

VRコンテンツの つくりかた 2



れいぞうこのドア

VR コンテンツのつくりかた 2

Daydream 特集

youten 著

2018-05-05 版 発行

目次

第 1 章	はじめに	5
1.1	Mirage Solo 対応について（2018.05）	5
1.2	賞味期限とリポジトリ公開について	6
第 2 章	Daydream を知る	7
2.1	Daydream とは	7
2.2	Cardboard と Daydream の歴史的経緯	15
2.3	Daydream standalone	16
2.4	AR と ARCore とポジショントラッキング	18
第 3 章	Daydream を使う	20
3.1	Daydream のセットアップ	20
3.2	デベロッパー向け設定と VR Entry Flow	23
3.3	Daydream View 互換機 BOBOVR Z5 を試す	29
第 4 章	Unity で Daydream アプリ開発の入り口を覗く	31
4.1	Hello, Daydream	31
4.2	Instant Preview	32
4.3	Daydream Elements	35
4.4	VR Video と Daydream Media App Template アプリ	37
4.5	Resonance Audio と Audio Factory アプリ	40
4.6	6DoF モーショントラッキングシステム NOLO を試す	42
第 5 章	Hello, Mirage Solo	48
5.1	本章について	48
5.2	Mirage Solo とは	48
5.3	セットアップ	50

第 1 章

はじめに

半年前にはもうそろそろ XR 元年がバズっていると考えていたのですが、よく言われる通り VR のハイプ・サイクルが少しフェーズが進んだところまでいってしまって落ち着いていたり、意外と AR が単独で盛り上がっており、iPhone のほんわかとした影響力を感じる 2017 年をお過ごしのみなさま、いかがおすごしでしょうか。筆者の youten と申します。

本書は、以下のような方をターゲットにしています。

- もうそろそろ日本でもデビューしてメジャー街道まっしぐらが約束されているはずの Daydream のことが知りたい
- Daydream View を買わずになんとか Daydream を試す方法が知りたい
- VR や Android や Unity やプログラミングについてちょっとは知っていて、Daydream アプリを開発してみたい

1.1 Mirage Solo 対応について（2018.05）

Daydream Standalone である Mirage Solo が発売となりましたので、それにあわせて一部のコンテンツを修正し、Mirage Solo の章を追記しております。セットアップは開発者向けオプションなど一部 2017.10 当時の記載が残っておりますが、適時読み替えて確認願います。

1.2 賞味期限とリポジトリ公開について

なお、本書は記載内容の賞味期限が切れてしまうことへのアップデート対応、あまり紙メディアが向いていない環境等のバージョンアップに追従するため、初版以降の PDF と関連ファイル一式を全て以下のリポジトリで公開しております。あらかじめご了承ください。

- <https://github.com/youten/howto-create-vr-contents2>
 - Re:VIEW の素材一式、出力 PDF、関連プロジェクトのソースコードを全て含む想定です。

本書のうち、私 youten が著作権を有する範囲のライセンスについては、文章は CC-BY 4.0 ライセンス^{*1}、ソースコードについては Apache License v2^{*2}を適用します。

本書籍は Re:VIEW で作成されており、その設定ファイル等について、MIT ライセンス^{*3}に基づき「C89 初めての Re:VIEW v2」リポジトリ^{*4}で公開されているものを利用させていただいております。

表紙他、高崎柚乃モデル^{*5}を CC-BY 4.0 ライセンス^{*6}に基づき利用させていただいております。

^{*1} <https://creativecommons.org/licenses/by/4.0/legalcode.ja>

^{*2} <http://www.apache.org/licenses/LICENSE-2.0>

^{*3} <https://opensource.org/licenses/MIT>

^{*4} <https://github.com/TechBooster/C89-FirstStepReVIEW-v2>

^{*5} © Gugenka ® from CS-REPORTERS.INC/YUNO <https://gugenka.jp/original/yno-3d.php>

^{*6} <https://creativecommons.org/licenses/by/4.0/legalcode.ja>

第 2 章

Daydream を知る



Daydream

▲図 2.1 Daydream ロゴ

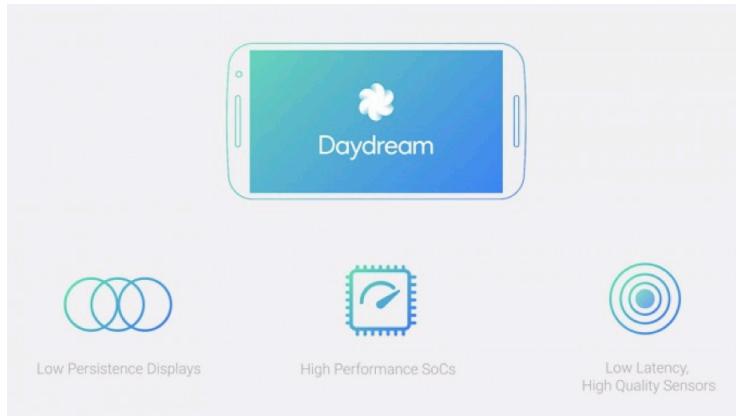
2.1 Daydream とは

Daydream とは、Google の Android スマートフォン向け VR プラットフォームです。ハードウェアからアプリマーケットまでを含む統合プラットフォームであり、主に以下に示す要素から構成されています。

- Daydream-ready と呼ばれる高スペックの Android デバイス
- ジャイロ・加速度センサを搭載し、Bluetooth で接続するコントローラとセットになったヘッドユニットの Daydream View
- Android 7.0 (Nougat) 以降での、低い入力遅延・高速な描画処理のための Android OS 内部での特別な対応
- Google Play とシームレスに融合し、HMD を装着したままでアプリのインストールから起動を行えるランチャー機能をもった専用のホームアプリ

それぞれの要素について、詳細を示します。

Daydream-ready phones



▲図 2.2 Daydream-ready phones の 3 要件

その母艦となるハードウェア要件があり、それを満たしたものを Daydream-ready phones^{*1}と呼びます。SoC、センサ、ディスプレイの3つの要件があります。

- 3D 画像処理と FPS を維持するための高いパフォーマンスを有する SoC
- 姿勢予測のための高い分解能と低遅延のセンサ
- 高い応答速度、低遅延のディスプレイ

SoCについては、現状 Daydream-ready phones に認定されている Android デバイスから判断すると Snapdragon 820 以上であれば満たすと考えられます。ただし、「MediaTek が Daydream のための SoC を開発する」というニュース^{*2}も発表されており、Snapdragon に限定はされていない模様です。

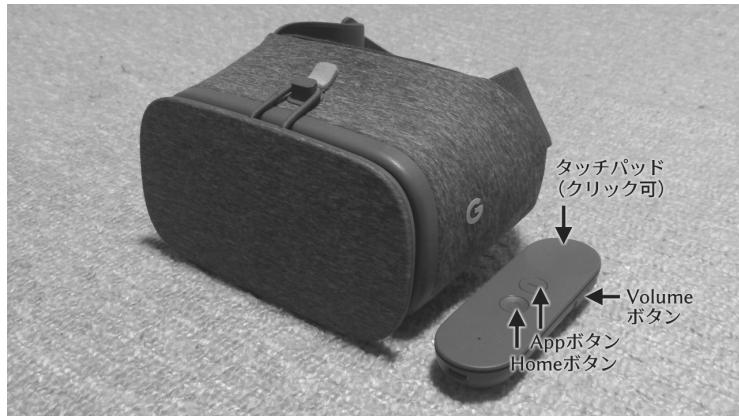
ディスプレイについては、応答性能の観点で有機 EL が主流ですが、応答性能の高い液晶も追って開発されました。「JDI の開発した VR 専用の超高精細・高速応答 液晶

*1 Daydream-ready phones <https://vr.google.com/daydream/phones/>

*2 MediaTek's Next-Gen SoCs To Support Google's Daydream | Androidheadlines.com [https://www.androidheadlines.com/2016/05/MEDIATEKS-NEXT-GEN-SOCs-SUPPORT-GOOGLES-DAYDREAM.html](https://www.androidheadlines.com/2016/05 MEDIATEKS-NEXT-GEN-SOCs-SUPPORT-GOOGLES-DAYDREAM.html)

ディスプレイ^{*3}」のように、同等の性能があれば液晶ディスプレイでも要件を満たします。Lenovo 社の一体型 Daydream 機である Mirage Solo は LCD を搭載しています^{*4}。将来は LCD を搭載した Daydream-ready phones も増えてくるものと思われます。

Daydream View とコントローラ



▲図 2.3 Daydream View（旧型）本体と専用コントローラ

専用コントローラ

Daydream の VR アプリを体験するには、レンズ付きヘッドユニットがセットになった Daydream View（図 2.3）本体と、Bluetooth 接続の専用コントローラが必要です。

このレンズつきヘッドユニット部ですが、特別な回路は搭載されていないため、レンズ部分が同等であれば、実は別の Cardboard ゴーグルを利用して動作させることができます。Daydream View には NFC タグが埋め込まれています。以前は NFC タグの読み込みが初期セットアップの途中で必要だったのですが、現在は不要になっています。

このコントローラは IMU センサを含み、クリックのできるタッチパッド、App ボタン、Daydream ボタン（旧 Home ボタン）とボリュームボタンを持ちます。また、Bluetooth で母艦である Daydream-ready の Android デバイスと接続して動作します。

このコントローラですが、BLE デバイスとしては単純なつくりになっています。

^{*3} ニュース&イベント：ニュースリリース | 株式会社ジャパンディスプレイ <http://www.j-display.com/news/2016/20161121.html>

^{*4} Lenovo Mirage Solo with Daydream | スタンドアロン型 VR ヘッドセット | レノボジャパン <https://www3.lenovo.com/jp/ja/vr-smartdevices/augmented-reality/lenovo-mirage-solo/Mirage-Solo/p/ZZIRZRHVR01>

Bluetooth LE (Low Energy) の用語で言うと、「ある Service のある Characteristic で 3 種のセンサとボタン・タッチパッドの情報を bit 列にして Notify しているだけ」です。

Daydream コントローラのハック

「iPhone でも Daydream コントローラは使えないか?」という話に始まった、Daydream コントローラをハックした事例があるので紹介します。

センサ類のイベント情報を BLE 経由で取得し、解析した話が「How I hacked Google Daydream controller - Hacker Noon^{*5}」という blog エントリに掲載されています。最終的には A-Frame を用いた iPhone 上の HTML コンテンツ上で Daydream コントローラの情報がとれるところまで簡潔にまとまっており、全貌を読み取るには少々の BLE の知識が必要ですが、ぜひ読んでみてください。

このハック内容によると、Daydream コントローラはセンサ情報を Bluetooth LE の GATT Notify で Central (スマートフォン母艦) 側に送信し続けています。その内容は、ジャイロ・地磁気・加速度センサそれぞれの X 軸・Y 軸・Z 軸の情報、Daydream・App・VolumeUp・VolumeDown・タッチパッドのクリック状態の各ボタンの on/off 情報、タッチパッドの XY 座標の情報などがぎゅっと 160bit の bit 列に詰められています。複雑な式による計算結果であったり、暗号化されたデータではないとのことです。

のことより、Daydream のコントローラ側はシンプルに IMU センサの取得値やボタン類の状態を送信するだけのつくりになっており、賢いアームモデルのエミュレーションはスマートフォン内部で行なっていることがわかります。

^{*5} How I hacked Google Daydream controller - Hacker Noon <https://hackernoon.com/how-i-hacked-google-daydream-controller-c4619ef318e4>

新型の Daydream View について



▲図 2.4 新型の Daydream View

2017 年 10 月 4 日、Google の新製品発表会が行われ、話題の Google Home ファミリーや Pixel 2/Pixel 2 XL とあわせて、新型の Daydream View^{*6}が発表されました（図 2.4）。日本では 12 月に入って発売されました。特にレンズの質がよく、値段アップは残念ですが良いアップデートです。

特徴を以下に述べます。

- 重い Android デバイスを装着時にバランスが悪くなりやすかったため、取り外し可能な上部のバンドを追加。あわせて頬にあたる部分のクッション形状を変更
- 熱暴走対策として、抑え蓋がヒートシンクの役割をするように
- 價格は\$99
 - \$20 アップ
- 日本でも発売
- 視野角が 10 度ほど向上
 - 約 90 度→約 100 度
 - シンプルな凸レンズから、特殊な形状のフレネルレンズに変更
- IPD 調節機構はなし

^{*6} Google Daydream View Tech Specs - Google Store https://store.google.com/us/product/google_daydream_view_specs?hl=en-US

新型 Daydream Viewについて、GoogleがYouTubeにプロモーション動画を公開しています^{*7}。この動画中に出てくるChromecastを使った、外部ディスプレイへのキャストは、10月上旬ごろからすでに既存のDaydreamで使えるようになっています。Daydreamアプリのホーム画面から「キャスト」を選択すると常時ストリーム状態になります。パフォーマンスには影響がありそうですが、大きな遅延はなくきちんと動作しており、デモなどいくつかのシーンでは役に立ちそうです。

Android 7.0 Nougat と VR モード

Android 7.0 NougatでDaydreamのために特別な対応が入り、AndroidプラットフォームとしてはVRモードというものが定義されました。Androidでは、デバイスの互換性について定義したCDD^{*8}というドキュメントがあり、その中にVirtual Realityという項目があります。CDDはAndroidデバイスそのものをつくる仕事でもしていなければあまり縁がないドキュメントなのですが、VRモードの定義の他、「Daydream」というVirtual Reality要件に対応したAndroidデバイスはどんな条件を満たさないといけないのか」という話はアプリ開発側の観点でも面白い情報が詰まっています。詳細を少しみていきましょう。

CDDの「7.9. Virtual Reality」の項目^{*9}に「7.9.1. Virtual Reality Mode」という記載があり、そこでは「VRモードをサポートすること。そのモードでは、通知などを双眼のHMDのためにステレオでレンダリングし、VRアプリケーションが動作している間はモノラル（单眼）のシステムUIを無効化すること。」とあります。ディスプレイの黒挿入モードやフレームバッファへのダイレクトレンダリング、センサ情報の低遅延伝達あたりはこの「VRモード」と直接は関係ない模様です。

^{*7} Meet Google Daydream View | Dream with your eyes open - YouTube <https://www.youtube.com/watch?v=PNBL2DpB1YE>

^{*8} Android Compatibility Definition Document <https://source.android.com/compatibility/cdd>

^{*9} Android CDD 7.9. Virtual Reality - https://source.android.com/compatibility/android-cdd#7_9_virtual_reality

7.9.2. Virtual Reality High Performance

If device implementations identify the support of high performance VR for longer user periods through the `android.hardware.vr.high_performance` feature flag, they:

- [C-1-1] MUST have at least 2 physical cores.
- [C-1-2] MUST declare `android.software.vr.mode` feature.
- [C-1-3] MUST support sustained performance mode.
- [C-1-4] MUST support OpenGL ES 3.2.
- [C-1-5] MUST support Vulkan Hardware Level 0 and SHOULD support Vulkan Hardware Level 1.
- [C-1-6] MUST implement `EGL_KHR Mutable_render_buffer`, `EGL_ANDROID_front_buffer_auto_refresh`, `EGL_ANDROID_get_native_client_buffer`, `EGL_KHR_fence_sync`, `EGL_KHR_wait_sync`, `EGL_IMG_context_priority`, `EGL_EXT_protected_content`, and expose the extensions in the list of available EGL extensions.
- [C-1-7] The GPU and display MUST be able to synchronize access to the shared front buffer such that alternating-eye rendering of VR content at 60fps with two render contexts will be displayed with no visible tearing artifacts.

▲図 2.5 CDD 7.9.2 Virtual Reality High Performance（冒頭）

つづけて「7.9.2 Virtual Reality High Performance」という項目に"high performance VR"という表現で、Daydream の要件が記載されています（図 2.5）。一部を抜粋して紹介します。

- これらの要件は `android.hardware.vr.high_performance` feature flag を有効にするのに必要な項目である
- [C-1-2] `android.software.vr.mode` feature flag の宣言が必須である

Feature Requirements^{*10}に記載されている通り、この二項目が「Daydream が有効であるか」を判定するためのフラグになっています。逆に言うと、Android 7.0 Nougat 以降を搭載しており、rooted な Android デバイスに必要な apk などを移植してこのフラグを有効にすると、非対応デバイスであっても Daydream が動作することが確認されています^{*11}。実態としては Snapdragon820 以降であれば快適に動くとのことで、この CDD で記載された要件を満たしていればそりゃ動かないはずはないよね、という話の模様です^{*12}。

*¹⁰ Daydream Feature requirements <https://developers.google.com/vr/distribute/daydream/functionality-requirements>

*¹¹ Reddit Got Daydream to work on my 5X, might work on other Android 7.0 phones as well https://www.reddit.com/r/Android/comments/5hno6u/got_daydream_to_work_on_my_5x_might_work_on_other/

*¹² Android よくできてんな。

- [C-1-11] 最低でも 3840x2160@30fps-40Mbps の H.264 のデコードのサポートが必須である(1920x1080@30fps-10Mbps が 4つ、あるいは 1920x1080@60fps-20Mbps が 2つ同時にデコード可能、に相当)
- [C-1-12] 最低でも 1920x1080@30fps-10Mbps の HEVC と VP9 のデコードのサポートが必須、できれば 3840x2160@30fps-20Mbps に対応すべき(1920x1080@30fps-5Mbps が 4つ同時にデコード可能、に相当)

4K の H.264 は対応してくれないと困るよ、というのが読み取れます。4K HEVC は SHOULD で書かれているので少し弱めです。

- [C-1-14] スクリーン解像度はフル HD (1080p) が必須、QuadHD (1440p) 以上を強く推奨
- [C-1-15] ディスプレイサイズは 4.7 インチから 6.3 インチの間であること
- [C-1-16] ディスプレイは VR モード時に最低 60Hz で動作すること
- [C-1-17] ディスプレイは Gray-to-Gray、White-to-Black、Black-to-White がの応答性能が 3ms 以下であること
- [C-1-18] ディスプレイは残像が 5ms 以下の低残像モードの対応が必須

このあたりがディスプレイ要件です。現時点の Daydream 対応スマートフォンは全て有機 EL ですが、VR 向けの低残像・高密度・高い開口率の液晶が徐々に発表されるプロダクトにも適用されはじめています^{*13}。

- [C-1-19] Bluetooth 4.2 と Bluetooth LE (Low Energy) と Data Length Extension 対応必須

BLE は Daydream コントローラ用の要件です。

- [SR] android.hardware.sensor.hifi_sensors の対応を強く推奨

hifi sensor は Android 6.0 から定義が追加された「高い忠実性 (high fidelity) をもつセンサ」のことです。Android の要件としては加速度センサやジャイロセンサの性能は「50Hz が必須」レベルなのですが、hifi sensor では「400Hz またはそれ以上が必須」となっています。頭部の姿勢予測に使われているものと考えられます。

^{*13} Pioneering the Frontier of VR: Introducing Oculus Go, Plus Santa Cruz Updates <https://www.oculus.com/blog/pioneering-the-frontier-of-vr-introducing-oculus-go-plus-santa-cruz-updates/>

2.2 Cardboard と Daydream の歴史的経緯

2014年5月、Google I/O 2014のkeynoteで、ダンボールにアクリル製レンズをはめ込んだゴーグル型のVRデバイス、Cardboardが発表されました。Googleが設計図を公開したことと、その構成要素がシンプルであったため、互換品が数多く制作され、世界中に広まっていきました。

Oculus Rift DK1/DK2によって確立された、ジャイロセンサによる姿勢予測、スマートフォン向けの（安価な）高精細で応答速度に優れた有機ELディスプレイと、複雑な光学系機構の代わりにソフトウェア側でレンズにあわせて画面表示を歪めておく樽型歪み補正（Barrel Distortion）を組み合わせ、安価なデバイスで高度なVR体験を実現する手法を、そのままその要素を搭載したスマートフォンで実現したのがCardboardでした。

ところが、Cardboardはあくまでエントリー向け、PCのそれと比較して機能が劣るのはもちろん、場合によっては安からう悪からう低品質なVR体験デバイスとしての地位を確立してしまいました。

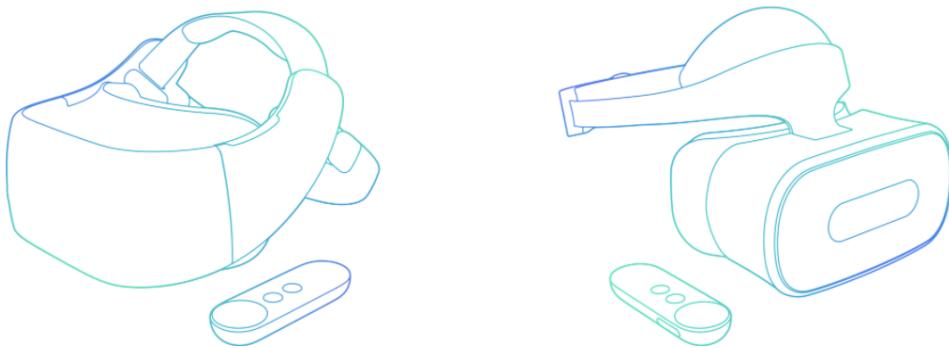
主に3つの要因があります。1つ目は、Androidデバイスにとって加速度・地磁気・ジャイロセンサの高い測定分解能がハードウェアとしての必須要件ではなかったこと。特にローエンドのAndroidデバイスに多いジャイロセンサ未搭載のデバイスではドリフト対応の副作用として、Google VR SDKは正常に動作しません。2つ目は、幅広いAndroidデバイスのスペックバリエーションに対して、VR空間を実現するのに十分なクオリティで60fpsを維持したCGのレンダリングが大変シビアであったこと。3つ目に、いわゆる「360度動画」と呼ばれる、ステレオ全天球などの形式である動画コンテンツの大きな解像度・fpsを問題なく扱うことができるハードウェアチップはごくハイエンド向けに限られることが挙げられます。

そんな状況の中、2016年5月、Google I/O 2016にて、Google VRの新たなプラットフォームとしてDaydreamが発表されました。Daydreamは前述の問題点が、特定の基準を満たしたハードウェア要件を満たすDaydream-ready phonesと、Androidプラットフォームとしてのソフトウェアでの対応が密に連携することで解決されます。また、それらに加えてコントローラの標準装備、HMDを外すことなくアプリを切り替えることができるホーム・ランチャー対応とGoogle Playへの融合、「なるべく快適にコンテンツを楽しむ効率をあげる」ための工夫がDaydreamにはたくさんつまっています。

モバイルVRの基準が、Daydream級にすみやかに移行してほしいと思います^{*14}。

^{*14}iOS 12でVRKitが出て、オルタナティブガールズとかときめきアイドルとかラブプラスEVERYとかがあっさりとCardboardを捨ててくれる未来を夢見ています。

2.3 Daydream standalone



▲図 2.6 Vive (HTC) (左) と Lenovo (右) の発表時の Daydream standalone

2017年5月、Google I/O 2017にて、スタンドアローン型のDaydreamヘッドセット、Daydream standalone^{*15}が発表されました。

ひとことで言うと「Androidが埋まっていて、電源を入れるとDaydreamのホームアプリが立ち上がるVRヘッドセット」です。特徴はTangoをベースとしたWorldsenseと呼ばれる技術により、ポジショントラッキング（平行移動）に対応していることです。

Vive (HTC) 製と Lenovo 製の2つ（図 2.6）が予定されていましたが、HTCはVive FocusというDaydream プラットフォームではないスタンドアロンヘッドセットへと方針を転換したためこちらからは離脱し、Lenovo 製のMirage Soloのみが2018年5月に発売となりました。

旧来のDaydream（Cardboardも同様）はヘッドトラッキングと呼ばれる、頭の向きのみを考慮したトラッキングになっています。体を大きく前に乗り出したり、敵の弾を避けようと横に上半身を躲すような動作に対応しておらず、そのような動作をした際には体の動きとヘッドセットで表示される視界に不一致が起きるため、酔いに繋がってしまいます。

そのため、「回らないソファにゆったりと座った状態で軽く首を動かすことを想定し、大きく体をひねって真後ろの敵を撃つようなコンテンツは避けること」といったことが推奨されています^{*16}。

*15 Daydream standalone <https://vr.google.com/daydream/standalonevr/>

*16 VR／AR／MRは「空間的コンピューティング」の波だ Googleが語るDaydream・Tangoの基

Daydream standalone でも大きく歩き回るようなコンテンツは安全上の観点から NG ではあるのですが、上半身ごと頭を左右に大きく動かしたり、立ったりしゃがんだりといった動作に対応します。

今までと同様にソファに座ってゲームなどのコンテンツを楽しむ際にも、「座る」という動作で首が上から下に移動したり、そして少しエキサイトして体を前に乗り出すような動作をしても、それを WorldSense が適切にトラッキングし、自然な視界を再現します。アーリが 6DoF に対応していると、今までの 3DoF 環境における「首の動きにあわせて空中でディスプレイがついてくるような違和感」とは縁がなくなります。

スタンドアローン型のヘッドセット、Oculus Go も先日の F8 にて発売がアナウンスされました。同時期にローンチとなった一体型、スタンドアロン VR ヘッドセットのこの 2 つについて、比較してみましょう（表 2.1）。

▼表 2.1 Daydream standalone(Mirage Solo) と Oculus Go

項目	Daydream standalone	Oculus Go	備考
ベース技術	Daydream + Tango	Gear VR	
発売元	Lenovo	Oculus	
アピリストア	Google Play	Oculus Store	
価格	\$399 (51,200 円)	\$199 (23,800 円, 32GB モデル)	
SoC	Snapdragon 835	Snapdragon 821	
ヘッドトラッキング	対応	対応	
ポジショントラッキング	対応 (WorldSense)	非対応	
発売時期	2018.05	2018.05	
コントローラ	3DoF + タッチパッド + ボタン	3DoF + タッチパッド + ボタン (トリガ)	

Oculus Go は Xiaomi 製で、Snapdragon 821 搭載とは思えない低価格を実現してきました。Fixed Foveated Rendering と 72Hz の Refresh レート、新しい液晶により Galaxy S7 の Gear VR より高性能で、S8 の Gear VR 環境と同等もしくは一部の観点では上回る性能であると言えます。

Mirage Solo は Snapdragon 835 を搭載し、カタログ上は一回り上のパフォーマンスで動作します。WordSense によるポジショントラッキングはかなり快適に動作し、システム上ある程度の歩行にはブラックアウトなどの安全上の制限がかけられているものの、上半身を大きく左右に揺らしたり、立ち上がったりしゃがんだりする動作に対しては、快適にポジショントラッキングが働きます。

これらのスタンドアローン機はどのようなメリットがあるのでしょうか。現行機の

「Galaxy S8 と Gear VR のセットで\$749」や「Pixel 2 と Daydream View のセットで\$749」という価格帯は気軽に購入できるものではありません。「Oculus Go の\$199」や「Daydream standalone の\$399」は販促目的での VR 体験キャンペーンなどに 20 台とか用意する際の費用や、物理的な取り回しにおいてメリットがあるため、そのあたりがハードルとなっていた層には確実に刺さると思われます。

2.4 AR と ARCore とポジショントラッキング

AR 戦国時代がいつのまにか始まりました。HoloLens の MR や、2 機種目が市販された Google Tango が盛り上がって…というテクノロジー主導ではなく、どちらかというとソーシャル主導で始まったように見えています。

写真 SNS である Instagram が台頭する中、楽しく"盛れる"snowなどのアプリによるリアルタイムの画像処理技術の進化競争が行われていた中、"たまたま"ポケモン GO がやってきました。ポケモン GO で、人は「カメラ映像に CG を重畳する AR」を思い出してしまったのです。この流れの分かりやすい例としては、Snapchat による、世界を AR で装飾する World Lenses^{*17}があります。

ポケモン GO はなんとなくあの UI になってしまったわけではなく、「現実世界に拡張した情報を重ね合せる」という本質的な AR サービスである Google マップを開発していたバックグラウンドがあって、十分に人の心が動いて、その結果として体が動いてもらうための機能を検討をした上で今のような実装になったことが、CEDEC 2017 のセッションで説明されています^{*18}。

そんな中、まっすぐに AR 機能を提供する ARKit が iOS11 でやってきました。ARKit は買収した Metaio^{*19}の技術が使われていると言われていますが、こちらもタイミングとしては"たまたま"できあがったものと推測されます。なぜなら、歴史的に iOS は「既存のサービスのクローンを作りやすくする機能、あるいはクローンそのもの」をぶつけてくることが多いのですが、ARKit についてはそうではありません。前述の写真を"盛る"ための機能として iPhone X の TrueDepth カメラシステムによる 3D ウ〇コがやってきた話とは違うのです。

そんな ARKit ですが、世間におおいに担ぎ上げられたせいなのか、Google は大慌てで

*17 Snapchat adds world lenses to further its push into augmented reality - The Verge <https://www.theverge.com/2017/4/18/15333130/snapchat-world-lenses-something-new-for-facebook-to-copy>

*18 『ポケモン GO』を開発したナイアンティックが目指す“世界を動かす AR”【CEDEC 2017】 ファミ通 App https://app.famitsu.com/20170901_1129022/

*19 Apple が定評のある拡張現実スタートアップ Metaio を買収していた | TechCrunch Japan <http://jp.techcrunch.com/2015/05/29/20150528apple-metaio/>

Tango を ARCore に方針転換することになってしまいました。その ARCore は ARKit とほぼ同等で、Depth もとれず、Area Learning による外界検出結果の保存・読み出しには対応していません。Tango と比較して、機能としてはかなりの劣化版です。

本質的な AR は Google マップすでに日常使われています。「カメラ映像に CG を重畳する」の延長線上としてみんなが期待する電腦メガネやオーグマーは、HoloLens がその入り口に立ったところで、もう少し将来の話です。今の AR ブームは波がすぎると、現状のニッチであっても市場ができた VR のように定着しないと思っています。

そんな ARKit と ARCore ですが、VR の世界から眺めて「モバイルにポジショントラッキング技術が来た」と考えると、次の一手が楽しみになります。Windows MR や Oculus の Santa Cruz のように複数のカメラでもなければ、Tango や iPhone X のように Depth カメラを持つわけではない、ハードウェアがよりシンプルで難易度の高い Visual SLAM に Apple と Google というモバイルの二大巨頭がパワーをつぎ込んでくれるわけです。

iOS 12 で VRKit が来るのが先か、\$199 の Daydream standalone 2 が来るのが先か、Tango の取り下げで凹んだ分はきっちり他の分野が凸ってくれるといいなと思っています。

第3章

Daydream を使う

この章では、Daydream の使い方について紹介します。セットアップから始まり、Google Play との連携具合や、開発する際に便利なデバッグ設定について記載します。

3.1 Daydream のセットアップ

初期セットアップ



▲図 3.1 Daydream のセットアップ

Daydream のアイコンから Daydream アプリを立ち上げます。まずは VR モードではない通常の 2D の UI で、初期セットアップウィザードが始まります。Daydream のアプリも通常のアプリと同様の Google Play で公開されており、そこからダウンロードする仕組みになっていますので、Google アカウントを選択します（図 3.1-左）。

Daydream 対応ヘッドセットを持っている場合のみ続行できます（図 3.1-中）。ただし、抜け道があり、Daydream View を持っていないとも、Bluetooth 経由で別の Android デバイスを Daydream コントローラエミュレータとして使うことができます。これについては後述します。



▲図 3.2 VR モードに移行する際に表示されるエントリー画面

Daydream コントローラのペアリングを行った後（図 3.1-右）、エントリー画面（図 3.2）を経て、ここ以降は VR モードでのチュートリアルが始まります。

チュートリアルとホーム画面



▲図3.3 チュートリアル画面

きちんとバンドがしまっていて、首をふってもヘッドセットがずれたりしないかの確認に始まり（図3.3-左上）、コントローラの使い方を順番に教えてくれます^{*1}。

とくにこのDaydreamボタン（旧：ホームボタン）の長押しによる中心のリセット（図3.3-左下）は重要操作なので覚えておきましょう。

^{*1} 2017年10月15日現在、手元の環境では一部の文字が豆腐になっており、見事に読みませんでした。日本語が正常に表示されている画面もあり、詳細は不明です。



▲図 3.4 Daydream のホーム画面

チュートリアルが終わると、Daydream のホーム画面が表示されます（図 3.4）。ピックアップアプリや動画が上段に、中段には最近使ったアプリが並びます。下段は左から、アプリ「Play ストア」、インストール済みアプリ一覧の「ライブラリ」、Chromecast でアプリ画面を写す「キャスト」、「設定」です。

セットアップが終わると、どのアプリを試せばいいのか迷うのですが、まずは「Google ストリートビュー」をオススメします。次に、徐々に VR 空間での使い勝手がよくなってきた「YouTube」で 360 度・3D（ステレオで立体的に見える）動画を閲覧しましょう。もしあなたが THETA ユーザで、全天球の画像をそれなりに保有してあるのであれば、「Google フォト」も大変オススメです。

ゲームなどのダイナミックなコンテンツもいいのですが、現状はこれだ！ というキラータイトルが定まっていないように思えます。まずは写真・動画系のアプリで Daydream の操作に慣れたあとでゆっくり探して見てください。

3.2 デベロッパー向け設定と VR Entry Flow

この節では、Daydream のデベロッパー向けの設定を用いて、不要な操作を省略したり、Daydream View がなくとも Daydream アプリの利用や開発を可能にする方法を紹介します。ユーザにトラブルなく Daydream を利用してもらうための特別なフローであ

る「VR Entry Flow」の他、様々な仕組みがあって、特別な設定や操作の上に可能となります。

VR Entry Flow

Daydream には、コントローラのペアリングや、必要な権限の設定を、ヘッドセットをかぶる前にチェックの上、必要な操作をユーザに促す VR Entry Flow^{*2}という作業フローが定義されています。

VR アプリの起動時など、非 VR の状態から VR モードに遷移する際に、以下のフローに従います。

1. 設定済みでなければ、コントローラをペアリングします
2. 必要があれば、コントローラのファームウェアをアップデートします
3. 必要があれば、Google VR Services を有効にして、必要な権限を設定します
4. Bluetooth や NFC の他、必要な設定が ON になっていなければ設定画面を呼び出します
5. 実施していなければ、Daydream ヘッドセットの設定を行います
6. ガイド（図 3.2）を表示し、スマートフォンをヘッドセットに挿入します
7. コントローラと接続し、ヘッドセットとコントローラの向きを初期化、キャリブレートします

これらのフローは Google VR Services によって自動的にハンドリングされ、通常アプリ開発者はこの流れを気にする必要はありません。ただし、ホームアプリのランチャー や、タスク一覧からあなたの VR アプリを立ち上げた際に、Android ライフサイクルに沿って起動したあなたの VR Activity は、VR Entry Flow によって割り込まれ、中断されます。Daydream アプリに限定された話ではなく Android アプリとして当然の話ではありますが、あなたのアプリがいつ中断されても問題がないように作っておく必要があります。

デベロッパー向けの設定

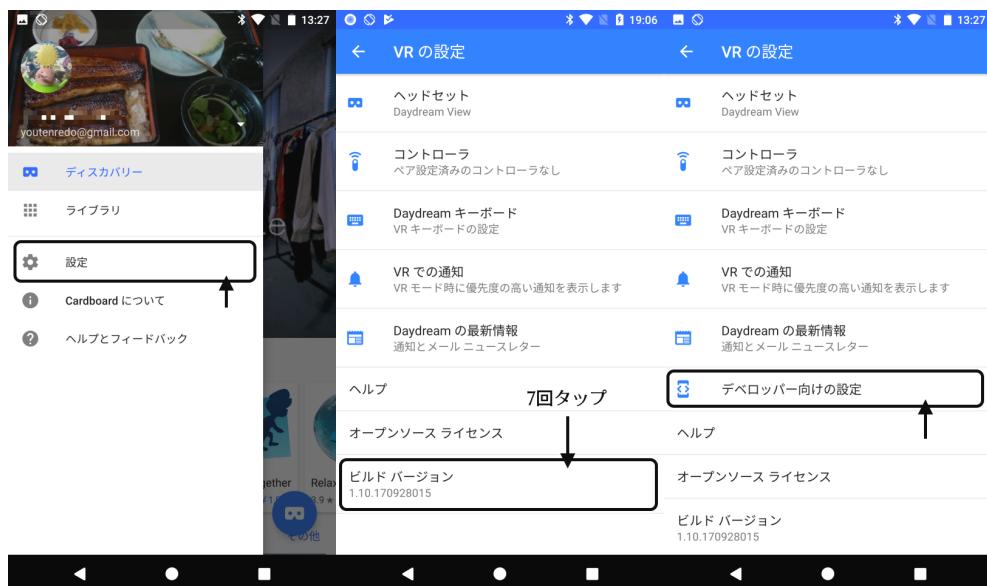
Daydream には、デベロッパー向けのデバッグ設定があります。このうち、「VR Entry Flow をスキップする設定」と「Bluetooth 経由でのコントローラのエミュレータを使う設定」の 2 つが大変重要ですので覚えておいてください。前者はコントローラを操作する必要がないアプリの利用時や機能の開発時に便利です。また、この 2 つの設定を組み合わ

^{*2} VR Entry Flow | Google VR | Google Developers <https://developers.google.com/vr/daydream/guides/vr-entry>

せると、Daydream View を持っていないとも Daydream アプリの利用や開発を行うことが可能になります。

デベロッパー向けの設定については公式サイトの「The Controller Emulator^{*3}」のページに記載されていますが、手順のみがシンプルに掲載されており、少しあわづらいです。もう少し噛み砕いて手順を確認していきましょう。

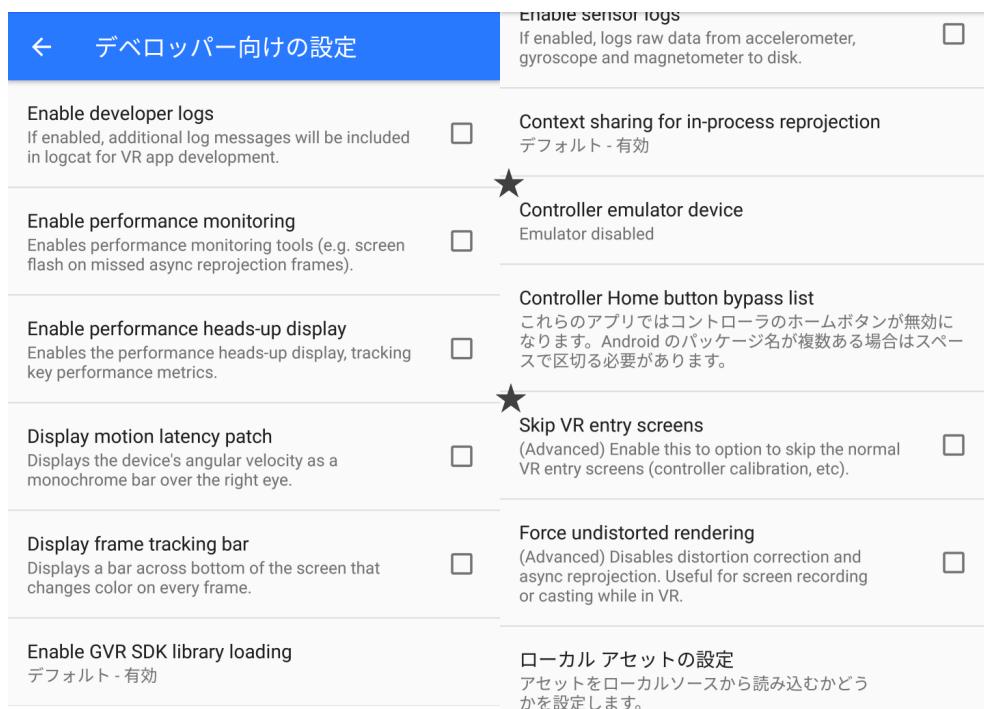
前節の Daydream の初期セットアップの最後、コントローラのペア設定画面（図 3.1-右）まで進めると、実は初期セットアップとしてはコントローラの設定を残して完了しているため、Daydream アプリを終了して、再度起動すると設定が完了したかのようにホーム画面が表示されます。



▲図 3.5 デベロッパー向け設定の有効化

ホーム画面の左上のハンバーガーメニュー(≡)から設定画面（図 3.5-左）を呼び出し、「ビルト バージョン」を 7 回タップすると（図 3.5-中）、「デベロッパー向けの設定」隠しメニューが表示されるようになります（図 3.5-右）。

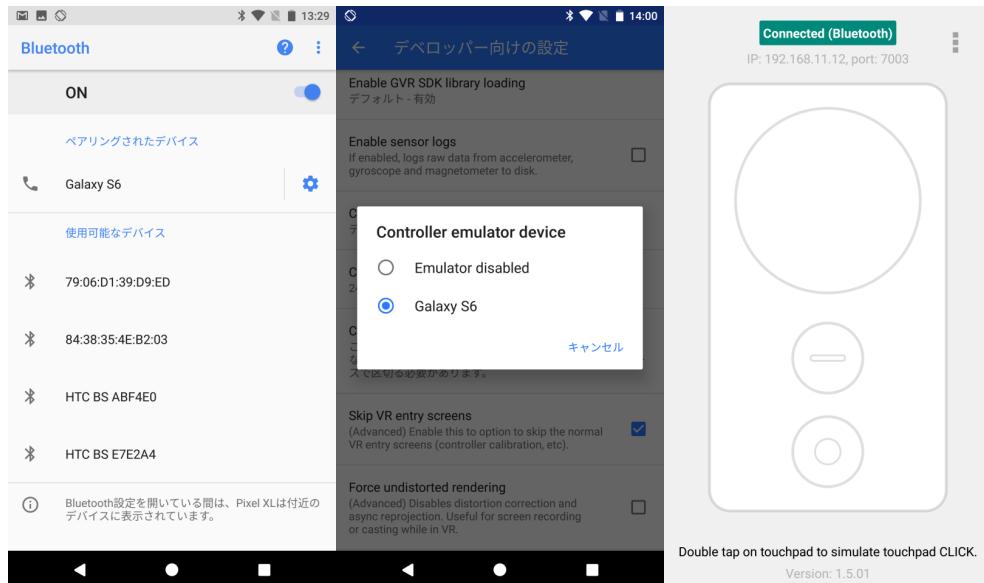
^{*3} The Controller Emulator | Google VR | Google Developers <https://developers.google.com/vr/daydream/controller-emulator>



▲図 3.6 デベロッパー向けの設定

ここまで、Daydream View とコントローラを使わないまま、セットアップを続けています。実際にアプリを起動した際にコントローラとのペアリングを要求されるのを回避するために、デベロッパー向けの設定中の、「Skip VR entry screens」を有効にしておきましょう（図 3.6）。

Daydream コントローラエミュレータ



▲図 3.7 コントローラエミュレータのセットアップ

続けて、コントローラエミュレータのセットアップを行い、動かしてみましょう。

エミュレータアプリは別 Android デバイスで

Daydream コントローラのエミュレータは、専用の Android アプリが公開されています。Daydream 母艦となる Android デバイスとは別の Android デバイスを用意してください。Android 4.4 KitKat 以降であれば動作します。Daydream-ready phone がもう 1 台必要というわけではありません。

Bluetooth のペアリング設定

まず、2 台の Bluetooth 設定を開いて、ペアリングを行います（図 3.7-左）。この際、エミュレータ側のデバイス名を区別がつきやすい名前に変えておくとわかりやすいです。

次に、Daydream のデベロッパー向け設定の「controller emulator device」を選択し、ペアリング設定を行ったデバイスを選択します（図 3.7-中）。

前述の公式サイト^{*4}から、「Controller Emulator」の apk をダウンロードし、インストールします^{*5}。このアプリは、Bluetooth 経由の他、WiFi や USB ケーブル経由で Unity や Unreal Engine などのエディタ上でのコントローラ操作のエミュレートにも使えます。

コントローラエミュレータを試す

セットアップが完了したら、実際に動かしてみましょう。Daydream アプリを立ち上げ、右下の VR ゴーグルアイコンをタップします。「Skip VR entry screens」設定が有効になっていると、本物のコントローラのペアリングも、Daydream ボタンの押下も要求されません。

二眼ステレオの VR モードになったら、もう一台の Android で「Controller Emulator」アプリを起動します。エミュレータアプリの表示が「Connected (Bluetooth)」になったら、正常に動作しています。うまくいかない際には、Bluetooth の設定や、アプリの起動順番に気をつけつつ、何度かやり直してみてください。アプリ UI 上にも説明文言が表示されていますが、タッチパネル部のクリックのエミュレートには、アプリ上でのダブルタップで対応しています。

Daydream View がなくとも

ここまで手順で、Daydream View がなくとも、Daydream-ready phone またはその設定が可能な Android デバイスがあれば Daydream の世界をちょっと覗いてみたり、Daydream アプリ開発を試してみることができます。ただし、あくまでお試しで。Daydream の真の力はきちんとした環境でぜひ体験してください。

^{*4} The Controller Emulator | Google VR | Google Developers <https://developers.google.com/vr/daydream/controller-emulator>

^{*5} インストール手順は省略します。Android の SDK が導入済みであれば「adb install」を、一般ユーザは「提供元不明のアプリのインストール」を試してみてください。

3.3 Daydream View 互換機 BOBOVR Z5 を試す



▲図 3.8 BOBOVR Z5 Daydarem Edition

Daydream View 互換機とは

前章で述べた通り、Daydream View のヘッドユニットは Cardboard と同等のもので、コントローラは BLE で（比較的）シンプルな情報を送信しているだけです。このため、ハックの上、クローンが作れるということです。

BOBOVR Z5

Cardboard 向け VR ゴーグルや、独自の VR のプラットフォームを展開している BOBOVR から、BOBOVR Z5^{*6}という Daydream View 互換機が登場しました（図 3.8）。\$59 で AliExpress 経由で購入しましたが、今は Amazon^{*7}でも取り扱いがある模様です。

ヘッドユニットのレンズまわりに難あり

Daydream のコントローラ部は問題なく動作しました。タッチパネル部の形状の違いから操作感が違うのですが、許容範囲だと思います。

ただし、公式サイトにも書かれていますが、旧型の Daydream View の視野角である

^{*6} BOBOVR Z5 <http://www.bobovr.com/product/bobovrz5/>

^{*7} Amazon.co.jp : BOBOVR Z5 Daydream View VR ゴーグル <https://www.amazon.co.jp/dp/B071GYQR8V>

90度を120度に広げたヘッドユニット部の独自拡張がよろしくなく、端が歪んで見えます。首を振ると顕著に感じられ、気持ち悪く感じます。

結論としては、Daydream互換機として動作はしましたが、格別安いというわけでもなく、決してオススメはできない機種です。日本特有の例のアレの話もありますし、おとなしく新型のDaydream Viewを購入してください。

第 4 章

Unity で Daydream アプリ開発の入り口を覗く

さて、概要を知って、使い方を知ったその次は、Daydream アプリ開発の入り口を覗いてみましょう。

Google VR SDK（略称：GVR SDK）として、様々なライブラリやツールが揃っています。古い情報も残っていたり、Cardboard 向けや iOS 向けの話もあり、さあ冒険に出かけよう、と足を踏み出しても迷子になりやすい状況とも言えます。

この章では、入り口からの歩き方として、はじめに試すべきサンプルやツールの紹介します。

4.1 Hello, Daydream

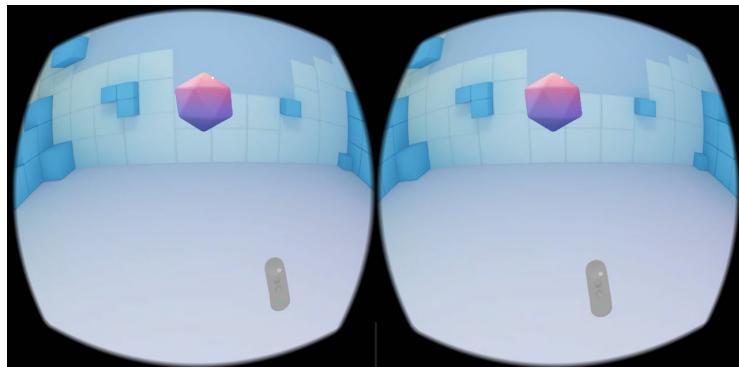
まずは Hello World からです。

Google VR SDK はネイティブ（Java, C/C++）や Unreal Engine もサポートしていますが、Unity の情報が一番多く、試す際にはオススメのプラットフォームです。Unity のバージョンは推奨されている「Unity2017.4 系の LTS Release」を、GVR SDK は最新の「v1.130.1」を利用しましょう。

基本的には以下のページの手順に従います。

- Get started with Google VR in Unity on Android
 - <https://developers.google.com/vr/unity/get-started>
1. 新規 3D プロジェクトを作成し、「GoogleVRForUnity_*.unitypackage」をダウンロードしてインポート
 2. API の自動アップグレード確認ダイアログが出る際には、了承して実行

3. 「GoogleVR > Demos > Scenes」配下の「HelloVR」シーンを開く
4. 「Build Settings」を開いて、「HelloVR」シーンをビルド対象に追加、Platform を「Andorid」に「Switch Platform」を実行
5. 「Player Settings」を開いて、以下の設定を変更
6. 「Other Settings」より「Package Name」を適当な文字列に変更
7. 同じく「Other Settings」より「Minimum API Level」を「Android 7.0 'Nougat' (API level 24)」に変更
8. 「XR Settings」の「Virtual Reality Supported」をチェックして有効に、「Virtual Reality SDKs」の「+」ボタンを選択して、「Daydream」を追加
9. 「Build and Run」を実行、apk の名前は適当に入力



▲図 4.1 HelloVR サンプルシーン

うまくいくと、無機質な部屋に、色のついた物体が浮かんでいるアプリが立ち上がりります（図 4.1）。Daydream コントローラを（ホーム長押しで）有効にして、ポインタをあててみましょう。ポインタで示して、タッチパネルをクリックすると、物体が部屋内のどこかにワープします。

4.2 Instant Preview

前節で紹介した HelloVR サンプルには Daydream 向けのステレオ表示とヘッドトラッキングの（自動的な）制御から、コントローラの接続状態のハンドリング、コントローラ操作での 3D オブジェクトに対する Ray（光線）制御によるヒット判定とメニュー操作と、基本が詰まっており、根幹としてはこれが Daydream の全てです。

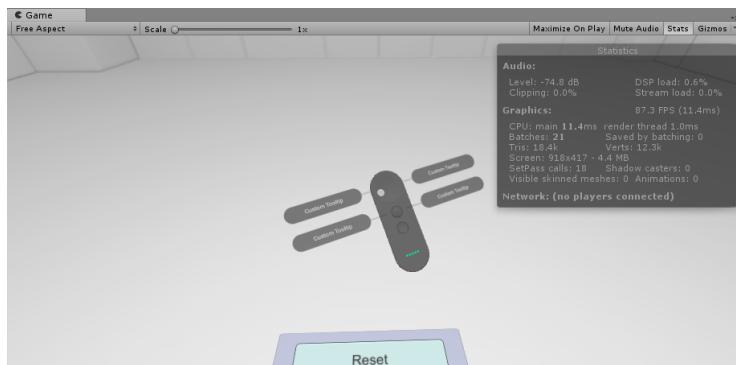
さて、つづけて具体的なアプリを開発していきましょう、となった際に、毎回ビルトと

Android デバイスへの転送、インストールを待つ…というサイクルを繰り返すのは効率的ではありません。できれば Unity エディタ上で再生して、開発サイクルを回していくたいですよね。

Unity エディタ上で頭の振りやコントローラ操作のシミュレーション

まず、Unity エディタの Game ウィンドウに対する操作で、頭を振ったり、Daydream コントローラの操作がシミュレートされています。HelloVR のプレビュー再生中に、以下の操作を試して見てください。

- Alt + マウス移動：頭の回転を操作
- Ctrl + マウス移動：頭の傾きを操作
- Shift + マウス移動：Daydream コントローラの傾きを操作
- Shift + マウス左クリック：Daydream コントローラのタッチパッドのクリック
- Shift + マウス中央クリック：Daydream コントローラの Home ボタンのクリック（recenter 扱い）
- Shift + マウス右クリック：Daydream コントローラの App ボタンのクリック
- Shift + Ctrl + マウス移動：Daydream コントローラのタッチパッド上のスワイプ操作



▲図 4.2 Unity エディタ上で Daydream コントローラのタッチパッドシミュレーションの様子

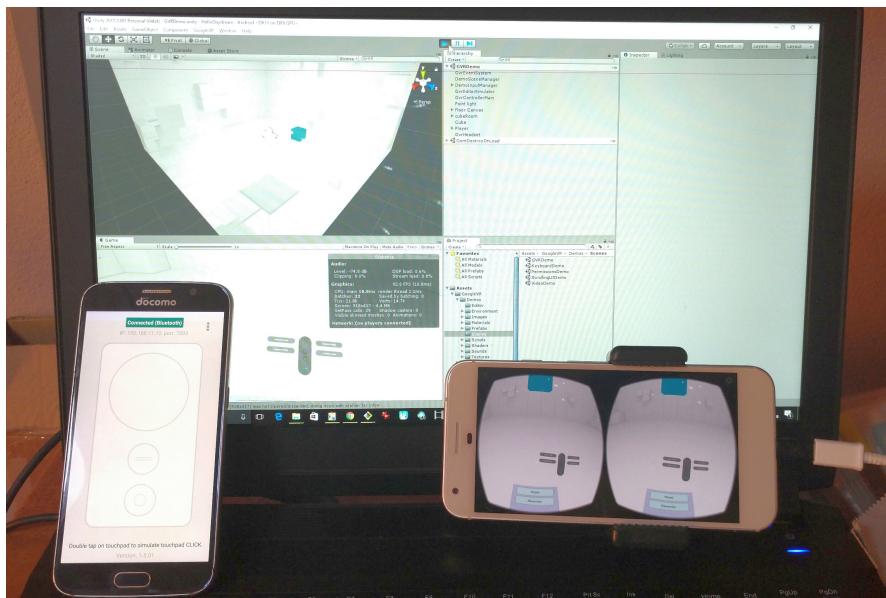
ひととおりの動作ができるようになっています。少々操作は複雑ですが、タッチパッドのスワイプ操作も可能です（図 4.2）。

ヘッドセットを被ってプレビューしたい

Unity エディタ上では平面でしかありませんので、ヘッドセットを被った上で立体で確認したいですし、首振りやコントローラ操作はもう少し直感的に試したいところです。そこで Instant Preview の登場です。

- Instant Preview

– <https://developers.google.com/vr/tools/instant-preview>



▲図 4.3 Instant Preview と Controller Emulator セットでの動作の様子

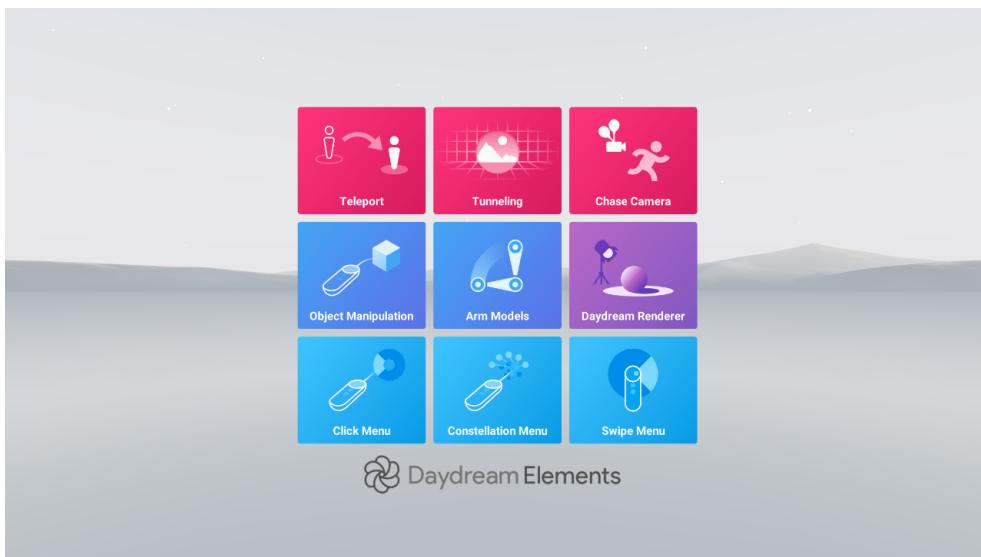
Instant Preview は、専用のアプリを Android デバイス上で動作させ、Unity エディタと通信させてこれを実現するものです。画像は Unity エディタから Instant Preview アプリへの映像ストリーミングで実現します。また、逆方向で Daydream コントローラの操作による入力も可能です。コントローラの操作については前章で紹介した Controller Emulator アプリによる Daydream コントローラのエミュレートにも対応しています（図 4.3）。USB・Wi-Fi 経由いずれも使えます。ただし、Wi-Fi 経由については映像ストリーミングのための通信帯域の観点や、特定のマルチキャストアドレスとポートによる通信が可能なネットワークである必要があり、USB 経由の方が安定している印象です。

前節で動かした HelloVR シーンに、Instant Preview に必要な Prefab がすでに含まれています。Daydream 母艦となる Android デバイスを USB で接続して、Unity エディタで再生すると、必要なアプリのインストールと起動は自動的に行われ、映像の再生がすぐに始まります。

コントローラについては、実デバイスであれば Daydream ボタンの長押しを、Controller Emulator であればアプリを起動して Bluetooth の接続を待つだけです。

adb (Android Debug Bridge) まわりの仕組みを使っているため、Unity エディタから Android アプリのビルドとインストール・実行が正常に動く環境であれば問題なく動作するはずです。うまくいかない際には紹介したページ内に「Troubleshooting in Unity」という項目がありますのでそちらを参照ください。

4.3 Daydream Elements



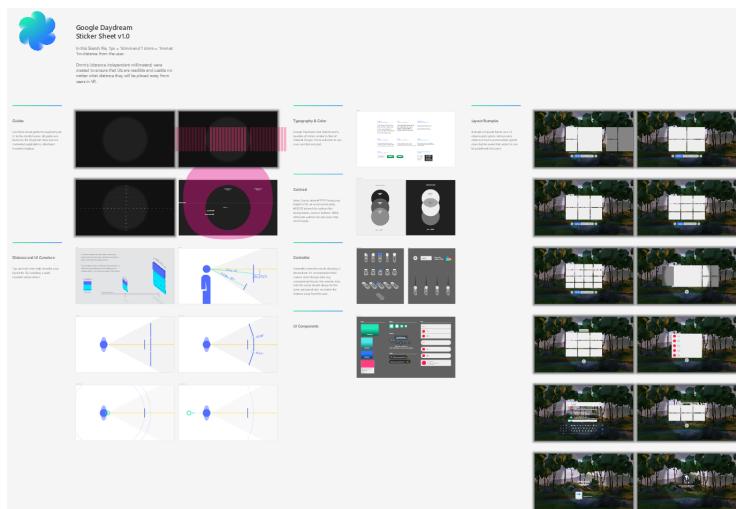
▲図 4.4 Daydream Elements 起動画面メニュー

- Daydream Elements Overview
 - <https://developers.google.com/vr/elements/overview>

Daydream Elements は Daydream のサンプル集です。アプリは Google Play で公開されている他、ソースコードも GitHub で公開されています。

ものすごくよくできたサンプル集で、以下に特徴をあげます。

- GVR SDK のライブラリ群の一つとして位置づけられ、多数のデモサンプルが含まれる
- 数多のデモサンプルは 1 つのアプリにまとまっていて、そのメニュー や階層構造と各デモ間の遷移も操作感が大変よい
- 各デモは丁寧な説明看板と遊び心がたっぷり詰まったインターラクションのあるプレイアブルなものになっている
- 統一感のある UI で Daydream のデザインガイドラインが直感的にわかるようになっている

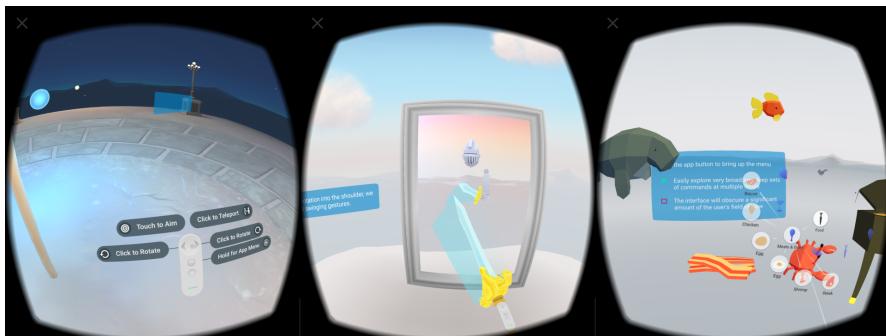


▲図 4.5 Daydream デザインガイドライン資料 (sketch ファイル) の一部

Daydream のデザインについては、ガイドラインがあります。Sticker Sheet というページ^{*1}から、sketch ファイルが 1 つ（図 4.5）と、Google I/O で解説した YouTube 動画へのリンクが貼られています。

Daydream のホームアプリはもちろん、Daydream Elements アプリや後述する Media App Template アプリもこのガイドラインでデザイン統一されています。一度目を通しておくとデザインガイドラインとその根拠がわかります。

^{*1} Design | Sticker Sheet <https://developers.google.com/vr/design/sticker-sheet>



▲図 4.6 Daydream Renderer (左)・Arm Models (中)・Constellation Menu (右)

光源数などに制限がある、モバイル向けに最適化された動的なライティングを行う Daydream Renderer、様々な移動やメニュー・インターフェースのデモ、傾きのみの 3DoF ながら腕の構造のエミュレーションによって擬似的に 6DoF に近い動きを実現する Arm Model とそのパラメータを設定し、確認するデモなど、多数のデモが含まれています。ぜひそれぞれのドキュメントを読みながら確認してみてください。

4.4 VR Video と Daydream Media App Template アプリ

GVR video plugin

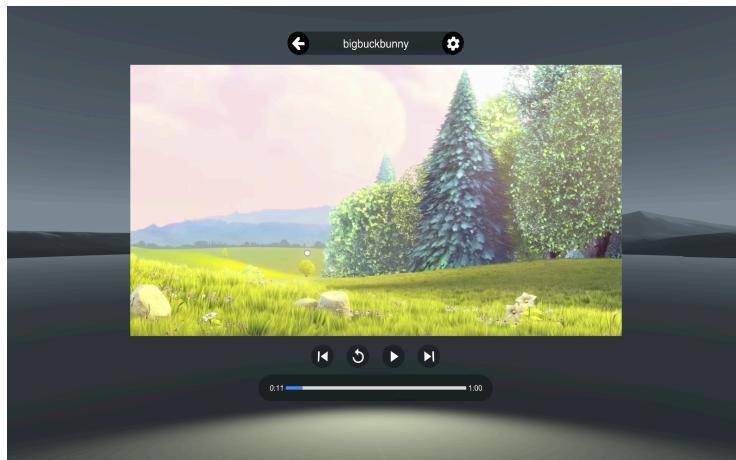
Daydream Elements と似た位置づけですが、GVR SDK for Unity には「GVR video plugin」という動画再生ライブラリが含まれます。

VR 動画として必要な、「モノラル/Top-Bottom のステレオ/Left-Right のステレオ」×「フラット /180 度/360 度」の 3x3 の 9 パターンの動画再生に対応しています^{*2}。ExoPlayer をベースにしており、シークやサムネイル取得も含め、ローカルファイルの再生のほか、HLS や DASH のような adaptive streaming^{*3}にも対応しています。

^{*2} フォーマットやスペックについては「第 2 章「Daydream を知る」」で述べた、CDD と Daydream の要件の項を参照ください。

^{*3} 直接の意味はネットワーク状態に応じて再生品質を制御するストリーミングのことです。ただし、ここでは「近年のいい感じのストリーミング技術」ぐらいの意味が近いと思われます。

Daydream Media App Template



▲図 4.7 Daydream Media App Template アプリでの 2D 動画再生中画面

この GVR video plugin にも、Daydream Elements と同様に、メディア再生アプリのテンプレートとなるひととおりの機能の揃ったサンプルアプリ（図 4.7）と unitypackage 形式のリリースパッケージ、そしてもちろんソースコード一式が GitHub で公開されています。

- Daydream Media App Template
 - <https://developers.google.com/vr/unity/media-app-template>
- GitHub: googlevr/media-app-template: Daydream Media App Template
 - <https://github.com/googlevr/media-app-template>



▲図 4.8 ファイラー表示（左）と VR 動画のためのステレオ・投影設定メニュー（右）

特徴を以下に述べます。

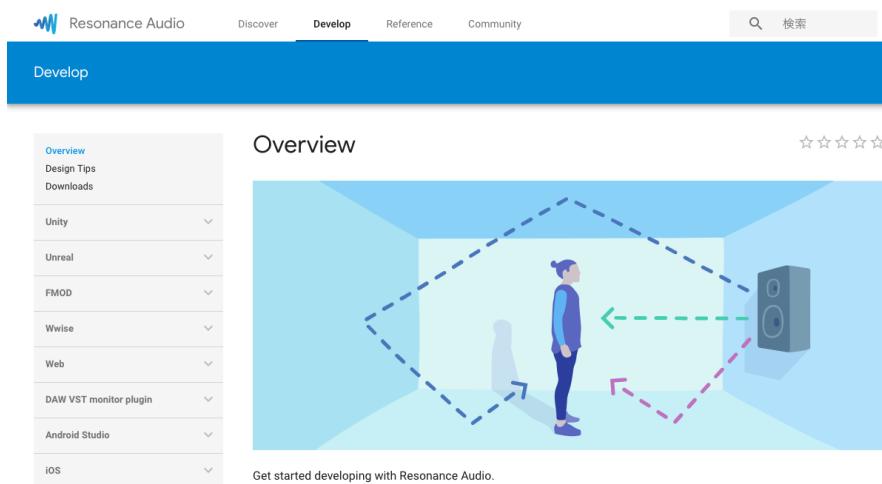
- メディアライブラリにアクセスするための Permission をユーザに確認するフロー実装に対応
- ファイラーとしてフォルダ表示と移動、ファイルサムネイル表示とそれらの選択に対応（図 4.8-左）
- 静止画・動画の再生に対応
 - 静止画またはフラット動画は縦スワイプによる拡大縮小、ドラッグによる表示位置の移動に対応
 - VR 動画としてのフォーマット選択メニューでステレオの Top-Bottom/Left-Right や 180 度/360 度の切り替えにも対応（図 4.8-右）
 - 動画のシークに対応、ポインタのオーバーレイによるシーク位置のサムネイルプレビューにも対応
- Daydream のデザイン・つくり観点でのガイドラインを満たしている

VR 動画はドラッグによって水平方向の回転も可能です*4。

*4 縦方向（ピッチ）の回転にも対応しているとベストだったのですが、それぐらいは自分で改造すればいいってことですよね、はい。

ローカルに VR 動画の資産がいくつかある方はぜひ試してみてください。サンプルアプリとは思えないレベルで作り込まれているなど感じました。ストリーミングサービスのバックエンドがあるなど、特定目的の Daydream メディアアプリを開発したい際には、大変有力な手助けになると思われます。

4.5 Resonance Audio と Audio Factory アプリ



▲図 4.9 Resonance Audio

Daydream Elements で操作と UI について、Daydream Renderer でグラフィック周りを、そして Media App Template で VR 界特有の静止画や動画の扱いについてカバーしてきましたが、そしてこの Resonance Audio^{*5}は名前の通り音の話です。

Resonance Audio SDK の守備範囲については公式サイト（図 4.9）に書いてある通りではありますが、（筆者も概ねこのレベルですが）「モノラルとステレオの違いまではわかるけどそれ以上ってなんだろう」ぐらいの方は、CEDEC 2017 のカプコンの岸 智也氏の公演レポ^{*6}が大変参考になるので一通り目を通しておくのをオススメします。

^{*5} <https://developers.google.com/resonance-audio/develop/overview>

^{*6} [CEDEC 2017] 基礎からすっかり分かる「これで解決！ ゲームに必要な 3D オーディオの全て」の聴講レポート - 4Gamer.net <http://www.4gamer.net/games/999/G999905/20170906078/>



▲図 4.10 Audio Factory アプリ

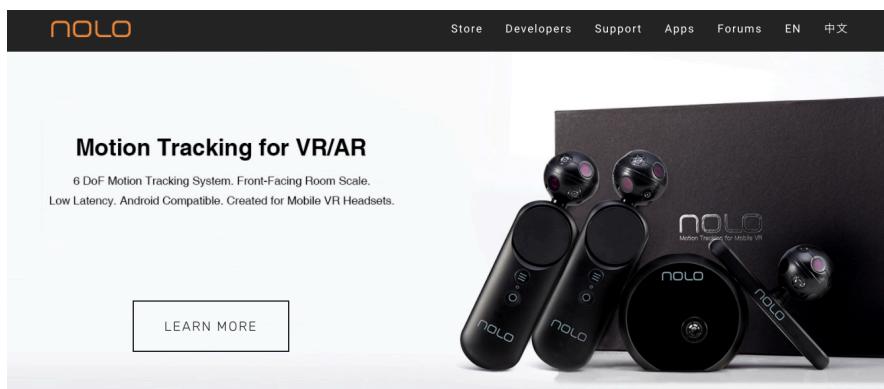
この、Resonance Audio SDK を利用したサンプルとして、Audio Factory アプリ^{*7}というデモアプリが公開されています。Spacial Audio を効果的に組み込んだモバイル VR でもここまでやれるということを示す最高のデモになっていて、ぜひヘッドホンを用意して体験してください。

また、Audio Factory アプリは GitHub でリポジトリ一式が公開されています^{*8}。

^{*7} <https://play.google.com/store/apps/details?id=com.google.vr.audiofactory&hl=ja>

^{*8} https://github.com/resonance-audio/audio_factory

4.6 6DoF モーショントラッキングシステム NOLO を試す NOLO と Kickstarter



▲図 4.11 NOLO 公式サイト

NOLO^{*9}は 6DoF のモーショントラッキングシステムです（図 4.11）。ベースステーション、ヘッドセットマーカー、2つのモーションコントローラからなるセットです。モバイル形態で利用時には、完全にケーブルレスで、フロント方向限定ながら 2つのコントローラを含むルームスケールでのポジショントラッキングを実現します。PC 経由で利用時にはケーブルが 1本だけ伸びますが、RiftCat + VRidge^{*10}を利用することで、低遅延の映像ストリーミングにて、SteamVR のアプリをプレイすることができます。

2017 年の 1月末に Kickstarter^{*11}でのクラウドファンディングが開始され、NOLO 一式のコースが \$89～\$109 とそれなりの価格で設定されました。Cardboard や Gear VR 環境のスマートフォンは持っていても、HTC Vive や Oculus Rift などの PC VR 環境を持っていないユーザにとって、NOLO は価格面においても期待された選択肢の一つでした。

ケーブルレスでポジショントラッキング付き、「6DoF のモバイル VR」としては Daydream standalone や Oculus の Santa Cruz や、ARKit/AR Core + Daydream が

*9 NOLO - Motion Tracking for VR <https://www.nolovr.com/>

*10 VRidge - Play PC VR on your Cardboard <https://riftcat.com/vridge>

*11 NOLO: VR Motion Tracking for Mobile and SteamVR Play by Lisa Zhao — Kickstarter <https://www.kickstarter.com/projects/243372678/nolo-affordable-motion-tracking-for-mobile-and-ste>

競合デバイスと考えられますが、NOLO はその先駆者と言えます。NOLO のもう一つの形態である PC 経由については、HTC Vive のケーブルが新型で細くなうことや、Steam VR 対応の Windows MR の登場もあって残念ながら今後はかなり厳しそうです。

モバイルモードでアプリを試す



▲図 4.12 モバイルモードでの NOLO 一式

さて、そんな NOLO ですが、back して届いたものを長らく放置していたのですが、いまさらになって試してみました。ここからは試した内容についてお届けします。

NOLO の公式サイト、「Support」内に英語版の「Product Manual」が公開されています^{*12}。初期セットアップの「Pairing」と、今回は「Android Phone」（Cardboard 相当と思われます）を選択します。

ベースステーションの電源を入れてペアリングモードにした後、コントローラ 2つとヘッドセットマーカーの 3 ユニットをペアリングしていきます。ここまででは操作もマニュアル通りで、LED の表示も親切であり、とくにトラブルなくセットアップが終わりました。NOLO のモバイルモードは Android の USB Host 機能を使っていますので、ヘッドセットマーカーと Android を付属の USB ケーブルで繋ぎます。Gear VR のヘッドセットと同様の仕組みですが、NOLO には Gear VR のような統合管理を担当するコンパニオンアプリがないため、このあたりから操作難易度が上がってしんどくなってきます。

NOLO Tetris^{*13}というアプリが公式サイトからダウンロードできる他、Google Play

^{*12} User Guide — NOLO - Motion Tracking for VR <https://www.nolovr.com/user-guide-en>

^{*13} NoloTetris - Google Play <https://play.google.com/store/apps/details?id=com.lyrobotix>.

でも公開されているようなので試して見ましょう。



▲図 4.13 ヘッドセットマーカー接続時の確認ダイアログ（左）と NOLO Tetris のプレイ画面（右）

手順としては以下の通りです。決して少なくない上に、とくにヘッドセットマーカーを USB で接続時に表示されるダイアログの OK がタップしづらくてたまりません（図 4.13-左）。

1. ベースステーションの電源を入れる
2. コントローラの電源を入れる
3. ヘッドセットマーカーを VR ゴーグルにセットする
4. スマートフォンとヘッドセットマーカーを専用の USB ケーブルで接続し、USB Host 機能の確認ダイアログで OK をタップする
5. スマートフォンを VR ゴーグルにセットする
6. トラッキングがうまくいっていなかったら 2. あたりからやりなおし

空中に浮かんでいるブロックから好きなのを選んで盤面に近づけると、通常のテトリスのようにハマリこみます（図 4.13-右）。ちゃんと横ラインを消すと消えるので確かにテトリスの模様です。

トラッキングがうまく行く勝率は3割ぐらいで、一度うまくいくとその後しばらくは安定します。ただし、そもそもすべてがうまくいってる状態でもヘッドセットもコントローラもプルプルと震え^{*14}、数秒に1回程度、10cmから30cmほどスイーツと滑ります。これは厳しい。

他のアプリも試して見ます。もう一つの宇宙戦争っぽいアプリはコントローラが動かず、SDK for Android Java のサンプルアプリはスクリーンショット例のようにきちんとログは出てくれず、正常に動いていないように見えました。Forumを探しているうちに見つけた NOLO GUN というアプリは動くには動いたのですが、銃をもったり弾をこめたりする操作性が悪すぎてとても辛かったです。

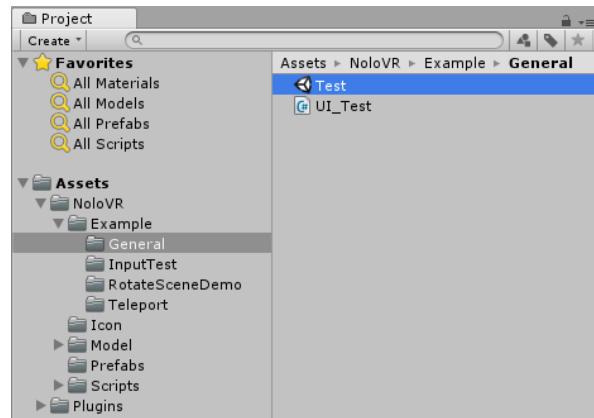
NOLO + Daydream

さて、そんなプルプルにもスイーツにもめげずに、Daydream で動くか試してみました。空中や水中で浮いてるようなケースに限定すればそれなりに楽しめる可能性がありますからね！ 基本的には GitHub 上で公開されている NOLO-Unity-SDK の「Get Started」^{*15}の手順通りです。この際に、Daydream コントローラは使わないため、前章で述べた「VR Entry Flow」の無効化設定は ONにしておくことをオススメします。

Unity のバージョンは 2017.2.0f3 で確認しています。NoloVR_SDK_1.1.7_20171017.unitypackage をダウンロードしてインポートします。

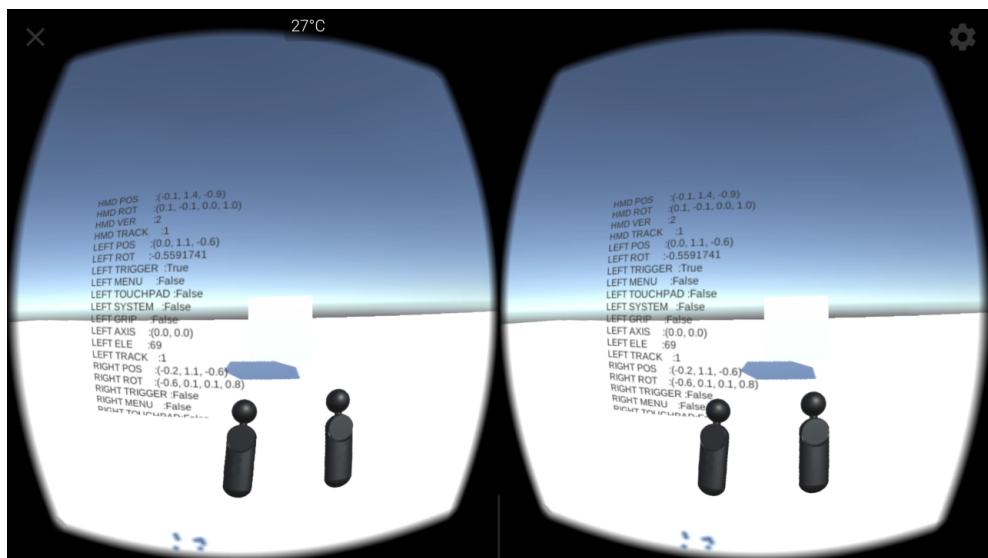
^{*14} ヘッドセットの位置認識がプルプル震えるとどうなるかというと、VR 空間上で世界がプルプルと震えます。SaGa2 の非常階段から最下層みたいな感じです。

^{*15} NOLO-Unity-SDK/GetStarted.md https://github.com/NOLOVR/NOLO-Unity-SDK/blob/master/Docs/en_us/GetStarted.md



▲図 4.14 SDK 内 Example フォルダ配下の Test サンプルシーンを選択

SDK 内、Exapmple フォルダ配下にはいくつかのサンプルシーンが含まれています。ひとまず SDK としてコントローラやヘッドセットマーカーの空間上の座標や向き、ボタンの状態などの情報が表示される、「NoloVR > Example > General > Test」シーン（図 4.14）を試してみます。



▲図 4.15 Test シーンの動作画面

前述の NOLO Tetris と同様に、プルプル揺れたりスイーツと水平方向に滑る現象は（残念ながら）とくに変わりはありませんでしたが、各ユニットの座標・傾き、ボタンの状態はきちんととれていることが分かりました（図 4.15）。首振り制御の競合などが懸念された Daydream としての動作もとく問題ないようです。API の動作が素直だと何かを作ってみようという気になりますね。

そんなわけで可能性は感じました。決してオススメはできませんが、機会があればちょっとだけ試す分には面白いデバイスだと思います^{*16}。

^{*16} 相性問題が多くセットアップ難易度が高すぎるとか、不安定ですぐ落ちるとか、ハック口がまったく開いていないとか、そういうデバイスではないということです。ガジェットのクラウドファンディング結果としては満足しています。

第 5 章

Hello, Mirage Solo

5.1 本章について

本章は主に発売前にお借りした Mirage Solo にて、ビルド番号 OVR1.180406.005 のファームウェアでの確認事項を掲載しています。今後のアップデートにより記載内容について変更される可能性があることをご了承ください。

5.2 Mirage Solo とは

ホーム > VR/AR > augmented-reality > lenovo-mirage-solo > Lenovo Mirage Solo with Daydream

Lenovo Mirage Solo with Daydream

VRの可能性を解き放つ

Lenovo Mirage Solo with Daydream は世界初¹のスタンドアロン型でかつDaydream対応、WorldSense™搭載のVRヘッドセットです。スマホも、パソコンも、ケーブルも必要なく、空間内の動きを把握する新しいテクノロジー「WorldSense™」により、一台で手軽にVRの世界を楽しむことができます。これまで以上の没入感で、新たな楽しみを今あなたに提供します。

1: Google Daydream VRヘッドセットとして、スマートフォン、外部センサー、PCとの接続を必要としないスタンドアロン型は世界初。
(2016年4月24日現在、レノボ調べ)

*お届けは、5月11日以降となります

販売価格: ￥55,296

税込・送料無料

予約販売開始

販売代理店モデルを購入



▲図 5.1 Mirage Solo - Lenovo

Mirage Solo^{*1}は 2018 年 5 月、Lenovo から発売となった単独で動作する一体型の VR ヘッドセットである、Daydream standalone 機です。Daydream standalone そのものについては「第 2 章「Daydream を知る」」すでに説明済みですので、そちらを参照願います。

製品仕様へのコメント

製品仕様^{*2}について、以下に抜粋してコメントしておきます。

- Qualcomm ® APQ8098 オクタコア プロセッサー（最大 2.45GHz）：Qualcomm Snapdragon835 VR Platform ベースです。
- Daydream 2.0: Android ベースで Daydream 2.0 Euphrates アップデートが全て含まれます。セパレート側の Daydream でも概ね対応されていますが、ChromeCastなどを用いた専用单眼画面のスクリーンキャスト、ホーム画面とは別の通知確認画面、左右アーム設定のクリック変更などが含まれます。
 - コントローラ操作による静止画・動画スクリーンショットは、現状開発者向けオプションで有効にする必要があります。全ユーザへの解放を計画しているとのことです。
 - YouTube アプリの「皆で見てる方向を共有」はアプリのアップデートで対応されました。
 - Chrome アプリの WebVR を含む VR フル対応は 2018.05 現在、準備中のこと。
- 5.5 型 IPS パネル (2560 x 1440 ドット)：液晶です。かなり綺麗です。「非球面フレネルレンズ、110° FOV」とセットで、現行機ではトップクラスの画質です。
- WorldSense 対応デュアル・モーション・トラッキング・カメラ：Android の標準カメラ API では使えません。モノクロで、とくに今後 API が解放される予定もないとのことです。
- Android N Pro オーディオ：Android 公式ドキュメントの「オーディオ レイテンシ^{*3}」の android.hardware.audio.pro の項のことを指していると思われます。

^{*1} <https://www3.lenovo.com/jp/ja/vr-smartdevices/augmented-reality/lenovo-mirage-solo/Mirage-Solo/p/ZZIRZRHVR01>

^{*2} <https://www3.lenovo.com/jp/ja/vr-smartdevices/augmented-reality/lenovo-mirage-solo/Mirage-Solo/p/ZZIRZRHVR01>

^{*3} オーディオ レイテンシ | Android NDK | Android Developers <https://developer.android.com/ndk/guides/audio/audio-latency>

5.3 セットアップ

それでは主に開発者向け観点でのセットアップを実行していきましょう。

初回セットアップ

それでは、電源を入れてセットアップを行いましょう。

起動後、まず Daydream コントローラの認識シーケンスが始まります。画面の指示にしたがって、コントローラの Daydream ボタン（一番下のボタン）やタッチパッドの領域を操作してください。

「Hi, there」画面にて言語を選択します。インターネットに繋がった Wi-Fi 環境に接続の上、Google アカウントのセットアップを行います。キーボードは選択した言語に依存しますので、Wi-Fi パスワードやメールアドレスを入力する際には、地球儀の用なマークを選択して、英語（半角英数）モードに切り替えてください。

その後、チュートリアルが始まります。チュートリアルも空間音響とポジショントラッキングがよく効いた素敵なデモになっていますので、できればヘッドホンかイヤホンを接続の上実施してください。

開発者向けオプションと **adb** の有効化

何はともあれ、開発者向けオプションを有効にして、adb を使えるようにしましょう。以下に、開発者向けオプションと adb の有効化手順を示します。



▲図 5.2 Dashboard 画面

- Daydream ボタンを 1 回押すと、Dashboard が表示されます（図 5.2）。
 - Dashboard では、よく使うアプリの起動や通知の確認、各種設定が行えます。
- 右上のギアマークの「設定」を選択し、さらに「全ての設定」を選択すると、2D ビューで設定画面が表示されます。
- 「設定」>「システム」>「端末情報」から、「ビルド番号」を 7 回タップします。メニュー構成は違いますが、お馴染みの操作です。「これでデベロッパーになりました！」という Toast が表示されれば成功です。
- ひとつメニュー階層を戻って、「開発者向けオプション」から「USB デバッグ」を有効にします。
- Android SDK がセットアップ済みの母艦 PC と USB ケーブルで接続し、adb devices コマンドなどで接続を試みると、お馴染みのフィンガープリント確認ダイアログが出ますので許可してください。確認が終了すると、adb が有効になります。

開発者向けオプションの罠

設定画面のような2Dビューを特殊な方法で実現しているのか、画面オーバレイ表示を行う開発者向けオプションが一部うまく動きません。ポインタの座標情報を表示する「ポインタ表示」設定はONにするとその後正常に画面が描画されなくなり、`factory reset` 必須になりますのでご注意ください。

Daydream のデベロッパー向け設定



▲図 5.3 Daydream のデベロッパー向け設定

「第3章「Daydream を使う」」の「デベロッパー向け設定と VR Entry Flow」でも紹介した、Daydream 向けデベロッパー向け設定は Mirage Solo でも有効です。「設定」>「Daydream」>「VR の設定」>「デベロッパー向けの設定」(表示されていなければ「ビルドバージョン」を7回タップ)にて開くことができます。

デベロッパー向け設定の紹介

一部抜粋して説明します。チェックしても特に効果がないように見えるオプションもあります。

- 「Enable performance monitoring」+「Enable performance heads-up display」: VR アプリ中にパフォーマンスマニタを HUD の形でオーバレイ表示します。ただし、非一体型と違ってスクリーンショットに写らないため、確認にはひと手間かかります。
- 「Force 6DoF in Apps」: 非 6DoF 対応アプリにて、強制的に 6DoF ヘッドトラッキングを可能にします。ただし、全天球動画など、カメラから特定距離に張り付いたオブジェクトに近づけるわけではありませんので、過度の期待は禁物です。
- 「Enable safety graphics」: デフォルトで有効になっていますが、これを外すことにより「カメラリセット位置から一定距離離れると注意事項とともにブラックアウトする」安全のためのガードを外すことができます。これを off にすることを前提としたアプリは Google Play では配布できません。展示向けなど、安全に配慮できるケースにおいては、WorldSense のポジトラの性能を最大限活かすことができます。
- 「Force undistored rendering」: 動画キャプチャや Chromecast などへのスクリーンキャスト時に、強制的に樽型歪みを解除します。あまり詳細が分かっていませんが、静止画・スクリーンショット絡みの設定を有効にする際に、あわせて有効にしておくのが良さそうです。
- 「Enable VR screen recording via chords」: 有効にすると、コントローラの「Daydream ボタン（一番下のボタン）+ボリュームダウンの長押し」で静止画スクショが、「Daydream ボタン+ボリュームアップボタンの長押し」で動画の撮影開始/撮影停止ができるようになります。
 - 「You will need to grant permission to Google VR Services」という、Google VR サービスへの権限設定の操作がわかりづらいのですが、次のように Mirage Solo 内で操作します。
 - Dashboard の「すべての設定」より、「アプリと通知」>「N 個のアプリをすべて表示」>右上の「:」より「システムを表示」>「Google VR サービス」>「権限」>ストレージの権限を ON

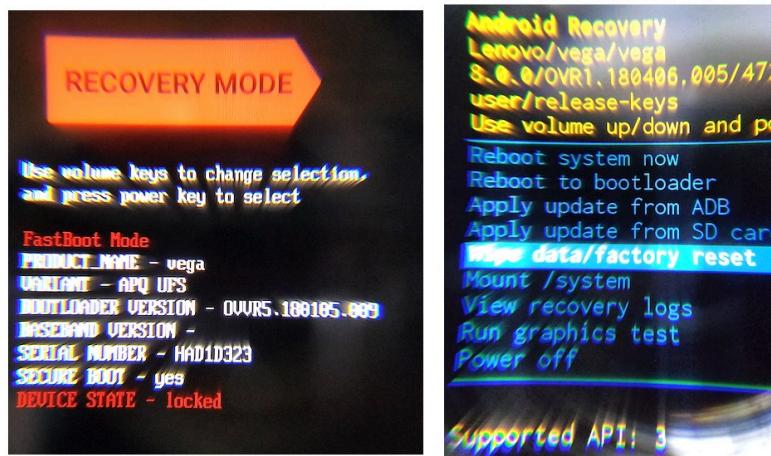
- 「Enable VR screen recording via ADB」：有効にすると、adb コマンドから静止画・動画キャプチャができるようになります。コマンドが少々複雑なため、alias や macro を組むのをオススメします。
 - 詳細はメニューにある通り、コマンドからヘルプを表示することができます。
 - `adb shell am startservice -n com.google.vr.vrcore/.capture.recorder.RecorderService --es command HELP`

adb 経由の静止画・動画キャプチャと alias

adb 経由の静止画・動画キャプチャですが、コマンドが少々複雑なこと、MTP が（2018.05 現在）動かないため、ファイル名指定と、adb pullを組み合わせた alias や macro を組むことをオススメします。以下、コマンド説明と alias 例です。

- `mirage-image` : 静止画キャプチャを行ったあと、メディアファイルを Mirage Solo から取得します。
 - `alias mirage-image='now=$(date "+%Y%m%d-%H%M%S");adb shell am startservice -n com.google.vr.vrcore/.capture.record.RecorderService --es command IMAGE --es path /sdcard/Pictures/${now}.jpg;sleep 1;adb pull /sdcard/Pictures/${now}.jpg :'`
- `mirage-start` : 動画キャプチャを開始します。
 - `alias mirage-start='now=$(date "+%Y%m%d-%H%M%S");adb shell am startservice -n com.google.vr.vrcore/.capture.record.RecorderService --es command START --es path /sdcard/Movies/${now}.mp4;'`
- `mirage-stop` : `mirage-start` とセットで使用します。動画キャプチャを停止し、メディアファイルを Mirage Solo から取得します。
 - `alias mirage-stop='adb shell am startservice -n com.google.vr.vrcore/.capture.record.RecorderService --es command STOP;sleep 1;adb pull /sdcard/Movies/${now}.mp4 :'`
- `mirage-photo` : 写真アプリを 2D ビューで起動します。
 - `alias mirage-photo='adb shell am start -n com.google.android.apps.photos/.home.HomeActivity'`

fastboot と factory reset



▲図 5.4 fastboot と recovery mode

他の Android デバイスと同様、fastboot メニューからの factory reset が可能です。操作不能に陥った際には実施しましょう。

- 電源 OFF の状態で、音量 (-) ボタンを押したまま、電源ボタンを 2 秒ほど押して話すと、fastboot モードで起動します（図 5.4-左）。
- fastboot モードにて、音量ボタンにて「RECOVERY MODE」を選択し、電源ボタンで決定操作をします。
- ドロイド君に赤い警告マークが出て「No Command」と表示された画面で、電源ボタンを押しながら、音量 (+) ボタンを押すと、RECOVERY MODE のメニューが表示されます（図 5.4-右）。
- 同じく音量ボタンと電源ボタンにて、「Wipe data/factory reset」を選択すると、factory reset が実施されます。「Wipe data」の通り、全てのデータは初期化されます。

その他便利な話・Intent 芸

Google Play のリモートインストール + adb shell am start で直接起動

ブラウザで Google Play のサイトを開き、リモートインストールすることが可能です。インストール後は `adb shell pm dump <package名>` にて対象の Activity を特定し、`adb shell am start` で直接起動しましょう。アプリの package 名は Google Play サイトの URL から確認できます。

いくつか起動コマンド例を載せておきます。

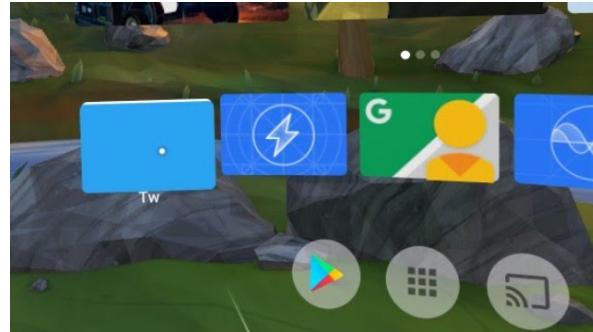
- Slack : `adb shell am start -n com.Slack/.ui.HomeActivity`
- Twitter : `adb shell am start -n com.twitter.android/.StartActivity`
- Google Photo (プリイン) : `adb shell am start -n com.google.android.apps.photos/.home.HomeActivity`

また、手順がとても面倒ですが、インターネットに接続されている状態であればメニュー操作でも起動することが可能です。Dashboard の「すべての設定」より、「アプリと通知」>「N 個のアプリをすべて表示」>起動したいアプリを選択>「アプリの詳細」で Google Play の該当ページを開く>「開く」を選択してアプリを起動する。

Daydream に対応した非 VR アプリも含めて起動するランチャーを作ってしまえば解決するかな、とも思っています。

直接起動するだけの Daydream アプリを作る

Intent もアプリ詳細を辿る操作も不便なので、「Daydream のルールにのっとりホームやライブラリにアイコンが表示される」かつ「Intent で特定パッケージのアプリを起動する」アプリを作ればいいわけです。



▲図 5.5 TwLauncher for Daydream standalone

作りました。

ひとまず、Twitter を直接呼ぶアプリを試しに作ってみました。Play で公開している^{*4}の他、GitHub にソースコード一式^{*5}を置いています。

ほぼコードを書いていないこのアプリがそれなりに便利だったので、既存のランチャー アプリを起動するショートカットアプリを作ればすべてが解決するのでは、と考えました。AtomicAdd Team の「小型デスクトップ（ランチャー）^{*6}」というアプリが便利だという情報を見つけましたので、これ向けにパッケージ名を変更したものを用意してみたところ、よさそうです^{*7}。ただし、いわゆる「ホームアプリ」を複数インストールするとややこしいことになりますのでそのあたりは知ってる人だけお試しください。

Bluetooth ヘッドホンなどを使う

Bluetooth 設定が塞がれていますが、Bluetooth 設定を Intent (`adb shell am start -a android.intent.action.MAIN -n com.android.settings/.bluetooth.BluetoothSettings`) にて直接開くと使うことができます。ただし、これについてはコントローラの Bluetooth 通信帯域確保のために意図的に塞いでいるとのことです。コントローラが不安定になる可能性があります。

^{*4} TwLauncher for Daydream standalone <https://play.google.com/store/apps/details?id=tw.youten.redo.tw>

^{*5} <https://github.com/youten/TwLauncher>

^{*6} <https://play.google.com/store/apps/details?id=com.atomicadd.tinylauncher>

^{*7} <https://github.com/youten/TwLauncher/blob/28e435de7c1f483691da5f8f73e0def648343c9a/tinylauncher.apk> 使ってみた動画はこちらのツイートを参照 https://twitter.com/youten_redo/status/995362623381159936

いろいろ動かない話

最後に、いろいろ動かない話を挙げておきます。今後のアップデートで改善されることを期待しています。

- Instant Preview が真っ暗
 - 近接センサを抑えると、コントローラ中継エミュレーションだけは動いているのを確認できます。こちらは修正予定とのことです。
- Vysor が動かない
- MTP が動かない
 - 正直びっくりしました。どうも Mirage Solo 内で一般ユーザがスクリーンショットなどを扱うことを考慮していないようです。前述したコントローラによるスクリーンショットを有効にした上で、adb pullまたは「2D View の Google Photo アプリを起動して自動バックアップフォルダを指定」することにより、インターネット経由で取得することができます。
- VR 対応 Chrome がない
 - Chrome VR (WebVR と VR モード正式対応のバージョン 68 と思われます) で正式対応予定とのことです。
 - Chrome Dev^{*8}という開発版の Android 向け Chrome がバージョン 68 に追いついていますので、これを install して、chrome://flagsという URL 入力から開く設定画面で、WebVR を有効にするとそれなりに動作します。「それなり」というのは、正式版ではない Chrome ではもともと機能が制限されていることと、マイク入力からの音声検索やタブ表示・切り替えなどが現状、正常に動作しません。
 - 2D View の Chrome は動作します。Chrome beta や firefox などの別ブラウザアプリを入れることもできます。
 - WebView や Chrome Custom Tabs は動作しましたが、筆者の確認手順ではカスタム URL scheme による Activity フックが動かない (OAuth 認証シーケンスが非一体型の Daydream と同じ実装では動作しません！) のを確認しています。Intent によるアプリ連携・分業が Android の最高のメリットなため、このあたりは早急に修正されてほしいなと思っています。

^{*8} <https://play.google.com/store/apps/details?id=com.chrome.dev>

あとがき

youten です。FGO に満足してきたのでガルパに手を出して、全バンドランク 20 にしてついでにアズールレーンに手を出しました。Live2D の力強さを感じます。

Daydream View がやっと日本に来る！ ということで Daydream 特集号になりました。4 月から（正確には 6 月末から）はじまった無職生活ですが、何か名刺になりそうなアプリを作ろう、その経験の中でもう少し価値のある具体的なコードが載る本を…と考えていたのですが、まったくそんな感じにはなりませんでした。Publishing（アプリ品質基準ルールと Play 審査の話）まわりも全く書けませんでした。今年度も半年すぎましたので、下半期は本気出していこうと思います！

NOLO はなかなか将来の厳しそうな子でした。ARKit/ARCore ポジショントラッキングがきっと仇をとってくれると思います。それではまた、どこかでお会いできましたら。

Twitter: @youten_redo

Web: ReDo -Refrigerator Door- れいぞうこのドア <http://greety.sakura.ne.jp/redo/>

VR コンテンツのつくりかた 2 Daydream 特集

2017 年 10 月 21 日 技術書典 3 版 v0.9.0

2017 年 10 月 23 日 v0.9.1

2018 年 5 月 5 日 Mirage Solo 対応版 v1.1.0

著 者 youten

(C) 2017 ReDo れいぞうこのドア