

17. Development of Simplified AR Library for Android

Android 向け簡易版 AR ライブラリの開発

Yuu TAMURA 田村 優
Supervised by Kei KOGAI 小飼 敬

Abstract — The idea of Augmented Reality (AR) was proposed already several decades ago. However, AR has become much familiar by the appearance of smartphones but it is hard to say that AR has become widespread around the world. Although there are various causes, I thought that one of that is “difficult to develop them”. So I thought that I could assist the first step of AR development by making a Library for those who learn AR for the first time.

1 はじめに

2016 年, 「VR 元年」と呼ばれたこの年は, VR の発展が目覚ましく世間の認知度も以前に比べて格段に上がった. VR, すなわち仮想現実感とは視界を完全に遮って完全な仮想空間の視界を提供するが, 似たようなものに AR, 拡張現実感がある. 拡張現実感とは仮想現実感と違い視界を完全には遮らず, 実際の視界に新たな情報を組み込みこんだような視界を提供する技術である. 視界を完全に遮るため屋外での使用に限界がある仮想現実感に比べ, 拡張現実感とは視界を遮らないために幅広い使い方が可能であり, 今後様々なシーンで拡張現実感が活用されるだろうと予想できる. しかしこのような予想は数十年前からされており, 実現に向けて様々な研究が行われてきた今でも世界に拡張現実感が浸透していると言われると肯定はできないだろう. 拡張現実感を一気に身近な物にしたスマートフォンでさえ火付け役になれなかった. このような実態を考えた時, まず挙げられる原因は「開発の難しさ」であると考えた. 拡張現実感を実装するためには膨大な知識と時間が必要であり, それが開発者を拡張現実感から遠のかせていたと考えられる. しかし近年 ARToolKit [1] などのライブラリが公開され以前に比べ開発が容易になっている. そこで今回, AR アプリケーションの開発の最初のハードルをできるだけ下げようと ARToolKit をさらに簡単に使用できるように改良することにした. デバイスは先ほど述べたように拡張現実感との相性がとても良く, 開発者が多い Android を選択した.

2 現状

2.1 開発環境

近年の Android 開発環境は以前まで使用されていた Eclipse から AndroidStudio [2] へと移り変わっている.

2.2 ライブラリ

主に以下のライブラリが提供されている.

- ARToolKit
 - OS X, Windows, Linux, iOS, Android など, 様々なプラットフォームで提供されている.
- NyARToolKit
 - Java, C#, ActionScript, Processing, Unity, C++, Android, Windows など, 様々なプラットフォームで提供されている.
- AndAR
 - 近年登場した Android 専用の AR ライブラリ. プラットフォーム上での AR を最小限の労力で実現することを目的に作成されている.

3 設計

3.1 環境

現状を踏まえて, 以下の環境で開発することにした.

- OS : Mac OS X Yosemite
 - 動作の安定性, 今後 iOS 版の開発をする可能性などを考慮し, Mac mini を使用した.
- 開発環境 : AndroidStudio 2.2.2
 - 近年の開発環境の遷移を踏まえて, これからのスタンダードとなるであろう AndroidStudio を選択した.
- 使用ライブラリ : ARToolKit version: 5.3.1
 - Android の最新 version に対応している他, 様々なプラットフォームに対応しているため今後の展開を考えて ARToolKit を選択した.
- 実行端末 : Nexus9
 - Android 7.0 が動作する Nexus9 を AR アプリケーションの実行端末として使用した.

3.2 コンセプト

以下の特徴を持つライブラリを作成する.

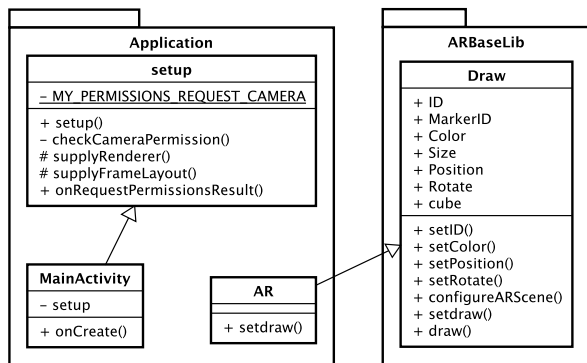


Fig.1: Class diagram

- 関数名や変数名などをできるだけ分かり易い単語で記述する。
- クラスを Main, Setup, AR の三つで構成し、各クラスの役割を明瞭化する。
- 拡張し易い構成にする。
- MainActivity^{*1}はできるだけ最小限にする

4 実装

本研究で実装したクラスのクラス図を Fig.1 に示す。

4.1 App モジュール

4.1.1 MainActivity

MainActivity クラスは親クラスを変更し、Setup を呼び出すだけとなっている。

4.1.2 Setup

Setup クラスは提供されるライブラリ内ではできない各種設定をするクラスである。難しい設定などを全て集めている為、ライブラリと共に提供されるクラスである。使用者は提供された Setup クラスを MainActivity と同じパッケージに入れるだけで実装が完了する。

4.1.3 AR

AR クラスは実際に AR で表示したい物体の設定をするクラスである。簡易化を図る為、表示できるのは Box のみである。Box の大きさ、色、位置、角度、対応する AR マーカを登録するクラスである。

4.2 ライブラリ

4.2.1 Draw

Draw クラス内に AR マーカの検出、及び Box の表示をする処理が入っている。複数のマーカに反応させる場合、ここの処理を変更しないといけない。

4.2.2 ARSimpleApplication

ARToolkit ライブラリで App モジュールに追加しないといけないクラスだったが、処理の明瞭化を図る為ライブラリ内に移動した。

^{*1} Android アプリケーションを実行した時、一番最初に実行されるメソッド

5 評価

5.1 達成項目

- 関数名や変数名は分かりやすくなったと言える。
- クラスをコンセプトの通りに構成し明瞭化されたと言える。
- 目標の一つである MainActivity の最小限化は達成したと言える。

5.2 未達成項目

- 拡張性は十分でなく、複数のマーカに反応させる為にはライブラリ内を変更しないといけない。

拡張性に関してはライブラリ内を変更しないといけない為に Android の知識が必要である。Android の開発をした事がある人であれば、拡張は難しくは無いだろう。しかし今回は「簡易化」が目的なので未達成と判断した。

6 考察

コンセプトの一つとして「拡張し易い構造」があったが、現時点では拡張性は低く達成できたとは言えない。しかし、Box を表示する事に限って言えば Android 開発をした事が無い人でも以前より簡単に AR を体験する事ができるようになったと言える。今後、このライブラリを更に改良して拡張性を持たせる事ができれば、Android で AR を体験する為のライブラリとして有用になると考える。また、表示した後の操作をサポートすることができれば更に可能性が広がるだろう。

7 おわりに

Android アプリケーション開発初学者でも、比較的簡単に AR を体験する事ができるようになったと考える。ディスプレイを見ながらカメラを動かすのではなく、カメラの視点と自分の視点がほぼ同じで直感的な操作ができる点で Android はとても有用であり、今後の発展も望めるだろう。しかし今のままではただ箱を表示するだけの機能しかなく、多彩な処理をしたい場合には向いて無い。今後は 3D オブジェクトの表示、Raspberry Pi など AR アプリケーションの出力先としての装置との連携など、更に AR の入門用ライブラリとしての十分な機能を実装したい。

参考文献

- [1] ARToolkit - Android Documentaion
http://artoolkit.org/documentation/doku.php?id=4_Android:android.about
- [2] Android Developers Site - Android Studio
<https://developer.android.com/studio/intro/index.html>