedi, C., Liapis, A., Yannakakis, G.N.: Contrastive learning of generalized game representations. In: 2021 IEEE Conference on Games (CoG). pp. 1–8. IEEE (2021)
32. Trivedi, C., Makantasis, K., Liapis, A., Yannakakis, G.N.: Towards general game representations: Decomposing games pixels into content and style. arXiv preprint arXiv:2307.11141 (2023)
33. Wang, J., Lan, C., Liu, C., Ouyang, Y., Qin, T., Lu, W., Chen, Y., Zeng, W., Philip, S.Y.: Generalizing to unseen domains: A survey on domain generalization. IEEE transactions on knowledge and data engineering **35**(8), 8052–8072 (2022)
34. Wang, L., Huang, B., Zhao, Z., Tong, Z., He, Y., Wang, Y., Wang, Y., Qiao, Y.: Videomae v2: Scaling video masked autoencoders with dual masking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14549–14560 (June 2023)
35. Wang, M., Deng, W.: Deep visual domain adaptation: A survey. Neurocomputing **312**, 135–153 (2018)
36. Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Yuan, L., Jiang, Y.G.: Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6312–6322 (2023)
37. Wenzel, F., Dittadi, A., Gehler, P., Simon-Gabriel, C.J., Horn, M., Zietlow, D., Kernert, D., Russell, C., Brox, T., Schiele, B., et al.: Assaying out-of-distribution generalization in transfer learning. Advances in Neural Information Processing Systems **35**, 7181–7198 (2022)
38. Wu, C.Y., Krahenbuhl, P.: Towards long-form video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1884–1894 (2021)
39. Wydmuch, M., Kempka, M., Jaśkowski, W.: Vizdoom competitions: Playing doom from pixels. IEEE Transactions on Games **11**(3), 248–259 (2018)
40. Xu, H., Ghosh, G., Huang, P.Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., Feichtenhofer, C.: Videoclip: Contrastive pre-training for zero-shot video-text understanding. arXiv preprint arXiv:2109.14084 (2021)
41. Xu, H., Gao, Y., Yu, F., Darrell, T.: End-to-end learning of driving models from large-scale video datasets. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2174–2182 (2017)
42. Yue, X., Zhang, Y., Zhao, S., Sangiovanni-Vincentelli, A., Keutzer, K., Gong, B.: Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2100–2110 (2019)
43. Zhu, Y., Li, X., Liu, C., Zolfaghari, M., Xiong, Y., Wu, C., Zhang, Z., Tighe, J., Manmatha, R., Li, M.: A comprehensive study of deep video action recognition. arXiv preprint arXiv:2012.06567 (2020)
44. Zhu, Z., Hou, J., Wu, D.O.: Cross-modal orthogonal high-rank augmentation for rgb-event transformer-trackers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22045–22055 (2023)