Proceedings of the Twenty-First AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2025)

Adaptive Visual Navigation Assistant in 3D RPGs

Kaijie Xu, Clark Verbrugge

Department of Computer Science, McGill University
Montreal, Quebec, Canada
kaijie.xu2@mail.mcgill.ca, clump@cs.mcgill.ca

Abstract

In complex 3D game environments, players rely on visual affordances to spot map transition points. Efficient identification of such points is important to client-side auto-mapping, and provides an objective basis for evaluating map cue presentation. In this work, we formalize the task of detecting traversable Spatial Transition Points (STPs)-connectors between two sub regions-and selecting the singular Main STP (MSTP), the unique STP that lies on the designer-intended critical path toward the player's current macro-objective, from a single game frame, proposing this as a new research focus. We introduce a two-stage deep-learning pipeline that first detects potential STPs using Faster R-CNN and then ranks them with a lightweight MSTP selector that fuses local and global visual features. Both stages benefit from parameter-efficient adapters, and we further introduce an optional retrieval-augmented fusion step. Our primary goal is to establish the feasibility of this problem and set baseline performance metrics. We validate our approach on a custom-built, diverse dataset collected from five Action RPG titles. Our experiments reveal a key trade-off: while full-network fine-tuning produces superior STP detection with sufficient data, adapter-only transfer is significantly more robust and effective in low-data scenarios and for the MSTP selection task. By defining this novel problem, providing a baseline pipeline and dataset, and offering initial insights into efficient model adaptation, we aim to contribute to future AI-driven navigation aids and data-informed level-design tools.

Code — https://github.com/Nortrom1213/VisualGuidance

Introduction

The design of game levels profoundly shapes player experience, directly impacting engagement, learning, and progression in virtual worlds (Yalçınkaya-Doma 2024). A key component of successful level design is the effective use of visual cues to guide players, particularly in complex 3D environments. Such guidance aims to enable intuitive navigation and reduce cognitive load, often by leveraging environmental affordances—properties suggesting action possibilities—to direct attention and movement (Gibson 2014; Irshad, Perkis, and Azam 2021). Designers use visual strategies like lighting, color, landmarks, and architecture to subtly orient players and indicate paths, fostering deeper immersion than explicit aids might allow (Dillman et al. 2018).

Although visual guidance principles are well studied in design and HCI, most analyses are qualitative or center on player responses rather than the cues' intrinsic, measurable properties (Interaction Design Foundation (IxDF) 2023). Existing work probes perceptions (Bøe 2024) or compares task completion under varied cues (Filén and Gemal 2024), but a systematic, computational account of what makes a visual cue effective for navigation remains largely open—highlighting an opportunity for data-driven methods to quantify the visual signatures of navigational cues.

This paper investigates the formalization and automation of recognizing such visual cues—Spatial Transition Points (STPs), the passable links between map regions—and the Main STP (MSTP), the most prominent STP that represents the designer's intended route to the player's current objective, all from a single game frame. We propose this as a new research focus, hypothesizing that learnable visual metrics within game environments correlate with these navigational elements. Identifying such metrics presents significant potential for applications like (1) post-development navigation assistants that subtly highlight implicit cues for struggling players (Figure 1a), and (2) pre-development design aids offering real-time feedback on visual clarity (Figure 1b).

To explore this, we introduce a two-stage deep learning pipeline that analyzes visual data. The system first detects potential STPs and then identifies the MSTP, employing contemporary computer vision techniques and parameter-efficient adaptation methods. The primary aim of our work, validated on a newly custom-built, diverse dataset from five Action RPGs (*Dark Souls I, II, III*, *Elden Ring*, *Black Myth: Wukong*), is to establish the initial feasibility and tractability of this novel problem by evaluating our baseline solution under varied data conditions. Our key contributions are:

- The formulation of automated STP/MSTP recognition from visual game data as a new research problem, with a proposed two-stage pipeline as a baseline solution.
- A new, diverse, richly annotated multi-game dataset for visual navigation analysis, to be publicly released to foster research and comparative studies.
- Initial empirical validation confirming the pipeline efficacy, particularly in low-resource, cross-domain settings.

(a) Player assistance: Green box highlights a hidden optimal boss route that players often miss; red box marks the normal main route.

(b) Design evaluation: Green marks a secondary STP; red the designer-intended MSTP; yellow a deceptive, impassable "air wall."

Figure 1: Examples of the system's application: (a) Assisting players by identifying easily missed STPs like hidden paths. (b) Aiding designers by evaluating visual guidance towards the intended MSTP versus potentially misleading alternatives.

Background

Our work builds upon several established areas of research. This section reviews relevant literature on visual analysis in game environments, computational methods for player, level analysis, and the application of deep learning in games.

Visual Perception, Affordances, and Wayfinding

The way players perceive and navigate virtual environments is deeply influenced by the visual information presented to them. Game environments often tell stories and guide players through their spatial design, a practice termed environmental storytelling (Fernández-Vara 2011). This relates to Gibson's (2014) ecological view of affordances, where environmental forms themselves suggest how one might interact with them. In games, these affordances are often communicated through carefully designed visual cues that signal what can be interacted with, where to look, or where to go (Dillman et al. 2018). Effective wayfinding in games relies on players' ability to form cognitive maps of the environment, often aided by landmarks, distinct paths, and spatial organization (Lynch 2023) in urban design and applicable to virtual spaces (Yalçınkaya-Doma 2024). HCI research for games investigates how these elements contribute to spatial knowledge and navigation efficiency, particularly in immersive virtual reality environments that emphasize embodied interaction (Lin et al. 2024; Marcus et al. 2025). Our focus on STPs and MSTPs is an attempt to computationally model the perception of critical navigational affordances based on their visual presentation.

Player Guidance and Level Analysis

Beyond manual design principles, computational methods are increasingly used to analyze and enhance player navigation within game levels. Player modeling, for instance, aims to capture player states computationally, enabling adaptive systems that can modify guidance strategies (Yannakakis and Togelius 2018; Hare and Tang 2022). Such models might analyze gameplay data to identify navigational challenges and inform personalized support. Separately, automated playtesting uses AI agents to evaluate level traversability or to identify gameplay imbalances by training agents to navigate based on environmental stimuli (Miller 2024). Other research applies graph-based or metric-driven analyses to evaluate the topological structure of game levels, which can indirectly relate to their navigability (Xu and Verbrugge 2025; Omidshafiei et al. 2020; Naying et al. 2023). Our research extends these efforts by focusing on the direct visual analysis of game scenes for explicit navigational markers, distinct from approaches on player telemetry or abstract spatial graphs.

Deep Learning for Visual Scene Understanding

Deep learning techniques, particularly Convolutional Neural Networks (CNNs), have significantly enhanced machines' abilities to parse complex visual information. These methods are applied to tasks such as object detection, semantic segmentation, and overall scene understanding (Betsas et al. 2025; Qi and Li 2023). In game research, deep learning has been used for diverse applications, from procedural content generation to controlling non-player characters and analyzing player affect (Liu et al. 2021; Mehta 2025). Object detection systems such as Faster R-CNN or YOLO have also been adapted to identify elements in games; however, their effectiveness often depends on specialized fine-tuning due to diverse artistic styles and visual complexities (Jung, Yang, and Min 2021; Hyde-Smith 2023). Furthermore, understanding dynamic scenes and player attention within them is an active area of research, with models aiming to predict human players' focus (Yazdani et al. 2025). The challenge of adapting models trained on one visual domain (e.g., real-world images or a specific game) to another (e.g., a different game with a distinct art style) is addressed by domain adaptation techniques, which are crucial for creating broadly applicable visual recognition systems for games (Csurka 2017; Patel et al. 2015). Our use of deep learning for detecting STPs/MSTPs and employing adapter modules for fine-tuning aligns with these trends, aiming to create a robust visual analysis tool for diverse game environments.

Methodology

In this section, we formalize the STP/MSTP detection task and present our two-stage pipeline—STP detection, MSTP selection, and optional retrieval-augmented fusion (RAF).

Problem Definition

A *Spatial Transition Point (STP)* is any traversable doorway, ladder, corridor, or passage that links two distinct map regions. A *Main Spatial Transition Point (MSTP)* is the unique STP on the designer-defined critical path toward the current macro-objective (inferred here as the next boss or primary level goal). We restrict our study to ARPGs with a clearly defined target (e.g., a boss encounter) in each single level for simplicity. The objective of this work is to develop an automated pipeline that, given a single input frame, (i) detects all STPs and (ii) identifies the single MSTP among them.

Pipeline Overview

We proceed in three logical blocks shown in Figure 2:

1. **STP Detection:** a parameter-efficient Adapter head fine-tunes Faster R-CNN to propose spatial transition points.
2. **MSTP Selection:** a lightweight selector network combines local patch features and global image features (both processed through dedicated network branches, with their fusion augmented by the same bottleneck Adapter mechanism to rank each STP proposal.
3. **Retrieval-Augmented Fusion (optional):** we combine selector scores with cosine-based retrieval scores from an offline feature bank, generating the final MSTP choice.

We adopt this two-stage approach rather than direct simultaneous classification because identical physical structures or visual cues may represent different navigation significance depending on their spatial relationships, viewpoint, or surrounding visual context.

Stage 1: STP Detection

We adopt Faster R-CNN (Ren et al. 2015) with a ResNet50-FPN backbone. Given an RGB frame $\mathbf{I}$, the detector predicts K bounding boxes $\{\mathbf{b}_i\}_{i=1}^{K}$ and confidence scores $\{s_i^{\text{det}}\}_{i=1}^{K}$. The network is trained with standard Region Proposal Network (RPN) + classification + regression losses.

Stage 2: MSTP Selection With Global Context

For each detector proposal $\mathbf{b}_i$ we crop a 224×224 local patch and pass it through a ResNet18 branch, producing a feature vector $\mathbf{f}_i^{\text{loc}} \in \mathbb{R}^{512}$. In parallel, a 64×64 thumbnail of the whole frame feeds a lightweight CNN to obtain a global feature vector $\mathbf{f}^{\text{glob}} \in \mathbb{R}^{512}$. This CNN, whose architecture is detailed in Figure 2 (Global Branch), transforms the thumbnail into a global feature vector $\mathbf{f}^{\text{glob}} \in \mathbb{R}^{512}$ followed by ReLU activation. The local feature vector $\mathbf{f}_i^{\text{loc}}$ and the global feature vector $\mathbf{f}^{\text{glob}}$ are concatenated, forming a 1024-dimensional vector $[\mathbf{f}_i^{\text{loc}}; \mathbf{f}^{\text{glob}}]$. This combined vector is then processed by an Adapter module with a bottleneck dimension of $r = 256$, resulting in the refined feature vector $\mathbf{f}_i \in \mathbb{R}^{1024}$:

$$\mathbf{f}_i = \text{Adapter}\big([\mathbf{f}_i^{\text{loc}}; \mathbf{f}^{\text{glob}}]\big).$$

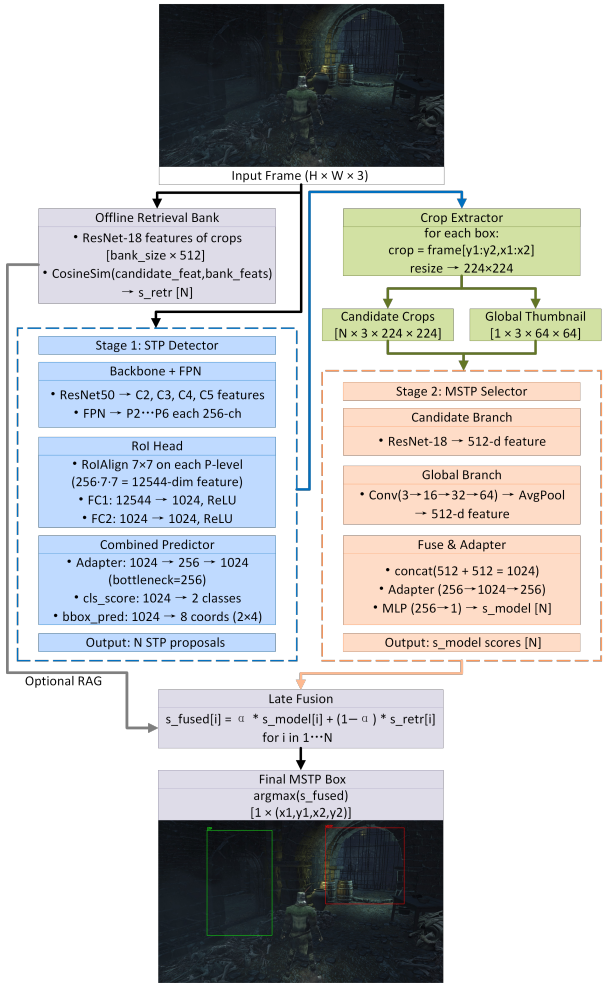


Figure 2: Top: example input frame. Center: the two-stage pipeline. (1) Offline Retrieval Bank holds pre-computed ResNet18 embeddings of annotated STP/MSTP crops. (2) STP Detector (Faster R-CNN + Adapter-augmented head) outputs N candidate boxes. (3) For each box, we extract a 224×224 local crop and a 64×64 global thumbnail. (4) MSTP Selector fuses 512-d local and 512-d global features via an Adapter bottleneck and a two-layer MLP, yielding scores s_i^{sel}. (5) We compute retrieval scores s_i^{ret} by cosine similarity against the bank, then form $s_i^{\text{final}} = \alpha\, s_i^{\text{sel}} + (1 - \alpha)\, s_i^{\text{ret}}$. The box with highest s_i^{final} is returned as the MSTP.

The resulting vector $\mathbf{f}_i$ is then passed through a two-layer Multi-Layer Perceptron (MLP) to output a scalar score s_i^{sel}. This MLP consists of a linear layer projecting from 1024 to 256 dimensions followed by a ReLU activation, and a final linear layer projecting to a single scalar score. Cross-entropy loss, applied to the scores of all candidate proposals, enforces the ground-truth MSTP to rank highest.

ResNet50-FPN was chosen for its strong performance in object detection tasks, while a lighter ResNet18 was selected for local feature extraction to maintain efficiency in the MSTP selection stage. The 64×64 thumbnail provides a

global context without significant computational overhead.

Parameter-Efficient Adapter Fine-tuning

To enable fast adaptation to a new game with limited images, we insert a bottleneck Adapter following Houlsby (2019):

$$\mathbf{x}' = \mathbf{x} + \mathbf{W}_{\text{up}}\big(\sigma(\mathbf{W}_{\text{down}}\,\mathbf{x})\big), \tag{1}$$

where $\mathbf{W}_{\text{down}} \in \mathbb{R}^{r\times d}$, $\mathbf{W}_{\text{up}} \in \mathbb{R}^{d\times r}$, $r \ll d$. Here, σ represents the ReLU activation function. During normal training, the adapter parameters are initialized to zero, reducing Eq. (1) to the identity mapping without affecting capacity. For adapter fine-tuning, we freeze the backbone and optimize only $\{\mathbf{W}_{\text{down}}, \mathbf{W}_{\text{up}}\}$ and the prediction heads.

Retrieval-Augmented Late Fusion

To further boost cross-game robustness, we build an *offline feature bank* $\mathcal{B} = \{(\mathbf{f}_j, \ell_j)\}_{j=1}^{M}$, where each $\mathbf{f}_j \in \mathbb{R}^{512}$ is a ResNet18 embedding of an annotated region, and ℓ_j is its corresponding class label (STP or MSTP) and source game information. This bank stores embeddings of STP/MSTP regions from various titles. At inference, we:

1. extract a feature $\hat{\mathbf{f}}_i$ for each candidate crop via the same ResNet18 branch,
2. compute a retrieval score $s_i^{\text{ret}} = \max_{(\mathbf{f}_j,\ell_j)\in\mathcal{B}} \cos(\hat{\mathbf{f}}_i, \mathbf{f}_j)$,
3. fuse scores $s_i^{\text{final}} = \alpha\, s_i^{\text{sel}} + (1-\alpha)\, s_i^{\text{ret}}, \quad \alpha \in [0,1]$,
4. pick the candidate with the largest s_i^{final} as the predicted MSTP.

This late-fusion design requires no additional training and empirically improves recall on unseen maps.

Data Collection and Annotation

In this section, we describe how we assembled our multi-game ARPG screenshot dataset, labeled all STPs, and identified the MSTP. A full two-stage protocol with expert review is planned for future releases.

Dataset Construction

We manually constructed a dataset from four commercially released third-person ARPGs (*Dark Souls I*, *II*, *III*, and *Elden Ring*), chosen for their intricate, nonlinear level architectures and clearly defined macro-objectives. In addition, to verify cross-franchise generalization, we captured some frames from *Black Myth: Wukong*, which—like the Souls series—is a third-person ARPG with explicit end-level targets that facilitate MSTP annotation. BMW's use of photogrammetric 3D scanning produces highly realistic, "real-world" visuals; while this dramatically enhances immersion, it also sometimes leads to spurious visual cues that can mislead naive transition-point detectors. All screenshots were taken at predetermined camera positions to ensure uniform coverage of potential transition areas. We disabled all in-game UI and held the player avatar in a fixed pose so that only the underlying environment informed our models. Each decision point was captured exactly once, ensuring that no single navigational junction appears in more than one frame across the dataset.

Annotation Protocol

Each frame was manually annotated by an expert player (one of the authors) who drew polygonal bounding boxes around every STP and marked the single MSTP according to the hierarchy described below. To maximize consistency, the annotator followed guidelines derived from established player conventions and community-verified walkthroughs.

STP Definition and Criteria An STP is defined as a visually distinct, traversable region that connects two separate map sub-regions. For annotation, an STP must have:

- **Clear Spatial Separation:** Evident environmental changes (in texture, lighting, geometry, or theme) signaling a transition.
- **Explicit Visual Cues:** Presence of unambiguous indicators such as arches, doors, stairways, or other common navigational affordances.
- **Functional Navigational Role:** The region must serve a recognizable transitional purpose aligned with game progression or established level structure.

MSTP Selection Hierarchy The unique MSTP within a frame was identified from the annotated STPs using a hierarchical filtering process:

1. **Navigational Continuity:** Verification of a direct, standard traversable path.
2. **Path Efficiency:** Prioritization of the shortest verified route to the primary level objective, confirmed via in-game path tracing.
3. **Route Connectivity:** Preference for STPs that link multiple alternative routes or offer greater navigational flexibility, if ambiguity remained.
4. **Designer Intent Confirmation:** In rare, highly ambiguous cases, consultation of official game documentation or expert design knowledge.

Handling Ambiguities Annotation ambiguities were systematically addressed: Overlapping STPs were merged if structurally similar, otherwise annotated separately with documented overlap. Regions with unclear boundaries received an uncertainty rating and explanatory notes. If multiple STPs were viable MSTP candidates, the above hierarchy was strictly applied to select one. STPs accessible only via non-standard movement mechanics (e.g., obscure sequence breaks) were generally excluded from MSTP candidacy unless critical to intended progression and supported by design documentation.

Final Dataset Summary

In total, our annotated corpus comprises 699 frames: 59 from *Dark Souls I*, 100 from *Dark Souls II*, 184 from *Dark Souls III*, 230 from *Elden Ring*, and 126 from *Black Myth: Wukong*. For *Dark Souls II* and *Black Myth: Wukong*, we adopt a 20%/80% train/test split to assess performance under limited-data conditions; for the remaining three titles, we use an 80%/20% split. We will publicly release this entire dataset under an open-access license and maintain an online repository with ongoing updates and expansions.

Experiments and Results

To evaluate the effectiveness and generalizability of our proposed two-stage pipeline, we perform comprehensive experiments using three distinct evaluation strategies:

- **Original dataset:** Comprising Dark Souls I, Dark Souls III, and Elden Ring (*DS I, DS III, ER*), split randomly into 80% training and 20% testing sets. This assesses model efficacy under standard training conditions.
- **Novel dataset (DS II):** Only 20% of samples used as a training set, to assess model adaptability and generalization performance under limited-data conditions.
- **Novel dataset (BMW):** Comprising Black Myth: Wukong (*BMW*), 126 frames with a 20%/80% train/test split, used to validate cross-game transferability on a realistic, third-person ARPG outside the Souls lineage.

We explore the following model variants:

A. Full training on the original dataset (**Full Version**)

B. Adapter-only on the original dataset (**Adapter Version**)

C. Continuing full training on the novel dataset after full training on original dataset (**Continue on Full Version**)

D. Adapter-only fine-tuning on the novel dataset after full training on original dataset (**Adapter on Full Version**)

E. Full training solely on the novel dataset (**New Full**)

F. Adapter-only training solely on the novel dataset (**New Adapter Version**)

We evaluate STP Detection (**Model 1**) and MSTP Selection (**Model 2**) across these datasets using the following standard metrics. Detailed results are in Tables 1 and 2.

Evaluation Metrics We assess Model 1 using standard object detection metrics:

- **mAP@IoU>0.5 (mean Average Precision):** Measures detection quality, balancing precision and recall at 0.5 IoU (Intersection over Union) threshold. Higher is better.
- **Mean IoU:** Assesses localization accuracy by averaging the IoU of correctly detected STPs with their ground-truth boxes. Higher values indicate better bounding box fit.
- **Recall:** The proportion of all STPs that the model successfully identifies. Higher values mean fewer missed STPs.
- **Composite Score:** For Model 1 training, this average of the metrics guides checkpoint selection, balancing detection quality and completeness.

For Model 2 (MSTP Selection), a classification task of choosing the correct MSTP from candidates, we report:

- **Accuracy:** The percentage of times the model correctly identifies the ground-truth MSTP. Given our setup (one true MSTP per candidate set), this directly reflects correct navigational choice.

For all experiments, the training split was further divided into 80%/20% train/validation subsets (the held-out test split was never used for tuning). Checkpoints were selected based on the highest composite validation score (mAP@0.5, mean IoU, and recall) for Model 1 and validation accuracy for Model 2, with early stopping applied.

Retrieval-Augmented Late Fusion To further enhance robustness—especially under domain shift to DS II and BMW—we assemble our offline feature bank $\mathcal{B}$ from three complementary sources:

- **Cross-game core bank:** For each title in the Original dataset (DS I, DS III, ER), we extract all annotated STP/MSTP region embeddings via the ResNet18 branch, compute a quality score (L2 norm plus 0.5×std), sort descending, and retain the top 100 per title.
- **DS II support bank:** We append *all* STP/MSTP embeddings from the DS II training split.
- **BMW support bank:** Likewise, we include *all* STP/MSTP embeddings from the BMW training split.

At test time, the retrieval score s_i^{ret} and final fused score s_i^{final} (using $\alpha = 0.8$) are computed for each candidate crop according to the procedure detailed in the previous section. The candidate with the highest s_i^{final} is selected as the MSTP.

Our hybrid feature bank, merging a cross-game core with specific support for DS II and BMW, improved performance on these unseen games without further model training. Although RAF's current boost to scores is modest (Table 2), its key advantage is being a training-free component. We see RAF as a promising avenue for systems that learn on the fly; in practice, the bank could be updated with new features from errors or new game areas, allowing continuous adaptation. For this study, however, we used a fixed bank to fairly test RAF's baseline contribution and its role as an alternative to adapter fine-tuning.

Training details Model 1 (STP detection) used Faster R-CNN with ResNet50-FPN, SGD (lr=5×10^{-3}, momentum=0.9, weight decay=5×10^{-4}), batch size 4, and horizontal flip augmentation ($p = 0.5$). Model 2 used a ResNet-18 local branch and a lightweight global CNN, SGD (lr=1×10^{-3}, momentum=0.9, weight decay=5×10^{-4}), batch size 4, 500 epochs, and adapter bottleneck $r = 256$; the backbone was frozen except for adapters and the prediction head. Local crops were 224×224 and global thumbnails 64×64. Unless noted otherwise, the random seed was fixed at 42, with multi-seeds starting from 0.

STP Detection Results (Model 1)

Learning Curves Figure 3 presents training and validation curves for Model 1. Detecting STPs poses unique challenges: they are functionally defined, visually diverse, and often subtle, unlike typical objects in standard computer vision benchmarks. Our dataset is also modest in scale compared to large CV datasets. Thus, while absolute mAP and Recall values may seem low, our focus is on the relative performance of different training strategies and their generalization capabilities, which these curves clearly illustrate.

On the original dataset (Figure 3a), the Full-network model consistently surpasses the Adapter-only variant. The gap between training and validation curves for the Full model indicates some overfitting, yet its validation performance is stable and superior, confirming the need for end-to-end fine-tuning with sufficient data.

Model Version	Test Set	mAP@IoU>0.5 (%)	Mean IoU (%)	Recall (%)
Original Dataset (DS I, DS III, ER)				
Full Version	Original	**22.04**	**72.05**	**36.31**
Adapter Version	Original	0.04	59.99	10.83
Novel Dataset DS II				
Full Version	DS II	16.87	71.62	29.20
Adapter Version	DS II	0.05	60.34	8.03
Continue on Full	DS II	9.94	72.03	25.55
Adapter on Full	DS II	**18.45**	**73.10**	**32.40**
New Full Version	DS II	3.43	71.03	13.87
New Adapter Version	DS II	0.07	58.97	20.44
Novel Dataset BMW				
Full Version	BMW	7.64	63.95	13.58
Adapter Version	BMW	0.14	57.80	13.58
Continue on Full	BMW	8.34	61.89	35.80
Adapter on Full	BMW	**9.50**	**64.50**	**38.20**
New Full Version	BMW	4.51	61.10	29.01
New Adapter Version	BMW	0.00	54.13	0.62

Table 1: Model 1 (STP Detection) Results. Bold values are the best results across models in each dataset category.

Model Version	Test Set	Accuracy (%)	+RAF Accuracy (%)
Original Dataset (DS I, DS III, ER)			
Full Version	Original	77.89	80.00
Adapter Version	Original	**80.00**	80.00
Novel Dataset DS II			
Full Version	DS II	66.25	72.50
Adapter Version	DS II	73.75	73.75
Continue on Full Version	DS II	73.75	–
Adapter on Full Version	DS II	67.50	–
New Full Version	DS II	72.50	–
New Adapter Version	DS II	**75.00**	–
Novel Dataset BMW			
Full Version	BMW	72.28	74.26
Adapter Version	BMW	**74.26**	74.26
Continue on Full Version	BMW	**74.26**	–
Adapter on Full Version	BMW	72.28	–
New Full Version	BMW	70.30	–
New Adapter Version	BMW	71.29	–

Table 2: Model 2 (MSTP Selection) Accuracy

In contrast, on the limited-data DS II split (Figure 3b), models trained from scratch ("Full on New" and "Adapter on New") show near-zero mAP and Recall. This failure to learn is expected given the extreme data scarcity (20 training images from DS II), highlighting the difficulty of learning robust features without prior knowledge. Continuing full fine-tuning ("Full→Full on New") offers marginal gains. However, adapter-only transfer ("Full→Adapter on New") achieves higher performance across all metrics, showing that parameter-efficient adapters best preserve and transfer cross-game knowledge under severe data constraints.

Taken together, these curves confirm our core finding for STP detection: full-network training is essential when data are plentiful, but under extreme low-data conditions, adapter-only fine-tuning offers the most robust cross-game transfer. For clarity, we focus our curve discussion on Model 1, as Model 2 and the BMW dataset experiments show similar relative orderings, but with smaller absolute differences.

Analysis of STP Detection (Model 1) We did not include generic object-detection baselines because such models achieved a Mean IoU below 10% in our experiments, providing virtually no usable information for this task and would not offer a meaningful comparison. On the **original dataset**, the end-to-end fine-tuned detector (Full Version) achieves its highest performance (mAP 22.04%, IoU 72.05%, Recall 36.31%), while the Adapter-only variant essentially fails (mAP 0.04%, Recall 10.83%). This large gap confirms that adapters alone lack sufficient capacity for accurate STP localization when abundant data are available.

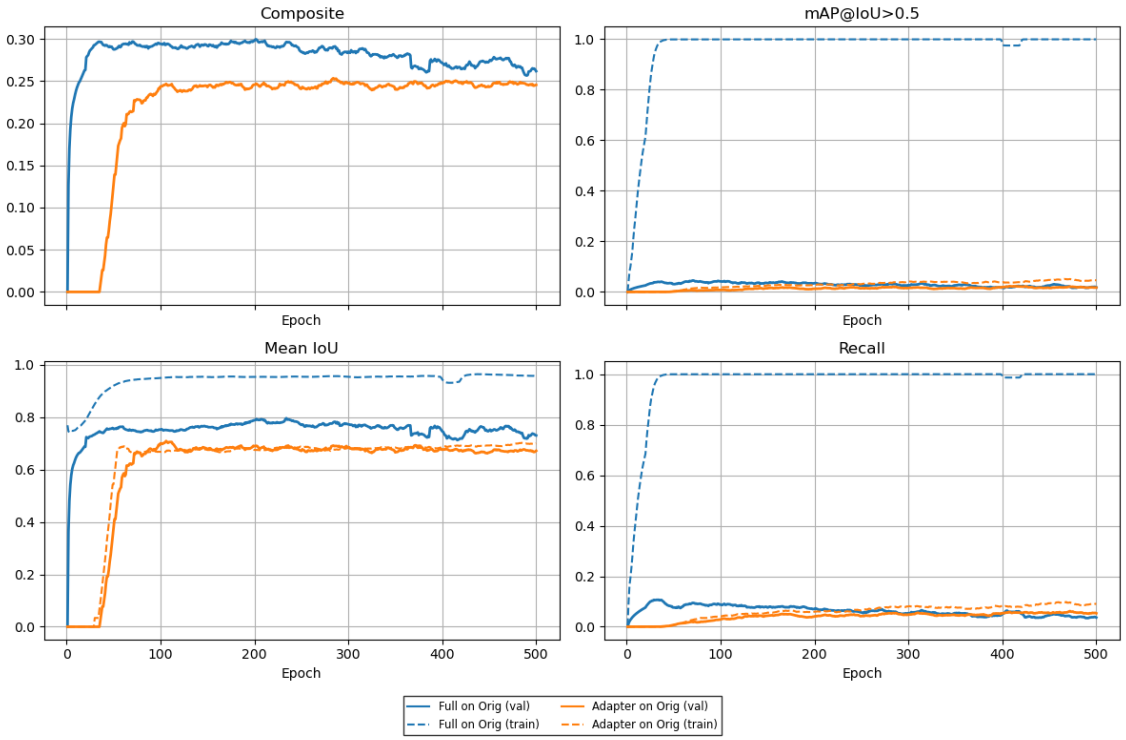
Under the limited-data **DS II** split, all models trained *from scratch* on DS II collapse (New Full: mAP 3.43%, Recall 13.87%; New Adapter: mAP 0.07%, Recall 20.44%). Continuing full fine-tuning (Continue on Full) recovers some performance (mAP 9.94%, Recall 25.55%) but still lags behind zero-shot transfer (Full Version: mAP 16.87%, Recall 29.20%). Notably, adapter-only transfer (Full→Adapter on New) now surpasses zero-shot full transfer (mAP 18.45% vs. 16.87%, Recall 32.40% vs. 29.20%), confirming that parameter-efficient adapters can not only preserve but even improve cross-game detection under extreme scarcity.

On the **BMW** test split, a similar pattern emerges: scratch-trained models remain weak (New Full: mAP 4.51%, Recall 29.01%; New Adapter: mAP 0.00%, Recall 0.62%), and zero-shot full transfer yields moderate scores (mAP 7.64%, Recall 13.58%). Continue-on-Full again boosts Recall (mAP 8.34%, Recall 35.80%), but adapter-only transfer (Full→Adapter on New) now achieves the best (mAP 9.50%, Recall 38.20%), outpacing both zero-shot and full fine-tuning. This reinforces our core finding: under severe data scarcity—even in brand-new domain—the adapter-only strategy delivers the most robust cross-game STP detection.

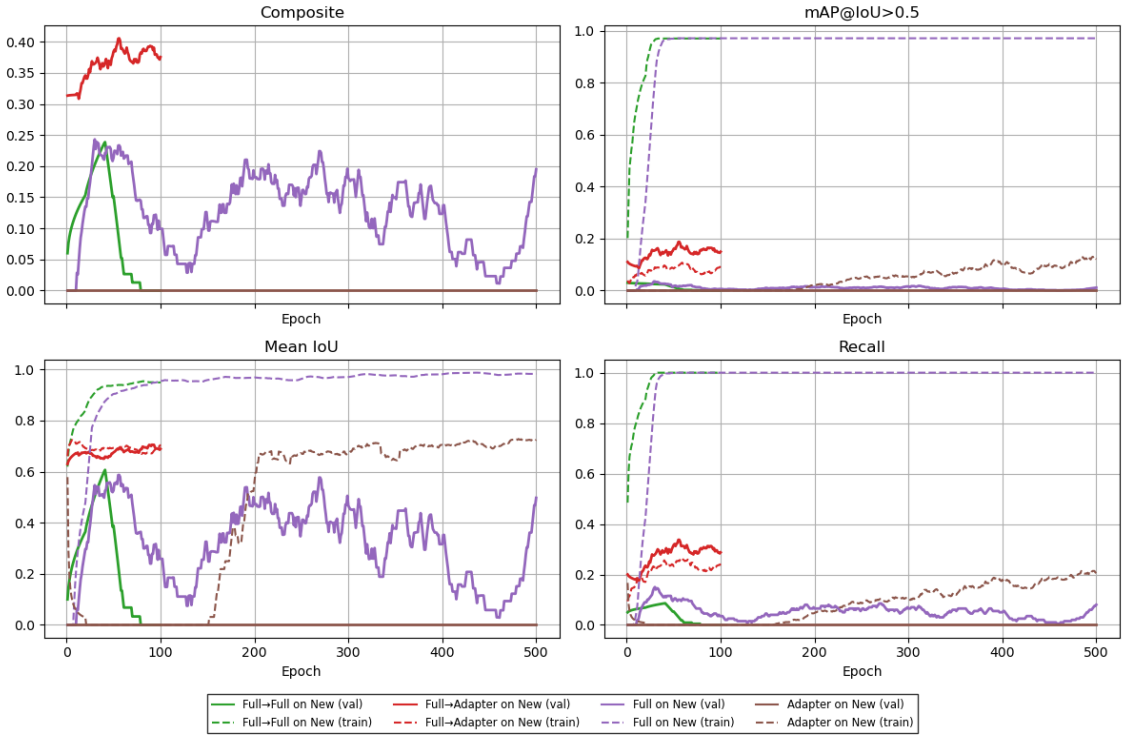
MSTP Selection Results (Model 2)

On the original Souls dataset, the Adapter-only variant achieves 80.00% accuracy—slightly above the Full model's 77.89%. Retrieval fusion raises the Full model's accuracy to 80.00%, perfectly matching the Adapter baseline, while it has no effect on the Adapter-only variant. This confirms that for the relatively simple MSTP classification task, lightweight adapters already capture the necessary features, and semantic retrieval cues are largely redundant. We also evaluated the MSTP selection model using five random seeds. We compare against two simple baselines: (i) Random-1/k, which uniformly selects among the k detector proposals in a frame, and (ii) Center-most, which chooses the candidate whose box center is closest to the image center. Our method achieved a mean accuracy of 76.62% ± 2.96% (95% CI [73.24%, 80.00%]), outperforming the strongest naive baseline (Center-most, 70.52%). A paired McNemar's test yielded $p = 0.04$, indicating that the improvement is statistically significant at the 5% level. The relatively low variance across seeds further demonstrates the robustness and reliability of our approach.

Under extreme scarcity in **DS II** (20% train), the adapter-only model again leads with 73.75% accuracy, compared to 66.25% for the zero-shot Full model. Retrieval fusion further boosts the Full model to 72.50%, nearly closing the gap. Continuing full fine-tuning (73.75%) matches the Adapter baseline, whereas scratch-trained models (New Full: 72.50%; New Adapter: 75.00%) remain competitive but do not surpass the adapter-only strategy. These results

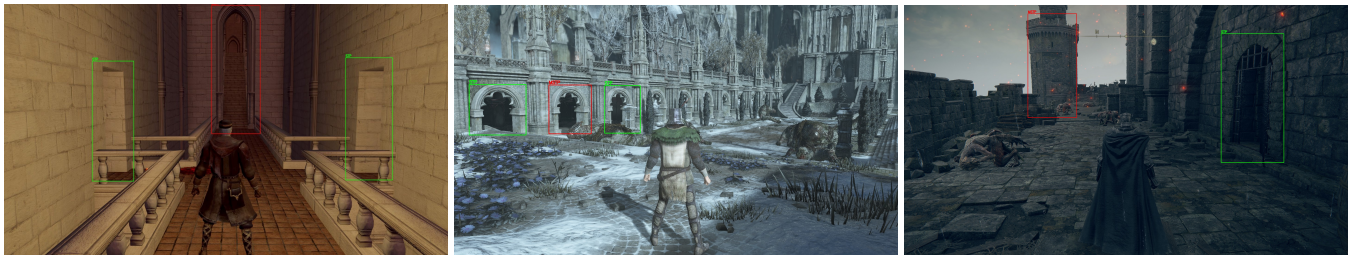


(a) Validation on the original dataset (*DS I, III, ER*).



(b) Validation on the limited-data DS II split (20% train / 80% val).

Figure 3: Model 1 (STP detection) training and validation curves. For each setting we plot the *validation* Composite score, mAP@IoU>0.5, Mean IoU and Recall (solid lines), together with the corresponding *training* curves (dashed). In (a) we compare Full vs. Adapter on the rich, original dataset; in (b) we compare four variants on the scarce DS II split.



(a) DS I (good): in a symmetric corridor, the doorway with the upward staircase breaks the symmetry, picked as the MSTP.

(b) DS III (bad): repeated arched openings produce multiple high-confidence STP candidates, while the real route is the far staircase.

(c) Elden Ring (good): the tower's strong contrast against the background of sky leads to a high score, thus one clear MSTP.

(d) Elden Ring (bad): the model also marks a small, high-contrast window in the wall as an STP, though the correct MSTP is identified.

(e) Black Myth (good): the bright canyon exit MSTP clearly beats another low contrast STP candidate.

(f) Black Myth (bad): the left cliff resembles a ramp, misleading the model; the true path is a plain gap behind the rocks on the right.

Figure 4: Qualitative outputs across four ARPGs. Green boxes mark ground-truth STPs; red boxes show the model's chosen MSTP. Panels (a, c, e) succeed with clear local cues. Panels (b) false high-contrast openings; (d) window misidentified as STP due to misleading perspective; (f) cliff face misread as ramp.

underscore the robustness of adapter-based transfer under low-data conditions, and show that RAF can partially offset a lack of end-to-end fine-tuning.

On the entirely new **BMW** domain, the same ordering persists: the Adapter-only model reaches the highest accuracy (74.26%), and retrieval fusion further elevates the Full model to 74.26% (from its zero-shot 72.28%). Full→Full on BMW also achieves 74.26%, showing that full-network fine-tuning plus RAF yields parity with adapter-only tuning. Scratch-trained baselines (New Full: 70.30%, New Adapter: 71.29%) lag behind. Considering the general task difficulty, less than half the frames (47.2%) contain only one STP; when restricted to multi-candidate cases ($k \geq 2$), all models stay above 50% accuracy, with the adapter-only model around 60%. In sum, adapter-only fine-tuning generalizes as well as—or better than—full-network training across both Souls and non-Souls ARPGs, with RAF providing a consistent but modest uplift to the Full models.

Real-Time Navigation Pilot Study

To verify that our offline metrics translate into usable in-game assistance, we conducted a small-scale pilot study involving live gameplay. A single player (an author) initiated each run at a level start point. At every navigational fork, the player followed the MSTP suggested by the **Full Version** model (trained on the Original Dataset with RAF enabled), using an inference threshold of 0.5 for STP detection to ensure a reasonable number of candidates for MSTP selection. The player only manually overrode the system's guidance if the chosen path led to an unambiguous dead-end, forcing a backtrack. We logged the total number of decision points encountered and the instances of such human intervention.

Across five representative levels—*High Wall of Lothric, Catacombs of Carthus, Grand Archives* (from Dark Souls III); *Forest of Fallen Giants* (from Dark Souls II); and *Black Wind Mountain* (from Black Myth: Wukong)—the system made 63 navigational decisions, requiring manual correction 11 times, yielding an approximate success rate of 82.5%. The per-level intervention/decision counts were: High Wall 2/15, Catacombs 0/8, Grand Archives 2/13, Forest of Fallen Giants 3/10, and Black Wind Mountain 4/17. On a single NVIDIA RTX 4090 GPU, the MSTP selector achieved a median inference latency of less than 20 ms. This performance is well within our decision-making interval of 0.2 seconds (200 ms) used during the live play (game running at 60 FPS), confirming real-time feasibility.

Key Observations from Pilot Study:

(1) *Opposing View Choices:* The system, analyzing single frames, sometimes struggled when competing STPs were in entirely opposite camera directions (e.g., the treasure-mimic alcove versus the rooftop ladder in High Wall). It tended to favor the currently visible or more centrally framed option. We plan to investigate multi-view stitching or short temporal window analysis to address this.

(2) *Potential Upward-Bias Artefact:* The model occasionally over-prioritized upward paths, particularly staircases (e.g., preferring a small upper stair before the Boss in High Wall of Lothric over the main downward progres-

sion stairs). This might stem from a prevalence of upward-leading MSTPs in the training data. Augmenting the dataset should help mitigate this.

(3) *Near-Field Ambiguity in Dense Layouts:* In areas with dense clusters of visually similar STPs in close proximity, such as the numerous short stair flights in the Grand Archives, the model found fine-grained ranking less reliable. This suggests a need for training data with richer examples of such close-range, complex decision points.

(4) *Promising Discovery of Unseen Paths:* Despite the challenges, the system showed encouraging generalization by identifying some designer-intended shortcuts or hidden paths not explicitly present in its training data. Notable examples include a concealed coffin passage in the Catacombs of Carthus and a hidden cave entrance (potentially leading to an optional boss) on Black Wind Mountain in BMW, even when the model was not trained on BMW data.

Qualitative Results

To better understand where our pipeline succeeds and where it fails, Figure 4 shows "good" and "bad" examples from our games. In each pair, green boxes are detected STPs, red boxes the chosen MSTP. Overall, our model proves highly sensitive to local contrast and shape cues—quick to pick out crisp, high-contrast details—but this strength also leads it to over-rely on individual structural features. As a result, incidental windows, decorative arches, or rock textures with strong edge definitions are sometimes treated as intentionally designed transition points. This behavior is reasonable and difficult to avoid given our frame-only, single-region focus, and it remains valuable in photorealistic settings (e.g., BMW) where fine visual differences matter. Mitigating this over-dependence on isolated local cues will be a central challenge for future improvements.

Limitations

Our work has several constraints. The dataset is modest in overall frame count and game title/genre representation, which may affect broader generalization. Our definitions of STP and MSTP are best suited for games with clearly defined main progression paths and can be subjective in more open-ended scenarios. The proposed baseline pipeline has not incorporated more specialized designs for explicit visual reasoning. We also have not evaluated the interpretability of our model's decisions or provided mechanisms to explain its outputs. Furthermore, our experimental validation has primarily focused on static image analysis, complemented by a small-scale pilot study for real-time interaction; we have not conducted large-scale player trials to assess how well our model guides a diverse range of users in practice, which remains necessary for comprehensive evaluation.

Future work will directly address these limitations. We plan to expand the dataset in both volume and variety with community contributions, including more game genres and visual styles. A key direction is to evolve the concept of visual navigation beyond fixed STPs/MSTPs in linear paths, exploring models that can understand more dynamic navigational affordances in open-world or systemic game designs. Methodologically, we will investigate advanced architectures for improved visual understanding and temporal information processing from video data. We also intend to develop interpretability analyses for the rationale behind our model's predictions. Large-scale user studies involving real-time interaction will be crucial for validating and refining its practical utility as both a player assistant and a design aid.

Conclusion

This paper addressed the challenge of automatically identifying designer-intended visual navigation cues in complex 3D RPG environments, formalizing the problem of recognizing STPs and the critical MSTPs from visual data. We introduced a two-stage deep learning pipeline that successfully detects potential STPs and subsequently identifies the MSTP using a combination of object detection, feature fusion, parameter-efficient adapters, and optional retrieval augmentation. Our aim was not to achieve state-of-the-art on a pre-existing benchmark, but rather to define this novel problem space and demonstrate its tractability.

The initial experiments, conducted on a diverse, newly created dataset from five Action RPGs, primarily served to establish the initial feasibility of our proposed solution and to highlight key performance characteristics. These tests revealed an important trade-off: for the dense STP-detection task, full network training is effective with enough data, while parameter-efficient adapter-only transfer is more robust for data-scarce scenarios and for MSTP selection across varying data regimes. Retrieval-augmented fusion showed potential as a low-cost enhancement. A real-time pilot further confirmed in-game feasibility. The insights gained from these initial evaluations affirm that the automated recognition of such visual cues is a valuable and solvable challenge.

In summary, by defining a new problem in automated visual navigation analysis and providing an empirically validated solution, this research opens promising avenues for creating AI systems that can more deeply comprehend and interact with the visual language of game environments, ultimately enhancing both player engagement and the art and efficiency of game design.

References

Betsas, T.; Georgopoulos, A.; Doulamis, A.; and Grussenmeyer, P. 2025. Deep Learning on 3D Semantic Segmentation: A Detailed Review. *Remote Sensing*, 17(2): 298.

Bøe, R. J. 2024. *The Role of Guidance Techniques on the Player Experience in Virtual Reality Games*. Master's thesis, The University of Bergen.

Csurka, G. 2017. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*.

Dillman, K. R.; Mok, T. T. H.; Tang, A.; Oehlberg, L.; and Mitchell, A. 2018. A visual interaction cue framework from video game environments for augmented reality. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, 1–12.

Fernández-Vara, C. 2011. Game spaces speak volumes: Indexical storytelling. *Proceedings of DiGRA 2011 Conference*.

Filén, P.; and Gemal, A. 2024. Auditory and Visual Feedback: Impact on Player Performance and Comprehension in First Person Shooters. Bachelor's thesis, KTH Royal Institute of Technology.

Gibson, J. J. 2014. *The ecological approach to visual perception: classic edition*. Psychology press.

Hare, R.; and Tang, Y. 2022. Player modeling and adaptation methods within adaptive serious games. *IEEE Transactions on Computational Social Systems*, 10(4): 1939–1950.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International conference on machine learning*, 2790–2799. PMLR.

Hyde-Smith, P. 2023. *Using Object Detection to Navigate a Game Playfield*. Master's thesis, Marquette University.

Interaction Design Foundation (IxDF). 2023. What is Games User Research?

Irshad, S.; Perkis, A.; and Azam, W. 2021. Wayfinding in virtual reality serious game: An exploratory study in the context of user perceived experiences. *Applied Sciences*, 11(17): 7822.

Jung, M.; Yang, H.; and Min, K. 2021. Improving deep object detection algorithms for game scenes. *Electronics*, 10(20): 2527.

Lin, X.; Li, R.; Chen, Z.; and Xiong, J. 2024. Design Strategies for VR Science and Education Games from an Embodied Cognition Perspective: A Literature-Based Meta-Analysis. *Frontiers in Psychology*, 14: 1292110.

Liu, J.; Snodgrass, S.; Khalifa, A.; Risi, S.; Yannakakis, G. N.; and Togelius, J. 2021. Deep learning for procedural content generation. *Neural Computing and Applications*, 33(1): 19–37.

Lynch, K. 2023. The Image of the City (1960). In *Anthologie zum Städtebau. Band III: Vom Wiederaufbau nach dem Zweiten Weltkrieg bis zur zeitgenössischen Stadt*, 481–488. Gebr. Mann Verlag.

Marcus, W.; Paay, J.; Langenheim, N.; and Yang, T. 2025. HCI Methods Supporting Urban Design Evaluation Using Virtual Environments. *Interacting with Computers*, iwaf014.

Mehta, N. 2025. The Role of AI in Game Development and Player Experience. *Available at SSRN 5101269*.

Miller, D. 2024. Automated Playtest and Crash Analyzer (APCA): An AI that Playtests Video Games to Detect Bugs and Potential Crashes. *ScienceOpen Posters*.

Naying, G.; Yuexian, G.; Khalid, M. N. A.; and Iida, H. 2023. A computational game experience analysis via game refinement theory. *Telematics and Informatics Reports*, 9: 100039.

Omidshafiei, S.; Tuyls, K.; Czarnecki, W. M.; Santos, F. C.; Rowland, M.; Connor, J.; Hennes, D.; Muller, P.; Pérolat, J.; Vylder, B. D.; et al. 2020. Navigating the landscape of multiplayer games. *Nature communications*, 11(1): 5603.

Patel, V. M.; Gopalan, R.; Li, R.; and Chellappa, R. 2015. Visual domain adaptation: A survey of recent advances. *IEEE signal processing magazine*, 32(3): 53–69.

Qi, J.; and Li, H. 2023. Application of a semantic segmentation method based on deep learning in unity scene construction. In *Third International Conference on Computer Vision and Pattern Analysis (ICCPA 2023)*, volume 12754, 774–777. SPIE.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Xu, K.; and Verbrugge, C. 2025. Quantitative Analysis of Visual Guidance in Level Transitions Using Multimodal Visual Metrics. In *2025 IEEE Conference on Games (CoG)*, 1–8.

Yalçınkaya-Doma, G. 2024. *Material Matters: The Effects of Materials On Spatial Experience and Navigation in Video Games*. Master's thesis, Bahçeşehir University.

Yannakakis, G. N.; and Togelius, J. 2018. *Artificial intelligence and games*, volume 2. Springer.

Yazdani, H.; Bosaghzadeh, A.; Ebrahimpour, R.; and Dornaika, F. 2025. A Computational–Cognitive Model of Audio-Visual Attention in Dynamic Environments. *Big Data and Cognitive Computing*.