

PlayBranch：ゲーム実況動画の再生中に プレイを継承するシステム

田中 祐玖^{1,a)} 畑田 裕二^{2,b)} Hautasaari Ari^{2,c)} 苗村 健^{2,d)}

概要：プレイヤーが実況をしながらゲームをプレイする「ゲーム実況」の視聴体験は、これまで視聴者のゲーム実況への参加といった側面からさまざまに探求されてきた。これに対して本研究では、事前に撮影されたゲーム実況動画の再生中の任意の時点において、視聴者がそのゲームプレイを継承できるシステムを開発した。これは、実況動画の撮影時にプログラム内の変数の値を逐次記録したログを作成し、そのログからゲームのシーンを復元することで実現される。さらに本システムを用いて、落ちものパズル「テトリス」とレースゲーム「Sliding Penguin」のゲーム実況動画の継承体験をデモとして開発した。研究室内の学生・研究者が参加するデモ体験を行ったところ、(1) 他人のプレイを修正・書き替える欲求、(2) 実況者との競争・比較・協力、(3) 実況スタイルやゲームの特性との相性、(4) 継承体験のUI/UXに関する課題についての示唆が得られた。

1. はじめに

プレイヤーが「実況」をしながらビデオゲーム（以下「ゲーム」とする）をプレイする行為であるゲーム実況は、ニコニコ動画^{*1}や YouTube^{*2}、Twitch^{*3}をはじめとする動画投稿・ライブ配信プラットフォームの普及とともに、デジタルコンテンツの一大ジャンルとして人気を博している。例えばゲームコンテンツに特化した配信プラットフォームである Twitch では、2023 年には毎月約 700 万人の実況者が配信を行っており、1 日あたりの平均サイト訪問者数は約 3500 万人を超えている [1]。

視聴者がゲーム実況に接する形態としては、動画投稿プラットフォームにアップロードされる事前に撮影・編集された「動画」と、リアルタイムに配信される「配信」の 2 種類が存在する。いずれのコンテンツでも、「実況」が指す行為はスポーツ中継のようにその場で起きているありのままを説明・解説することのみならず、プレイヤーが自身の感想やリアクションをありのままに見せながらゲームをプレイすることや、ゲームプレイと並行して行う（ゲーム

とは無関係な）雑談なども含まれているのが、ゲーム実況というジャンルの特徴である。ジャンルが多様であるのと同じく、視聴者がゲーム実況を視聴する動機も多岐にわたる。他人のプレイから自身のゲームプレイにおける攻略のヒントを得る、ゲームを購入する前に内容を様子見するといった情報収集に加えて、実況者のリアクションを娯楽として楽しむこと、実況者とその周囲に形成されているファンコミュニティにおける交流なども、ゲーム実況コンテンツの魅力である。

ゲーム実況の視聴体験向上は、Human-Computer Interaction (HCI) 領域で注目されている研究トピックである。これまで、メタデータの可視化や感覚刺激の提示などを活用してゲーム実況の視聴体験を向上させる研究 [2], [3], [4], [5] がさまざまに行われてきた。既存の体験価値を増幅させるのみならず、視聴者がゲームプレイに介入する新たな視聴形態を提案するといったインタラクションの拡張 [6], [7] も行われている。ただし、ほとんどの先行研究はゲーム実況の中でもライブストリーミング配信に焦点を当てており、投稿された動画の視聴体験を扱った研究は少ない。さらに次節で述べるように、ゲーム実況コンテンツに対する「参加」や「委譲」といったパラダイムの探求・実装は盛んに行われている一方で、いずれのシステムにおいても実況者と視聴者の主従関係は固定的であった。

これに対して本研究では、実況動画の撮影時にプログラム内の変数の値を逐次記録したログを作成することで、動画の任意の時点で視聴者が実況者からゲームプレイを継承

¹ 東京大学大学院 学際情報学府

² 東京大学大学院 情報学環

a) ttanaka@nae-lab.org

b) hatada@nae-lab.org

c) ari@nae-lab.org

d) naemura@nae-lab.org

*1 <https://www.nicovideo.jp/>

*2 <https://www.youtube.com/>

*3 <https://www.twitch.tv/>

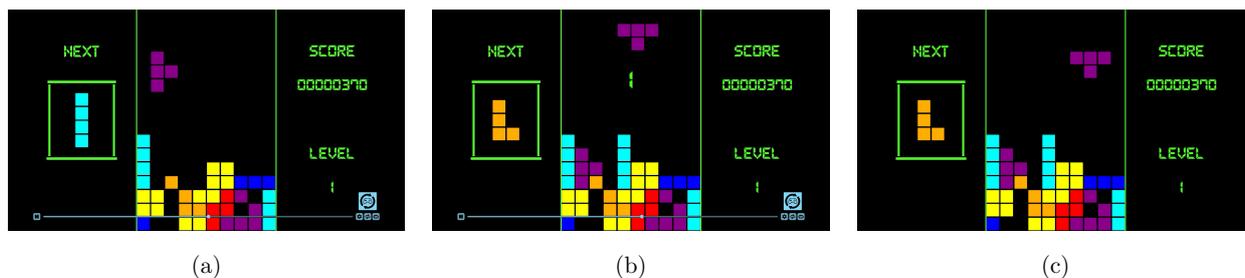


図 1: 落ちものパズル「テトリス」にゲーム実況動画を継承するシステムを実装した様子。図はそれぞれ (a) ゲーム実況動画の視聴画面と (b) 移行時のカウントダウンの画面、そして (c) 継承後の画面の様子を示す。

できるシステムを構築した。提案システムのデモ体験を通じて、ゲーム実況動画におけるプレイ継承がゲーム実況の視聴体験やゲームプレイにもたらす影響を調査する。

1.1 用語

本稿では、議論をわかりやすくするために視聴者がゲーム実況コンテンツに対して行う「参加」「移譲」「継承」という3つの行為を以下のような意味で区別する。

- **参加**：視聴者が実況者のゲームプレイに対して何らかの形で関与する行動を指す。この関与には、実況者と同じゲームセッションに直接加わる形態のみならず、視聴者がゲームに影響を与える（アイテムを出現させるなど）行為や、実況者に代わってゲーム操作の一部を担う共同行為も含まれる [6]。
- **移譲**：実況者が意図したタイミングでゲームの状態を視聴者に共有し、視聴者がゲームを操作する役割を引き継ぐことを指す。この場合共有されるゲームの状態を選択するのは実況者であり、視聴者は共有された状態から自由にゲームをプレイできる。
- **継承**：視聴者が任意のタイミングで実況者のゲームの状態を引き継ぎ、自身でゲームプレイを進められることを指す。これは視聴者が受動的に実況を観るだけではなく、能動的にゲームに介入する形態である。視聴者だったユーザーが継承したゲームを新たに実況するといった重層的なインタラクションを創出する可能性がある。

2. 関連研究

2.1 ゲーム実況視聴体験の向上

ゲーム実況体験を扱う研究では、視聴者にゲームのさまざまな情報を提示することを通じて視聴体験を向上させることに焦点が当てられてきた。例えば、ゲーム実況の視聴に関するエンゲージメントを高めるためには、コミュニケーション機能の強化やゲームのメタデータの可視化によるゲーム内容の理解促進が重要であることが知られている [2]。そうした知見を踏まえて Hammad ら [3] は、ゲー

ム実況配信において敵の情報をオーバーレイ表示したり、実況者に見えていない経路情報を視聴者が確認できるようにしたりと、視聴者の理解を補助するシステムを開発している。他にも、3D ゲームのプレイ映像からゲームのマップを再構築し、視聴者が自由にカメラを操作できるシステムも提案されている [4]。視聴者が利用できる機能を拡張する以外にも、4D モーションチェアを用いて FPS ゲームの観戦体験を拡張し、カメラの動きや発砲に連動したモーションエフェクトによって、視聴者の Enjoyment や Immersion が向上することを実証した研究もある [5]。以上の先行研究は、ゲーム実況の視聴体験が持つ既存の価値を増幅させるものであった。次節では、ゲーム実況と視聴者のインタラクションを拡張する研究について紹介する。

2.2 ゲーム実況への参加と移譲

ゲーム実況をただ視聴するのみならず、視聴者がゲーム実況に介入する手法を提案した研究もある。Lessel ら [6] は、視聴者投票でゲーム進行を決定する「Rocket Beans TV」の分析や「Twitch Plays Pokémon」における視聴者参加型の拡張インターフェースの開発を通じて、視聴者の意思決定への関与方法を検討した。特に後者では、コメントの反映方法を視聴者の多数決で決定できる仕組みを導入することで、視聴者間のインタラクションが促進されることが示唆されている。ゲーム実況への視聴者の参加が持つ設計課題は Glickman ら [7] によって整理されている。彼らは Unity と Twitch API を用いた実装実験を通じて、遅延、画面共有、注意管理、プレイヤーの主体性、実況者-視聴者関係、スケジュール変更という6つの主要な課題を指摘している。さらに Striner ら [8] は、視聴者参加のデザインスペースを体系化し、主体性 (Agency)、参加のタイミング (Pacing)、コミュニティ形成 (Community) という3つの設計概念を提示している。

他方、Google Stadia の「State Share」機能 [9] は、実況者が指定したタイミングでのゲーム状態を視聴者と共有する「移譲」体験を実現した数少ない例である。現在は利用できなくなっているものの、この機能を使って発行された

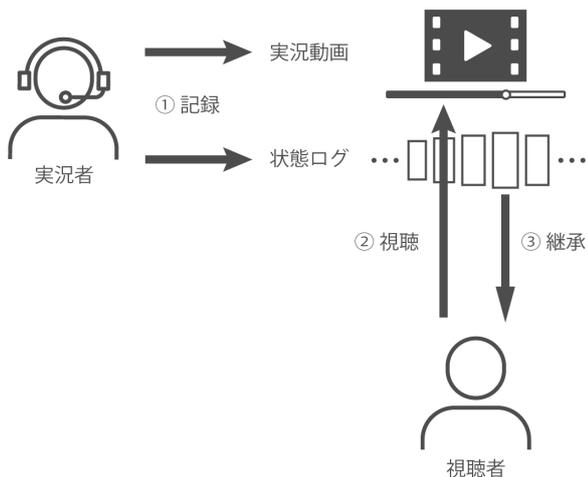


図 2: 提案するシステムの概要図. ゲーム実況者はゲーム実況動画の撮影と同時にゲームのプログラム内の変数の値を逐次記録して状態ログを作成する. 視聴者は撮影されたゲーム実況動画の視聴中にプレイを継承することができる. 継承は該当するフレームのシーンを状態ログから復元することで行われる.

リンクに視聴者がアクセスすると、クラウド上で起動したゲームで共有時のゲームの状態が復元されその続きから視聴者が遊ぶことができた。

ゲーム実況と視聴者のインタラクションを拡張する研究の多くは、リアルタイム配信における体験を対象としており、事前録画された実況動画と視聴者のインタラクションに関する議論は十分に行われているとは言えない。さらに本節で紹介した先行研究や既存のサービスでは、実況者主導のインタラクションモデルが採用されており、視聴者が主体的にゲームプレイを引き継ぐ「継承」という概念は十分に探求されていない。そこで本研究では、事前に撮影されたゲーム実況動画の再生中の任意の時点において視聴者がそのゲームプレイを継承できるシステムを開発し、ゲームプレイの継承というインタラクションが持つ価値を探索的に調査する。

3. 提案

本章では「継承プレイ」システムの主要機能である記録、視聴、継承について説明する。システムの概要を図 2 に示した。本システムは Unity 6 で実装した。

3.1 記録

本システムでは任意の時点からプレイの継承を可能にするため、ゲームのプログラム内で用いられている変数を毎フレーム記録した「状態ログ」をゲーム実況動画の撮影と同時に作成する。ゲームのプログラム内で用いられている変数には、独自に定義された変数に加えて各オブジェクトの位置やアニメーションの再生されているフレームも含ま

れる。他の変数から計算が可能である変数など、ゲームのシーンの復元に必要のない変数も存在する。ゲームプレイを再生するには、プレイヤーの入力を逐次記録する手法も考えられる。しかしこの手法ではゲームの初めから順に再現していく必要があり、ゲーム実況動画の視聴時にシーカーでの操作ができなくなるといった制約がかかる。また処理のタイミングの微細なずれにより物理演算の挙動が変わり、シミュレーションの結果が大きく異なる可能性がある。そのため本システムでは状態ログを作成する手法を選択した。

状態ログの作成には Unity Asset「Ultimate Replay 3.0^{*4}」を使用した。Ultimate Replay 3.0 はシーンの記録と再生が行えるアセットである。シーン上のオブジェクトにアセットのコンポーネントをアタッチするとそのオブジェクトの変数の値を記録できるようになる。またアセットに含まれているコンポーネントで対応できないものがある場合にはカスタムコンポーネントを作成することができる。事前に登録された Prefab の生成や破壊も記録することができる。

一例として落ちものパズル「テトリス」に本システムを実装した。テトリスは 4 つの正方形をからなる「テトリミノ」と呼ばれるブロックを落として操作し、横一列に隙間なくブロックを並べてその列を消していくゲームである。テトリスは広く知られているゲームであり、ルールがシンプルでかつ操作も簡単であるため、本システムのデモ用のゲームとして選ばれた。

テトリスのプレイ継承のために記録した変数を表 1 にまとめた。まず、各テトリミノの位置や回転と、既に落下しているかのフラグが記録される。またテトリミノを構成する 4 つのブロックそれぞれについて、その位置と既に消されていて非表示になっているかが記録される。テトリミノの生成を担うスポナーについて、次に出現するテトリミノの種類が記録される。他にも現在のスコアやゲームオーバーフラグ、ゲームオーバー UI が表示されているかが記録される。ステージのレベルはスコアからその都度自動計算されるため記録されない。記録する変数は Unity コンポーネントごとに定められている。表 1 に記載されている状態ログのデータ量が記録するデータ型に必要なデータ量よりも大きくなっているのは、各コンポーネントで記録する変数に関する情報も状態ログに出力しているためである。またテトリミノは Prefab として実装されているため、Prefab の生成や破壊に対応するための UInt16 型の ID も追加で記録されている。実際にゲームプレイを記録したところ、1 分 26 秒のゲームプレイに対して 38,557 バイトの状態ログが出力された。状態ログのファイルの大きさはゲームプレイの内容によって変動するものの、多くの場合 Full HD で同じ時間だけ撮影した動画よりも小さい。

^{*4} <https://assetstore.unity.com/packages/tools/camera/ultimate-replay-3-0-261736>

表 1: テトリスで記録する変数の一例

オブジェクト	変数	型	データ量 (byte/frame)
テトリミノ			37 (合計)
	位置・回転	Float (×6)	22
	落下フラグ	Bool	1
ブロック			30 (合計)
	位置	Float (×3)	14
	表示/非表示	Bool	1
テトリミノスポナー			11 (合計)
	次のテトリミノの種類	Int	4
スコア UI			13 (合計)
	スコア	Int	4
ゲームオーバー UI			9 (合計)
	表示/非表示	Bool	1

ゲーム画面や実況音声の撮影には「OBS (Open Broadcaster Software)^{*5}」を使用し、Full HD で撮影したものを mp4 ファイルとして出力した。現時点ではゲーム実況動画の撮影と状態ログは同期されていないため、状態ログに合わせて動画を調整する必要があった。

3.2 視聴

撮影されたゲーム実況動画は、図 1a のように本システム内で再生し視聴することができる。一時停止や再生の切り替えがボタンで操作できるほか、再生速度の調節や、シークバーを用いた再生位置の変更が可能である。

3.3 継承

視聴者はゲーム実況動画の視聴中に図 1a の右下にあるボタンを押すことでプレイを継承することができる。切り替え時に動画の再生時点のゲームのシーンが復元され、図 1c のように視聴していた状況からそのままプレイを開始できる。スムーズにプレイを開始できるように、図 1b のように切り替え直前に 3 秒のカウントダウンを表示するようにした。

4. 評価と議論

研究室内の 20 代男女 3 人と第一著者・第二著者を合わせた 5 人のグループがテトリスに本システムを実装したデモを体験し、ゲーム実況動画のプレイ継承が持つ面白さや本システムの課題などに関する議論を行った。まず第一著者が本システムを用いて作成したゲーム実況動画を他の全ての参加者が体験した後、各参加者がその場で作成したゲーム実況動画を別の参加者が継承する体験を行なった。以上の工程は約 1 時間に渡って行われた。議論で挙げられた意見は付箋にメモされ、最終的に第一著者・第二著者によって 4 つのグループに分類された。以下では、議論の中で得られた 4 つのグループである (1) 他人のプレイを修正・上

書きする欲求、(2) 実況者との競争・比較・協力、(3) 実況スタイル・ゲームの特性と相性、(4) 継承体験の UI/UX に関する課題について述べる。

4.1 他人のプレイを修正・上書きする欲求

参加者からは、実況者のミスを見かけた際に修正やリカバリーを試みたいという意欲が多く報告された。この介入欲求は、実況とは異なる選択をした場合のゲームの成り行きを見てみたいという探究的な興味にも基づいていると考えられる。また、他者が作り出した初期状態 (例えばテトリスのブロックの積み方など) に対する違和感など、実況者のプレイに対する批判も介入のモチベーションの一つとして確認された。

4.2 実況者との競争・比較・協力

実況者との競争・協力関係の形成も、継承体験の面白さとして確認された。同一条件下において実況者と自分のプレイを比較できることは、参加者に競争意識をもたらすと考えられる。特に参加者の反応から、実況者が友人など身近な存在である場合、この競争・協力要素がより強く作用する傾向が見られた。また、実際にプレイすることで実況者のスキルを体験的に理解し、再評価する機会にもなっていることも示唆された。

4.3 実況スタイルやゲームの特性との相性

実況スタイルやゲームジャンルによって、継承体験が質的に異なることも示唆された。例えば感情表現が豊かでテンションの高い実況動画の方が、淡々とした実況動画よりも、多くの参加者がプレイの継承に対してより積極的な態度を示した。一方で淡々とした実況スタイルの場合でも、実際のプレイを通じて予想以上の難しさを実感するなど、新たな気づきが得られる場面も報告された。ゲームの特性については、テトリスのような瞬間的な判断が要求されるゲームでは、継承の影響が比較的小さく感じられるのでは

*5 <https://obsproject.com/ja>

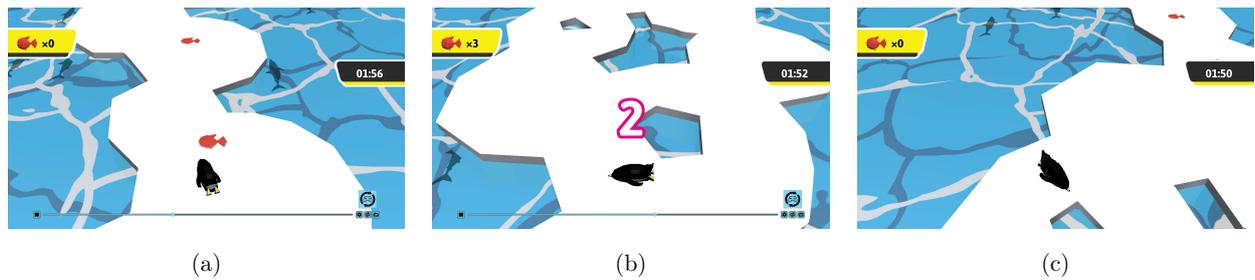


図 3: 「Open Video Game Library [10]」のレースゲーム「Sliding Penguin」にゲーム実況動画を継承するシステムを実装した様子。

ないかという意見が挙げられた。本稿の評価ではテトリスの体験しか行わなかったが、参加者の間ではタイムアタックのような競争的側面を持つレースゲームや、より長期的な思考を要するゲームの方が継承との親和性が高いのではないかという議論がなされた。

4.4 継承体験の UI/UX に関する課題

システムのユーザビリティに関する課題も指摘された。参加者からは、視聴中に即座の引き継ぎを可能にする UI やプレイの前後関係を明確に把握できるインターフェースの必要性が指摘された。また、プレイ開始後は自身のプレイに没入してしまい、実況内容との関連性が薄れやすい傾向も観察された。これは、実況内容との関連性を維持するための工夫が必要であることを示す重要な知見である。さらに、ゲームジャンルに応じた適切な表示方法の検討も必要性が指摘された。例えば、テトリスのような即時性の高いゲームでは、実況との並行表示が必ずしも効果的でない可能性が示唆されている。

5. デモンストレーション

デモでは著者らやゲーム実況動画の投稿経験のある実況者によるゲーム実況動画を継承できる。対応するゲームには前章で述べた図 1 のテトリスに加えて、図 3 のような飯田ら [10] が公開している研究や実験のためのゲームのライブラリ「Open Video Game Library」にあるレースゲーム「Sliding Penguin」がある。

6. 研究の限界

本システムは Unity Asset を使用して実装されているため、本システムの適用範囲は Unity を用いて実装されたゲームに限られる。またゲームのプログラムを変更する必要があるため、適用にはソースコードへのアクセスが必要になる。またゲームプレイ中にオブジェクトが多く生成されるものは動作が重くなったり状態ログのデータ量が大きくなったりする可能性があり、今後の改善が必要である。

7. 今後の展望

7.1 システムの発展

本システムは将来的には github などを通じて unitypackage の形式で配布される予定である。ゲーム実況動画のプレイ継承がフレームワーク化されることで、新しいゲーム実況の楽しみ方やそれに基づくエコシステムやファンコミュニティの創出が期待される。ただし、こうした将来展望を実現するためにはいくつかの技術的課題が存在する。第一に、状態ログとゲーム実況動画を適切に同期させ、それらを簡単にアップロードできる実況者向けの投稿プラットフォームが必要である。本システムの普及においては、技術的な知識が豊富でない実況者でも容易に利用できるインターフェースの実現も重要な問題である。第二に、視聴者の快適な継承体験を実現するシステム設計が必要である。具体的には、視聴中の画面からシームレスに継承プレイを開始できる機能の実装や、状態ログのロード時間の最適化が求められる。特に、ネットワーク環境による遅延の影響を最小限に抑える工夫が重要となる。

7.2 デザインスペースの探索

本システムの活用に向けて、適切なゲームジャンルやユースケースの体系的な探索が必要である。ゲームジャンルの観点では、状態管理の容易さという技術的制約と、継承プレイによって得られる体験の独自性という価値的側面を考慮する必要がある。例えば、資源管理型のシミュレーションゲームでは、実況者が構築した特徴的な状況を引き継ぐことで通常のプレイでは経験できない独特の意思決定場面が生まれる可能性がある。また、アクションゲームにおけるボス戦や高難度セクションなど、明確な課題に直面する場面での継承は、実況者の失敗を引き継いでクリアすることによる達成感や、逆に同様の失敗を経験することによる実況者への共感など、多様な心理的效果が期待できる。さらに、ゲームジャンルだけでなく、実況者や視聴者の特性がシステムの効果に与える影響についても検討が必要である。第 4 章 4.3 節でも述べたように、実況者のスタイルや視聴者が持っている視聴目的によって、継承プレイの意

義や効果が異なることが予想される。

また本システムの展開として、継承されたプレイ自体を新たなゲーム実況動画として投稿し、それをさらに別の視聴者が継承できる連鎖的な仕組みも構想している。この機能により、単一のゲームプレイに対して複数の視聴者が実況者としても参加する形態が生まれ、従来にない重層的なゲーム実況体験が創出される可能性がある。

8. まとめ

本研究ではゲーム実況動画の視聴中にプレイを継承できるシステムを提案した。このシステムは実況者のプレイを受け継いで視聴者がプレイできるインタラクティブな仕組みを提供することで、ゲーム実況動画と視聴者のインタラクションを拡張する。今後は、本システムのブラッシュアップに加えて、ゲームプレイの継承体験が持つ価値や本システムが可能にするインタラクションの探索を行っていく。

謝辞 本研究は中山未来ファクトリーの支援を受けて行われた。

参考文献

- [1] 日経クロストrend: 毎月約700万人が配信 Twitch、CCO が語るライブ配信市場の進化, <https://xtrend.nikkei.com/atcl/contents/watch/00013/02343/>. Accessed 2024-12-22.
- [2] Kriglstein, S., Wallner, G., Charleer, S., Gerling, K., Mirza-Babaei, P., Schirra, S. and Tscheligi, M.: Be Part Of It: Spectator Experience in Gaming and Esports, *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI EA '20, New York, NY, USA, Association for Computing Machinery, pp. 1–7 (online), DOI: 10/g8d9pd (2020).
- [3] Hammad, N., Harpstead, E. and Hammer, J.: GameAware Streaming Interfaces, *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY Companion '23, New York, NY, USA, Association for Computing Machinery, pp. 248–253 (online), DOI: 10/g8pt5r (2023).
- [4] Hartmann, J. and Vogel, D.: Enhanced Videogame Livestreaming by Reconstructing an Interactive 3D Game View for Spectators, *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, Association for Computing Machinery, pp. 1–14 (online), DOI: 10/g8d9ph (2022).
- [5] Yun, G., Lee, H., Han, S. and Choi, S.: Improving Viewing Experiences of First-Person Shooter Gameplays with Automatically-Generated Motion Effects, *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, Association for Computing Machinery, pp. 1–14 (online), DOI: 10/gksk4g (2021).
- [6] Lessel, P., Mauderer, M., Wolff, C. and Krüger, A.: Let's Play My Way: Investigating Audience Influence in User-Generated Gaming Live-Streams, *Proceedings of the 2017 ACM International Conference on Interactive Experiences for TV and Online Video*, TVX '17, New York, NY, USA, Association for Computing Machinery, pp. 51–63 (online), DOI: 10/gt7ff3 (2017).
- [7] Glickman, S., McKenzie, N., Seering, J., Moeller, R. and Hammer, J.: Design Challenges for Livestreamed Audience Participation Games, *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '18, New York, NY, USA, Association for Computing Machinery, pp. 187–199 (online), DOI: 10.1145/3242671.3242708 (2018).
- [8] Striner, A., Webb, A. M., Hammer, J. and Cook, A.: Mapping Design Spaces for Audience Participation in Game Live Streaming, *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, Association for Computing Machinery, pp. 1–15 (online), DOI: 10/gksmcv (2021).
- [9] Google for Developers: Google GDC 2019 Gaming Announcement, <https://www.youtube.com/watch?v=w7hc4R8JAJY> (2019). Accessed 2024-12-16.
- [10] Iida, K., Ina, Y., Hayashi, D., Yanase, Y. and Watanabe, K.: Open Video Game Library: Developing a Video Game Database for Use in Research and Experimentation, *Proceedings of the 29th ACM Symposium on Virtual Reality Software and Technology*, VRST '23, New York, NY, USA, Association for Computing Machinery, pp. 1–2 (online), DOI: 10/g8wng3 (2023).