

ユーザジェネレート型コンテンツのためのリアルタイム配信システムの設計とプラットフォームの構築

山下 恭平^{1,†1,a)} 時村 晋太郎^{†2} 長内 大茂^{†2} 杉浦 一徳^{†1,b)}

概要: スマートフォンや SNS の普及とともに、コンテンツや流行の流れは目まぐるしく変化している。その中でも、YouTube Live やニコニコ生放送、Twicas といったリアルタイム配信プラットフォームの市場は拡大しており、5G といった広帯域のモバイルネットワークの供用開始により、より高品質やコンテンツの配信も可能となっている。しかし、このようなリアルタイム配信は未だ専門的な知識が必要であり、また高度な処理を長時間行う環境が必要である。本研究では、リアルタイム配信を支援する環境として、BaaS(Broadcast as a Service) やクラウドコンピューティングを用いた、初心者でも扱いやすい配信環境のプラットフォームの構築を行った。本実験ではモバイル端末向けに UI 設計を行い、映像伝送には WebRTC を用いて、5G や 4G 通信下において実験を行った。

キーワード: コンテンツミキシング、リアルタイム配信、クラウドコンピューティング、5G

Design and implementation of Real-time broadcasting system for User Generated Contents

Abstract: With the spread of smartphones and SNS, the flow of content and trends is changing. Among them, the market of real-time broadcasting platforms such as YouTube Live, Nico Nico Live, and Twicas is expanding, and the availability of high-bandwidth mobile networks such as 5G is making it possible to broadcast higher quality content. However, the real-time broadcasting requires specialized literacy and an environment. In this research, we implement a platform for real-time broadcasting that is easy to use even for beginners, using BaaS (Broadcast as a Service) and cloud computing as a support environment for real-time broadcast. The prototype is designed with UI for mobile devices, and experiments are conducted under 4G and 5G communications using WebRTC for video streaming.

Keywords: Contents Mixing, Realtime broadcasting, Cloud computing, 5G

1. はじめに

リアルタイム配信は、映像やコンテンツのミキシングやエンコーディングを行い、YouTube Live やニコニコ生放送、Twicas といったリアルタイム配信プラットフォーム等を通して配信を行う一種のコンテンツ形態である。近年ではスマートフォンなどモバイル端末の普及により、TikTok

といったより手軽に配信できるサービスが登場しており、投げ銭機能といった視聴者が配信者に対して電子的なアイテムや金銭を送る機能も登場 [1] し、リアルタイム配信の市場は拡大している。しかし、リアルタイム配信はコンテンツの入力、ミキシング、エンコーディング、ストリーミングを同時にリアルタイムで行う必要があり、負荷の高い処理を長時間に渡って安定して行う必要があることから、スマートフォンといった処理能力が限られたモバイル端末では非常に厳しい条件である。現在、スマートフォンから行う事のできる配信は、映像に対してフィルタリングを入れる程度であり、基本的にはカメラ入力をそのまま配信している。また、モバイル端末からの配信にたいして厳しい制約を課しているサービスも存在する。例えば YouTube

¹ 情報処理学会
IPSSJ, Chiyoda, Tokyo 101-0062, Japan

^{†1} 現在、慶應義塾大学
Presently with Keio University

^{†2} 現在、ソフトバンク株式会社
Presently with Softbank Corporation

a) sharoron@kmd.keio.ac.jp

b) uhyo@kmd.keio.ac.jp

Live ではチャンネル登録者数が 1,000 人以上必要で、そもそもモバイル端末からの配信は現状行えない [2]。本研究では、モバイル端末からのリアルタイム配信の制約である処理能力をクラウドコンピューティングにより支援するプラットフォームを構築、また配信に関するリテラシーはアプリケーションで支援することで、配信の経験が少ない人やネットワークに関する知識がなくとも、表現の幅を狭めることなく配信を実現する。

2. 本研究の目的

本研究では BaaS のメカニズムを個人向けに活用することで、配信に関する環境やリテラシーをサービスやアプリケーション上でバックアップし、場所や環境を問わずリアルタイム配信を行うことのできるプラットフォームの構築、それを操作するモバイル端末に最適化したアプリケーションの開発を行った。本実装により、作成したプロトタイプを実際にユーザに使用してもらい、リアルタイム配信を体験してもらった。実際に配信を行った経験のある人に対しては、配信する際にできる表現の幅や、配信手法のあり方について評価を行った。

3. 関連研究

3.1 Broadcast as a Service

BaaS(Broadcast as a Service) とは「サービスとしての配信」であり、配信に必要な機能、例えばネットワーク、エンコード、デコードやストレージといった各機能をオンデマンド方式にて提供するサービス形態である [3]。ユーザが必要なおきのみ動作させることが可能であり、全ての処理や映像伝送が IP 通信で行われるため、ネットワークに接続されていれば場所を問わず利用することが可能である。また、各機能がエコシステムで動作するため高い拡張性があり、必要な機能だけの利用、または追加で必要な機能があれば組み込むことが用意である。さらに、配信に関する環境をサービスとして提供するため、基本的にはインターネット上でミキサー等の設定が可能である。そのため、カメラは現地で撮影する必要があるものの、編集に関してはオンラインで操作が可能である。

3.2 事例

現在、この BaaS は主に放送業界で活用されている。この仕組みを利用したサービスは

- Singular.Live[4]
- LTN[5]

がある。これらはいずれも放送局向けといったプロフェッショナル用途を想定しており、TV プログラムの生産やキュレーションに力を入れている。プロフェッショナル用途で使用される映像伝送プロトコルには NDI や SRT(Secure Reliable Transport) が使われており、テレビ局向けのカメ

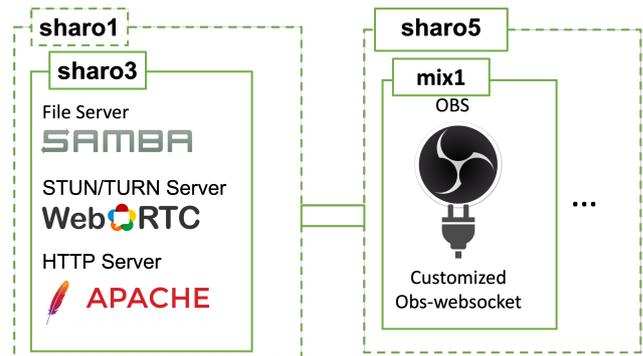


図 1 サーバ構成

Fig. 1 Server environment

ラでもこちらにサポートしているため使い勝手が良い。しかし大半がオプションであったり高額なライセンスが必要 [6] であり、個人では手を出し難い。また、プロフェッショナル用途のプロトコルは映像間の同期や映像クオリティのための細かい設定を行うことを前提としており、高度なネットワークの知識も必要である。

3.3 本研究への活用

本研究では BaaS のメカニズムを利用することで、リアルタイム配信に必要なコストの掛かる環境をサービスとして提供し、一つの配信環境プラットフォームとして構築した。

4. 実装手法

4.1 サーバ部

複数ユーザが利用する際に独立して動作させるため、ミキシングソフトウェアは仮想マシン上で動作させ、ユーザやセッションに応じて仮想マシンを複製する(以下、この複製したマシンを「ミキシングサーバ」と言う)。クライアントアプリケーションを提供する HTTP サーバや WebRTC のシグナリングを担当する TURN/STUN サーバ、ファイルサーバなどの複数ユーザが利用するサーバは共用サーバにて運用した。サーバ構成は図 1 に示す。

この時、アカウントやセッションの管理を行うためのサーバアプリケーションは JavaScript でプログラミングを行い、NodeJS で動作する(以下、これを「コントロールサーバ」と言う)。ミキシングサーバである OBS をコントロールするため、obs-websocket[7] を使用し、コントロールサーバを介してクライアントから OBS をコントロールする。

4.2 ネットワーク部

本研究の実験環境ではサーバは大学内ネットワークに設置、1Gbps の環境にて運用を行った。本研究の調査に使

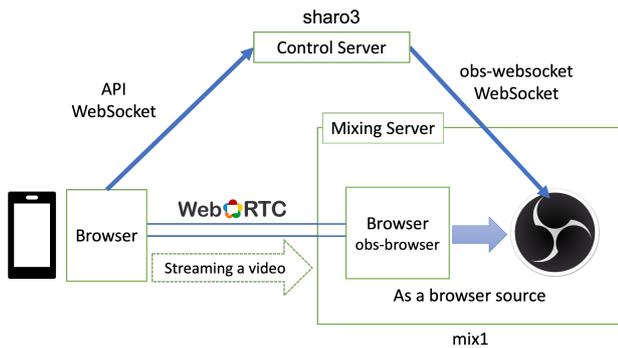


図 2 ネットワーク環境
Fig. 2 Network environment

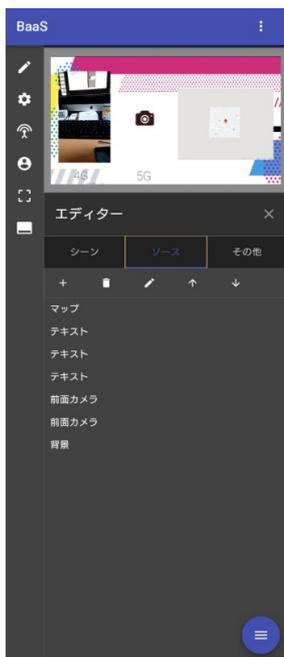


図 3 プロトタイプ
スマートフォン
Fig. 3 Prototype
Smartphone

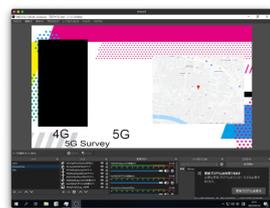


図 4 サーバ画面
Fig. 4 Server Screen

用したスマートフォンは iPhone 12 Pro 又は Sony Xperia Pro であり、キャリアは Softbank の 5G/4G にて行った。ネットワーク構成は図 2 に示す。クライアント-コントロールサーバ-ミキシングサーバ間はいずれも WebSocket にて常時接続を行い応答性を高めた。

4.3 ソフトウェア部

ミキシングソフトウェア上に表示されるプレビューと限りなく近いプレビューをクライアントアプリケーション上に表示することで、ユーザにとって直感的に操作できるようにした。アプリケーションとサーバ画面はそれぞれ図 3、図 4 に示す。また、操作手順は基本的に OBS に準拠しており、モバイルデバイスでも扱いやすくするため、UI を最適化し構築した。クライアントは HTML ベースのアプリ

表 1 検証の条件

Table 1 Condition of survey

日時	2020 年 11 月 14 日
端末	iPhone 12 Pro
エリア	秋葉原エリア
使用回線	Softbank 4G & 5G
VPN	L2TP+IPSec & IKEv2
アプリケーション	Periscope HD[8] NDICam[9]

表 2 検証の条件

Table 2 Condition of survey

日時	2020 年 11 月 12 日
端末	iPhone 12 Pro
エリア	秋葉原エリア
使用回線	Softbank 4G & 5G
アプリケーション	TVU[12]

ケーションで提供し、開発言語は JavaScript と TypeScript を採用した、

Babel により JavaScript を EC5 記法に変換したため、現在発売されているほとんどのスマートフォンにて利用できる。ライブラリ群として React と Material-UI を使用した。

5. 実証

5.1 事前検証

5.1.1 RTSP/NDI と VPN を使った手法

プロトタイプを用いず、従来の方でサーバにあるミキシングサーバへ映像を伝送した場合について検証を行った。VPN に接続し、RTSP により映像をミキシングサーバにある OBS へ映像を伝送を行う方法である。条件は表 1 の通りである。iOS では RTSP ビュアは多数あるものの、iPhone を RTSP サーバとして動作させるアプリケーションは限られており、VPN を通じて動作させた場合、通信が切断された際に OBS が自動的に再接続されないという問題が発生した。また、5G/4G エリアを跨ぐ場合、ネットワークは一度切断され、ミキシングサーバ側でリロードを行う必要がある。IKEv2 の再接続機能を用いて、RTSP のプロトコル上の通信を常時接続した状態にできないか試みたが、効果はなかった。NDI は OBS へプラグインを導入することにより、サポートされるプロトコルの一つである [10]。基本的にローカルネットワークで使用することが想定されており、モバイル回線からミキシングサーバへ入力する場合、VPN の接続が必須である。しかし、NDI は mDNS を用いてクライアントを識別しているため、Discovery Server という別ネットワークに存在するデバイスを認識するサーバが必要となる [11]。そのため、各ミキシングサーバをユーザ毎に独立して動かす場合、設定が非常に煩雑になることから採用を断念した。

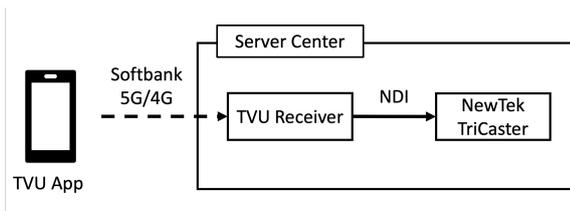


図 5 ネットワーク環境
Fig. 5 Network environment

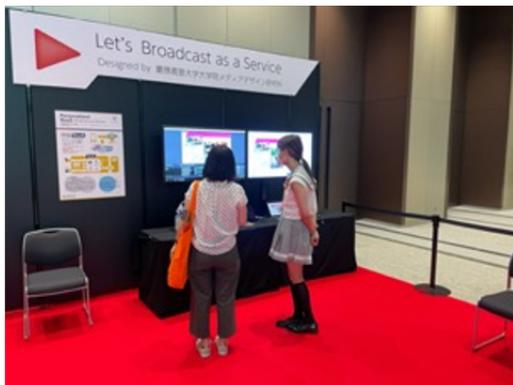


図 6 展示光景
Fig. 6 Exhibition

5.1.2 TVU を用いた手法

TVU は docomo や au などのキャリアネットワークを利用した無線映像伝送システムである。プロフェッショナル向けであり、専用機材のレンタルに 1 日数万円必要 [13] など費用が高額なため個人では扱いきれないが、比較として検証を行った。本実証ではサーバセンター内にある TVU レシーバに対して映像伝送を行っている。本実証実験のみ環境が異なり、図 5 のように構築を行った。結果は正常に映像を送信することはできたものの、4G/5G エリアを跨ぐ移動を行ったところ数秒の暗転が見られた。また使用したアプリケーションは積極的に HEVC/H.265 エンコードを行い通信量を削減するようになっており、1 時間を超える長時間使用を行ったところ、過負荷によるオーバーヒートやブロックノイズの発生した。

5.2 KMD Forum における実証

2021 年 7 月 3 日に東京ポートシティ竹芝オフィスタワーにて「ちょっと先のおもしろい未来 -Change Tomorrow-」[14] が開催された。その際に Softbank 株式会社の BaaS 展示コーナーにて、「個人向け BaaS の体験コーナー」として、本研究のプロトタイプを展示、実際に訪問者に対し図 6 のように配信を体験し、実証実験を行った。

この展示は、Softbank 株式会社の展示は放送局向けの BaaS の展示の隣で行い、プロ機材としての BaaS と個人向け配信としての BaaS のように対比して展示し、10 代未満から 60 代までの男女約 300 人が訪れた。

表 3 竹芝調査の条件

Table 3 Conditions of Takeshiba survey

日時	2021 年 7 月 20 日 11 時より 7 時間
場所	東京ポートシティ 竹芝オフィスタワー 1F
使用端末	Sony Xperia Pro 2 台
通信種別	Softbank 5G エリア [15]
コーデック	VP9 + opus
クライアントブラウザ	Chrome for Android 91.0.4472.164
サーバブラウザ	Chromium Embedded Framework 75.1.16+g16a67c4
カメラ解像度	1920x1080@30FPS
ビットレート	可変

5.2.1 配信の UX に関する調査

訪問者に対し実証実験を行い、被験者から以下の様な意見を得た。

- (1) 自宅で気合を入れて配信を行っていたのが、ブラウザ上で完結するのは気が楽でいい
- (2) スマートフォン 1 つで機材が全て整うのであれば、知人と一緒に配信を楽しむこともできそう
- (3) 複雑な配信もスマートフォンのみのできるのであれば、他の人と差別化した配信ができる

配信を行った経験の無い人からは「配信がどのように動作しているのかわからなかったが、視聴者として配信者によりインタラクティブなコミュニケーションが取れるのであれば、より面白いコンテンツが登場しそう」という意見が得られた。

5.2.2 技術的知見

本展示は午前 11 時から午後 6 時と長時間の展示であり、基本的にプロトタイプを動作させているスマートフォンは 1 台のみで、2 台の端末を交互に展示した。本調査の条件は表 3 の通りである。

結果として、1920x1080@30FPS の映像をサーバへ送信していたが、1 時間ほどで過負荷のためオーバーヒートし、解像度は 640x360 まで自動的にダウンスケールされ、またブロックノイズや映像の遅延が目立つようになった。また、2 時間半ほどでバッテリー残量がなくなり、バッテリーは急速に消耗した。.. ミキシングサーバから YouTube Live サーバに対しては安定して通信できており、2 時間 24 分の配信を連続して行った。

5.3 秋葉原における実証

本研究のプロトタイプでは、クライアントとサーバ間の映像伝送に WebRTC を用いた。この WebRTC は 2011 年に初版が発行された技術ではあるが、映像伝送のエンコードに VP8 や VP9, H.264 といったメジャーなものから

表 4 秋葉原調査の条件

Table 4 Conditions of Akihabara survey

日時	2021年7月20日 17時55分より18分間
調査エリア	秋葉原 中央通り沿い
使用端末	Sony Xperia Pro 2台
5G 端末の通信種別	Softbank 5G[15]
4G 端末の通信種別	Softbank 4G
コーデック	VP9 + opus
クライアントブラウザ	Chrome for Android 91.0.4472.164
サーバブラウザ	Chromium Embedded Framework 75.1.16+g16a67c4
カメラ解像度	1920x1080@30FPS
ビットレート	可変



図 7 秋葉原調査

Fig. 7 Akihabara survey

最新のコーデックである AV1 が使用できるとフル HD で 30FPS の映像が 1Mbps で伝送することができる [16] . 本調査では WebRTC で最もメジャーな VP9 を用いて長時間配信した際の通信品質を 4G , 5G 通信下で比較した . その際の条件は表 4 の通りである .

5G に接続されたスマートフォンと 4G のみに制限したスマートフォンの 2 台を並べ , WebRTC を用いて映像伝送を行い , 同時に RTMP で YouTube Live へ配信を行った [17] にてアーカイブの再生が可能である . . 結果 , 5G はスムーズに伝送されており , ノイズや大きな遅延は発生しなかった . 4G での配信ではパケットづまりが発生し , 映像に青みがかかるノイズの発生やフレームが落ちる現象が見られた . 実際に配信した画面を図 7 に示す .

転送量と解像度の推移を解析すると図 8 のようになった . 起動した直後は共に大きい解像度での映像伝送を試みたが , 5G では可変ビットレートで動作し , 安定した映像伝送ができていたのに対し , 4G では最低限の固定ビットレートに引き下げられた . 理論値は 5G 通信の上り 103Mbps , 4G 通信の上り 15Mbps であるが [18] , 現実的には 4G 通信下においてはビットレートの高くなる高解像度の映像伝送には向かない . 5G/4G エリアをまたいだ配信を行ったところ , 通信種別が切り替わるたびに WebRTC の通信は切断された . 現状は 5G/4G の切り替えはシームレスには行

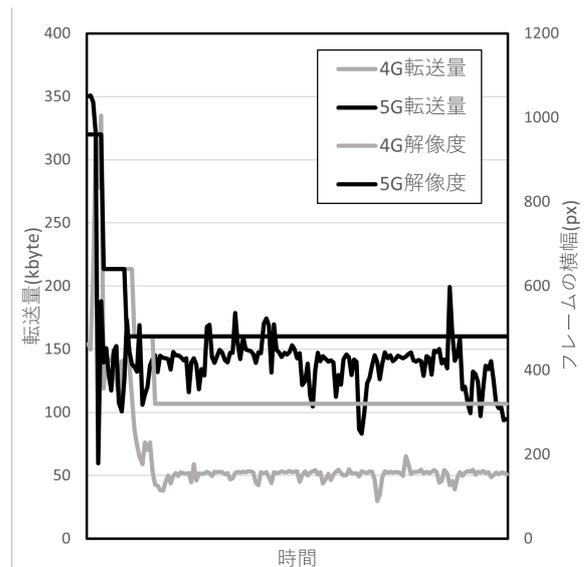


図 8 転送量と解像度の推移

Fig. 8 Bandwidth and resolution

かず , 切断されるたびに再接続が必要になることから , 配信スタイルに応じてはネットワーク種別の固定 , ないしは WebRTC の再接続の高速化を検討する必要がある .

6. 結論と今後の課題

調査結果により , 環境やリテラシーをプラットフォームやアプリケーション上でサポートすることで , 配信を行う際に必要な時間やコストを減らせた . さらに , 配信を行ったことのない人に対しても新たに始める動機になりうることを , アンケートから得ることができた . 5G 通信下では大容量通信を安定して行うことができ , 通信環境によるクオリティの制限を受けづらいため , WebRTC においてもクオリティの高い配信が可能であった . しかし , WebRTC は基本的に通信品質に応じてビットレートや解像度を可変で調整するため , よりクオリティの高い配信を行うためにはソフトウェア上のコントロールが必要である . また , 高ビットレートの映像をエンコードする場合 , ハードウェアの過負荷によるオーバーヒートによる処理能力の低下を引き起こす恐れがある . 安定した長時間に及ぶ映像配信は今後も調査が必要である . アマチュア向けやプロフェッショナル向け共に , 映像伝送をネットワーク上で行う事例が増えてきている . また写真や動画といったメディアはクラウド上で管理されることが多くなり , より一層インターネット上にてコンテンツの管理がしやすくなる . 今後は通信や配信環境に限らずコンテンツ面に関しても , 配信のあり方を変えていきたい .

参考文献

- [1] 三菱 UFJ リサーチ & コンサルティング: ライブ配信サービス (投げ銭等) の動向整理 , (オンライン) , 入手先 <https://www.caa.go.jp/policies/policy/consumer_policy/policy_coor>

(参照 2021-08-16)

- [2] YouTube: Create a live stream on mobile, (online), available from <https://support.google.com/youtube/answer/9228390> (accessed 2021-08-03).
- [3] bmc: Broadcast as a Service, (online), available from <https://www.bmcuk.tv/baas> (accessed 2021-08-16).
- [4] singular.live: Top Page, (online), available from <https://www.singular.live> (accessed 2021-08-18).
- [5] LTN Global Communications, I.: Top Page, (online), available from <https://ltnglobal.com> (accessed 2021-08-18).
- [6] NewTek: NDI HX, (online), available from <https://www.newtek.com/ndihx/products/upgrade/> (accessed 2021-08-04).
- [7] Palakis: obs-websocket - GitHub, GitHub (online), available from <https://github.com/Palakis/obs-websocket> (accessed 2021-08-18).
- [8] Team, A. D.: Periscope HD, App Store (online), available from <https://apps.apple.com/us/app/periscope-hd-h-264-rtsp-cam/id1095600218> (accessed 2021-08-18).
- [9] Gilbert, M.: NDICam, App Store (online), available from <https://apps.apple.com/jp/app/ndicam/id1074065391> (accessed 2021-08-18).
- [10] Palakis: obs-ndi - GitHub, GitHub (online), available from <https://github.com/Palakis/obs-ndi/releases> (accessed 2021-08-18).
- [11] NewTek: NDI Discovery And Registration, (online), available from <https://support.newtek.com/hc/en-us/articles/218109477-NDI-Discovery-and-Registration> (accessed 2021-08-17).
- [12] Corporation, T. N.: TVU Anywhere Pro, App Store (online), available from <https://apps.apple.com/jp/app/tvu-anywhere-pro/id883099579> (accessed 2021-08-18).
- [13] スターコミュニケーションズ株式会社: TVU One レンタル ,(オンライン), 入手先 <https://starcom.co.jp/tvu/price.html> (参照 2021-08-17)
- [14] ちょっと先のおもしろい未来実行委員会: トップページ ,(オンライン), 入手先 <https://change-tomorrow.tokyo> (参照 2021-08-18)
- [15] Softbank: サービスエリアマップ ,(オンライン), 入手先 <https://www.softbank.jp/mobile/network/area/map/> (参照 2021-08-16)
- [16] Google: AV1 Encoder for WebRTC, New Origin Trials, and More, (online), available from <https://blog.chromium.org/2021/03/chrome-90-beta-av1-encoder-for-webrtc.html> (accessed 2021-08-16).
- [17] 山下恭平: 5G4G 比較 in 秋葉原 ,(オンライン), 入手先 <https://www.youtube.com/watch?v=r74uxCcZw1M> (参照 2021-08-10)
- [18] Softbank: [SoftBank 5G] SoftBank 4G、SoftBank 3G との違いを教えてください。 ,(オンライン), 入手先 <https://www.softbank.jp/support/faq/view/10993> (参照 2021-08-18)

謝辞 Softbank 株式会社の皆様より実証実験のご協力を
いただいたことに深謝する。