

VR 環境内における VR アプリケーション開発支援の検討

塩澤 秀和[†] 渡邊 海[†] 丸茂 棕一[†]

概要：VR アプリケーションの開発では、特にその終盤にかけて HMD の着脱を頻繁に繰り返さなければならず、従来のソフトウェア開発にはなかった大きな負担となっている。そこで、複数の開発者が HMD を装着したままの状態、VR 空間内のオブジェクトを視覚的に編集し、さらにテキストベースのコーディングもできるような共同開発支援システムの実現方法を検討する。

1. 背景と目的

近年、HMD (Head Mounted Display, VR ゴーグル) の普及にともない、企業や個人による VR (Virtual Reality) アプリケーション開発が増加している。VR アプリケーションの開発では、一般的に開発用の PC と実行時に用いる HMD を使用する。開発者は、ほとんどの時間、PC の画面に表示されたゲームエンジン等の開発用ソフトウェアを用いて、テキストベースの言語で開発を進めるが、VR アプリケーションが完成に近づくにつれて、HMD を装着して実機で動作を確認する必要のある作業が増えていく。

一般に HMD は頭部に固定する必要があり着脱に手間がかかるので、HMD の着脱を繰り返さなければならないのは、開発者にとって大きな負担である[1]。特に、VR アプリケーション開発の終盤では、実機における動作確認と表示の調整のために、HMD (場合によっては複数の機種) を頻繁に着脱する必要がなる。これは、開発でも製品でも同様な平面ディスプレイを開発に用いるビデオゲームなどの従来のソフトウェア開発にはなかった負担である。

また、アプリケーションの実行時のエラー等の情報は、通例、PC 上の開発用ソフトウェアの通知欄に表示されるので、開発者はエラーが起きるたびに HMD を外して PC の画面を確認することになる。そのため、VR アプリケーションのデバッグは従来のソフトウェアに比べて非常に手間がかかると言われており、デバッグ用のログ文字列を VR 空間内に表示するテクニックが用いられることもある。

本研究では、以上のような問題意識から、VR アプリケーションの開発において、開発者が HMD を装着したままの状態、VR 空間内のオブジェクトを視覚的に編集し、さらにテキストベースのコーディングもできるような開発支援システムの実現を目指す。これによって、特に開発の終盤における開発者の HMD の着脱回数を減らし、開発作業の負担を軽減できると考えている。

さらに、このように開発者が開発中の VR 環境内に入ることの帰結として、プロジェクトに関わる複数の開発者が、作業空間である開発中の VR アプリケーションの環境内に入ることになるので、VR 環境内における VR アプリケーションの共同開発支援の実現も視野に入れる。

2. 関連技術等

ゲームエンジン Unreal Engine 4 の VR モード[2]は、まさに本研究と同様の問題意識から開発が始まった[1]もので、開発者は VR 環境内で、オブジェクトを設置、複製、削除、編集して、VR 世界を構築することができる。しかし、現在の VR モードは、まだオブジェクトの設置と編集を可能にした段階で、コードデバッグの支援機能や複数の開発者による共同開発を想定した機能はない。

VR ライブサービス「バーチャルキャスト」[3]では、ユーザはライブ配信用の VR 環境内でプログラミングを行うことで、実際のライブ配信で利用可能な簡単なオブジェクトを作成することができる。このサービスでは、VR 環境内にバーチャルな PC が表示されるので、アバターをその前に置いて、デスクトップ画面を見ながらタッチタイピングによって文字を入力してコーディングを行う。

VR 環境内でテキストコーディングを実現したものとしては、RiftSketch[4]がある。ユーザは、VR 環境内に表示されたテキストエディタの画面とバーチャルなキーボードを用いて、動的かつリアルタイムに 3D オブジェクトを生成することができる。キーボードは、現実世界のキーボードを撮影した映像が表示される。しかし、このシステムでは、開発者は VR 空間内で移動ができないため、空間全体を見渡すことが困難であり、世界に入り込むような VR アプリケーションの開発には適さない。

VR 環境内で共同プログラミングを実現する技術としては、VR サービス Neos VR で開発中の機能である LogiX[5]が存在する。これは、3D のビジュアルプログラミング言語であり、VR 空間内で複数人が同じコードを編集することができる。この例は、VR 環境内における共同開発支援が開発者のコミュニケーションの円滑化に効果的であることを示唆している。

単に HMD を用いた開発とデバッグを円滑化する技術としては、PC 用の VR をスタンドアロン型の HMD を通して遠隔プレイ可能にする ALVR (Air Light VR) [6]を用いる手法もある。ALVR では遠隔プレイだけでなく PC の開発用ソフトウェアの画面も HMD に転送することができる。ただし、コーディングやテストプレイのために開発用ソフトウェアの全体を表示させる場合は、VR の 3D 空間内に、PC 画面を 2D 画像として表示することになる。

[†] 玉川大学工学部ソフトウェアサイエンス学科
shiozawa@eng.tamagawa.ac.jp

3. 本研究の提案

3.1 本研究の方向性

我々は、高度な VR アプリケーションの開発では、将来的にも自由度の高いテキストベースのプログラミング言語の使用が主流であると想定した。また、通常のディスプレイと HMD を使用したときの疲労度等を考慮すると、長時間表示装置を見続けて作業しなければならない開発者は、両眼視差による 3D 表示が特に必要な場面を除き、基本的に平面ディスプレイによる表示を好むと考えた。

よって、本研究では、VR アプリケーション開発のすべての工程を VR 環境内だけで行うことは目的とせず、今後も VR アプリケーションの開発は、序盤から中盤では従来型の 2D のユーザインタフェースを主に用い、終盤にかけて HMD による 3D 表示を多用すると想定して、従来型の開発環境と調和する手法を実現したいと考えている。

3.2 VR オブジェクトの視覚的な編集の実現方法

本研究では、VR 空間に存在するオブジェクトの情報を、開発者が視覚的に編集できるようにするため、位置や角度などのプロパティ（属性）と動作のスキプト（プログラム）に分類し、それらの存在を付箋のような板状のオブジェクト（「タグ」と呼ぶ）で表すことを考えている。

そして、タグを図 1 のように VR オブジェクトに付加するように表示する。開発者は一人称視点の VR 空間内でこれらのタグを選択して各種の操作することで、対応するオブジェクトの情報を参照したり、視覚的に編集したりすることを可能にしたいと考えている。

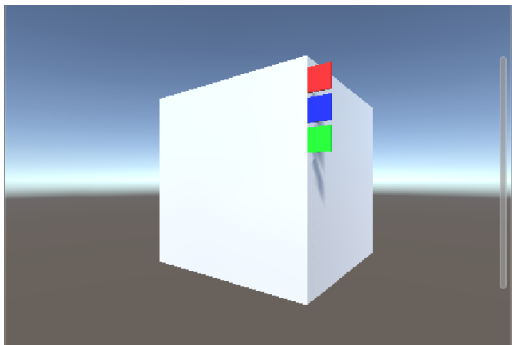


図 1 VR オブジェクトへのタグ付け

また、VR アプリケーションのデバッグ時には、実行中のスキプトに対応するタグを点滅させるなど、その表示をダイナミックに変化させることにより、開発者が HMD を着用して VR 表示だけを見ている場合でも、視覚的にプログラムの流れを理解できるようにする。

3.3 VR 環境内でのコーディングの実現方法

VR 環境内でテキストベースのコーディングを実現するために、本研究では、VR 空間内にコーディング用のテキストエディタを表示することを考えている。これは、図 2 に示すように、VR 空間全体に対するものと個別に VR オブジェクトに対するものを想定している。個別のオブジェクト

に対するテキストエディタは、対象のオブジェクトのタグを選択している場合にだけ表示され、対応するスキプトのソースコードを編集可能にする。

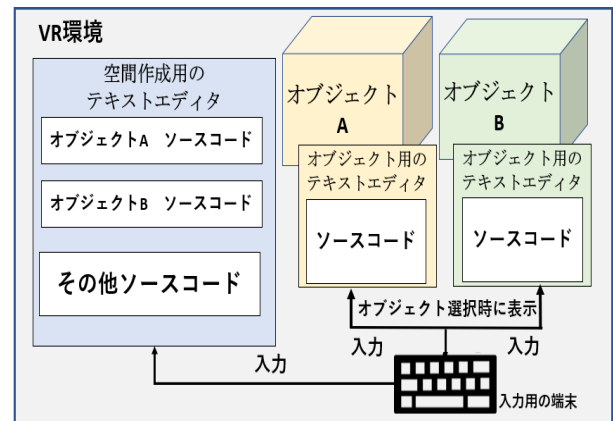


図 2 VR 環境内でのコーディング支援

この際、大きな問題となってくるのが文字入力の方法である。本研究では、入力効率の観点から現実世界の物理的なキーボードを使用したいと考えているが、実際にどのようなインターフェースを採用するかは検討段階である。

さらに、VR 環境内で共同コーディングをサポートするためには、複数の PC でソースコードを共有し、共同レビューや編集権の排他制御を実現する必要がある。この際、それぞれの開発者が開いた VR オブジェクトのテキストエディタをどのように表示するか検討している。

4. 評価方法

本研究によるシステムが実際の VR 開発に効果的であるかどうかを評価するため、VR アプリケーション開発の経験がある学生を対象とした評価実験の実施を検討している。

具体的には、本研究手法を採用しない場合と採用した場合で、基本的な VR アプリケーションの開発にかかった時間および開発の工程ごとの HMD の着脱回数を計測することや、作業の負担軽減に効果的であったかどうかを尋ねるアンケートを実施する計画である。

参考文献

- [1] 喫茶板東: Unreal Engine 4 は、VR ゲームへの対応にも超積極的だった！【CEDEC 2016】、ファミ通.com, 2016. <https://www.famitsu.com/news/201608/30114482.html>
- [2] Epic Games, アンリアルエンジン VR モード, <https://docs.unrealengine.com/ja/Engine/Editor/VR/index.html>
- [3] バーチャルキャスト, <https://virtualcast.jp>
- [4] Brian. Peiris: RiftSketch, 2014. <https://github.com/brianpeiris/RiftSketch>
- [5] Tomáš Mariančík: Introducing LogiX - VR multiplayer visual programming system LogiX, 2017. <https://neovr.com>
- [6] polygraphene: ALVR - Air Light VR. <https://github.com/polygraphene/ALVR/>