

# UE4 による VR で使用するペイントツールの実装と応用

## Mounting and Application of Painting Tool Used in VR Using UE4

○上原諒祐<sup>\*1</sup>, 三浦研<sup>\*2</sup>, 安田溪<sup>\*3</sup>

Ryosuke Uehara<sup>\*1</sup>, Ken Miura<sup>\*2</sup>, Kei Yasuda<sup>\*3</sup>

\*1 京都大学工学部建築学科 学部

Student, Graduate School of Engineering, Kyoto University

\*2 京都大学工学研究科建築学専攻 教授・博士 (工学)

Professor, Graduate School of Engineering, Kyoto University, Dr. Eng.

\*3 京都大学工学研究科建築学専攻 研究員・修士 (工学)

Researcher, Graduate School of Engineering, Kyoto University, M. Eng

キーワード：ペイントツール; VR; Unreal Engine 4; 実験手法  
Keywords: Paint Tool; VR; Unreal Engine 4; Experimental Approach.

### 1. 開発背景と目的

近年の VR 技術の進歩や Unity や Unreal Engine 4 などのゲームエンジンの普及により、VR は設計実務における合意形成だけでなく建築計画学の研究においても用いられるようになった。

建築計画学の実験では、被験者が空間内で何らかの活動をしている間にその様子を記録したり、アンケートによる印象や機能などの評価をしたりすることが多く、中には空間を体験している間に思ったことやマップなどを被験者に書いてもらうというような実験も行われる。そのような手法は、VR と実空間の間での空間認知の差異の問題が解決されれば、VR を用いた実験においても有効である。VR で空間体験をしている間に被験者が何らかの記述を行うには、任意の空間を VR で体験しているとき、仮想的に手に持っているキャンバスやノートに任意の記述を行い、その内容を画像として保存するというペイントツールのような機能が必要であり、本研究においてはその機能を Unreal Engine 4 を用いて実装した。

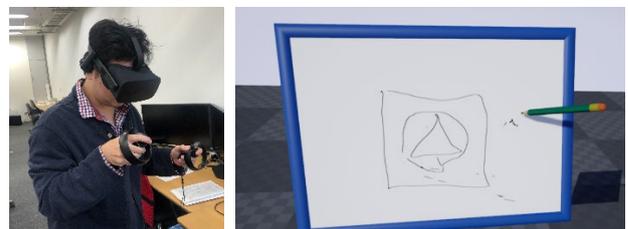
### 2. Unreal Engine 4

株式会社 Epic Games が無料で公開しているゲームエンジンであり、UE4 と略される。様々な機能を C++ のみではなく、ビジュアルスクリプティングツールである BluePrint (以降 BP と記述する) を用いて実装することができることが特徴である。本研究において実装する機能も BP のみを用いて実装されたものであり、基本的な UE4 の使用方法を理解していれば容易に実装が可能である。なお、本研究において使用された UE4 のバージョンは 4.22.3 である。

### 3. ペイントツールについて

#### 3.1. ペイントツールの仕様

本研究において、筆者らは以下に記述する仕様のペイントツールを実装した。VR 機材はモーションコントローラーが付属するものであれば任意のものを使用することができ、本研究においては VR 機材として Oculus Rift、Oculus Touch を使用する。VR アプリケーション内において、利用者は仮想的に左手に A3 サイズのキャンバスを持ち、右手に消しゴム付きの鉛筆を持つ。鉛筆の先がキャンバスに触れているとき、触れている場所に黒い線を描く。右手の Oculus Touch の B ボタンを押した状態を保っている間は持っている鉛筆を逆さに持ち消しゴムとして利用し、B ボタンを離したら鉛筆に持ち直す。アプリケーションを終了したときに描かれたスケッチを画像ファイルとして保存する。以下にペイントツールの利用の様子を図として示す。(図 1、図 2)



左：図 1. VR でスケッチをする様子

右：図 2. VR 中のスケッチの内容

#### 3.2. ペイントツールの建築計画学の研究への応用例

VR 空間でなく実存の都市空間において、被験者にオノマトペを記述させながら都市を歩いてもらうという歩行実験が行われた。<sup>1)</sup> この手法は実空間において行われたも

のであるが、CG で再現された空間において VR を用いてこの手法の実験を行えば、任意の条件のみを変更した複数の空間による対照実験を行うことが可能である。その際、被験者はオノマトペの記述を VR の使用中に行われなければならない、そのためには本研究で実装したペイントツールのような機能が必要となる。

#### 4. UE4 における Canvas Render Target オブジェクト及び諸関数

Canvas Render Target とは UE4 のテクスチャのオブジェクトの一種であり、本研究のようにゲームプレイ中にテクスチャ編集をする用途に利用することができる。BP や C++にて、テクスチャパラメータが使用された Dynamic Material Instance を使い、Set Texture Parameter Value 関数の引数に Canvas Render Target オブジェクトを代入することで、本研究のような 3D モデルの表面に絵を描くなどの操作が可能となる。

以下に 3D モデル表面に線を描き、描いた線を消すことができるスクリプトの内容を示す。

Tick イベントから Line Trace By Channel 関数や Line Trace For Object 関数などを呼び出し、以下のようなスクリプトを組む。(図 5)



図 5. LineTrace By Channel 及び Find Collision UV

Line Trace がヒットするオブジェクトは描画したい対象に限定する必要があり、そのためには線の描画をする

3D モデルに専用のオブジェクトチャンネルなどを用意する必要がある。

ヒットしたコリジョンが Complex Collision の場合、返り値の Out Hit ピンから Find Collision UV 関数でヒットした場所の 3D モデルにおける UV 座標を取得する。この座標を用いて次のスクリプトを組む。(図 6)

Begin Draw Canvas to Render Target 関数の返り値の Size から出力されたテクスチャのサイズに先に取得した UV 座標を乗算することで、テクスチャのどこの座標を編集するのかを指定することができる。また、同関数の返り値の Canvas から Draw Line 関数を呼び出すことで、1 フレーム前にヒットした UV 座標を記憶し、その座標と現在のフレームにヒットした UV 座標をそれぞれテクスチャサイズに乗算した値を線分の始点と終点にすれば、UV 座標に対応した 3D モデルの表面の位置に線を描くことが可能である。描画を終了するためには End Draw Canvas to Render Target 関数を使用する。

描画された Canvas Render Target を画像として保存するには Export Render Target 関数を使う。画像フォーマットは hdr であり、引数の File Name には、「.hdr」の拡張子を記載する必要がある。

#### 5. Canvas Render Target の応用

Canvas Render Target は本研究と同じ要領で他にも様々な応用が可能である。例えば、今日のアイトラッキング機能が搭載された VR を使用すれば、UE4 で両目の視線の始点と方向を取得することができ、この値を用いて人間が物体のどこに注視したかを物体にそのまま描画することも可能である。

#### 参考文献

- 1) Yusuke Kita et al. : Depiction and Analysis of Urban Modality Based on City Walks Using Onomatopoeia (Transactions of AIJ), Vol 83 No. 749, pp.1285-1295, 2018.7. (in Japanese) 北雄介・門内輝行：オノマトペを用いた街歩きによる都市の様相の記述と分析、日本建築学会計画系論文集, Vol 83, No.749, pp.1285-1295, 2018.7.

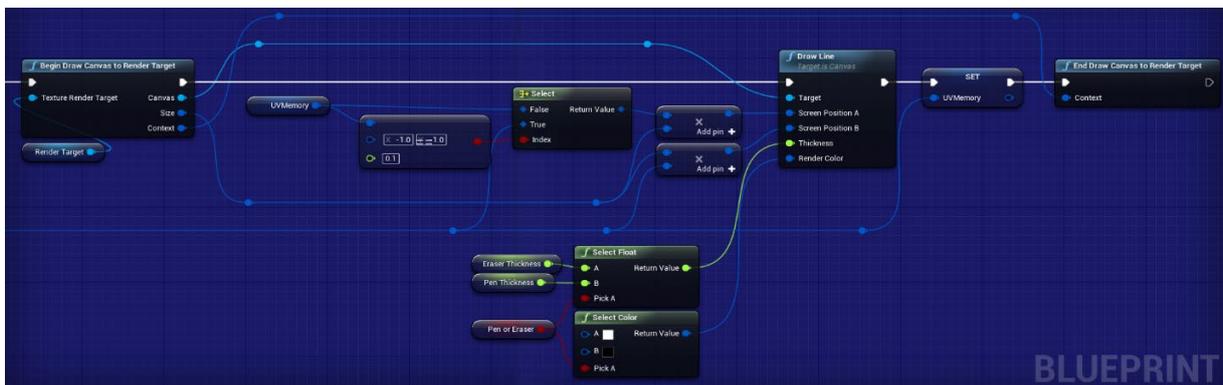


図 6. キャンバスへの線の描画の処理、Tick イベントによって呼び出される。