

スマートフォンで操作する 双方向大型窓デジタルサイネージシステムの試作

山之上 卓¹ 中原光稀¹

概要: 従来の大型窓デジタルサイネージシステムを拡張し、「双方向大型窓デジタルサイネージシステム」を開発した。このシステムは、大型窓デジタルサイネージの画面の近くにいる不特定多数の人々が、自分の持つスマホでその画面の表示を制御することを可能にするものである。このシステムは、大型画面と不特定多数の人との間を双方向に情報が流れることで、新しいエンターテインメントとして利用できる可能性がある。このシステムの利用例として、窓に大きく表示されたゲーム画面を、その近くにいる不特定多数の人々が、それぞれが所有するスマートフォンを使って制御することが可能なゲームを開発することができた。

キーワード: 双方向, デジタルサイネージ, スマートフォン

Experimental Development of an Interactive Big Window Digital Signage System Controlled by Smartphones

Takashi Yamanoue^{†1} Mitsuki Nakahara^{†1}

Abstract: This manuscript describes experimental development of an interactive big window digital signage system, which can be controlled by smartphones. This system is an extended version of the big window signage system, which was reported previously. This system enables unspecified people, who are near the screen of the signage, to control the picture of the screen. This system may be used as a new entertainment. We could develop an interactive game which enables an unspecified person to play by controlling its scene using their smartphones.

Keywords: Interaction, Digital signage, Smartphone

1. はじめに

人通りが多い場所で多くの人々に情報を伝える手段の一つとして、デジタルサイネージ[1]が良く利用されている。デジタルサイネージは内容の更新が容易であり、動画の表示も可能な場合もある。我々は、プロジェクションマッピングや大型LED表示装置と比べて、安価に気軽に設置可能なデジタルサイネージとして、建物の室内から窓に映像を投影し、窓全体に光を通すスクリーンを張ることで窓全体を使う「大型窓サイネージ」システムを開発している[2]。ここで、横長の大きな窓全体に、明るい映像を投影するため、複数のプロジェクタを使う方法を採用した。

しかし、単に映像を流すだけでは情報の流れが画面から人への一方通行であり、他の多くの大型表示媒体と変わらず、それらと比べて大きな注目を集める可能性は低い。

今回、従来の大型窓サイネージシステムを拡張し、「双方向大型窓デジタルサイネージシステム」(図1)を開発した。このシステムは、そのシステムの画面の近くにいる不特定多数の人々が、自分の持つスマホでその画面を制御可能にするものである。このシステムは、大型画面と不特定多数の人との間を双方向に情報が流れることで、新しいエ

ンターテインメントとして利用できる可能性がある。このシステムの利用例として、窓に大きく表示されたゲーム画面を、その近くにいる不特定多数の人々が、それぞれが所有するスマートフォンを使って制御することが可能なゲームを開発することができた。



図1 双方向大型窓デジタルサイネージシステム

¹ 福山大学
Fukuyama University
2

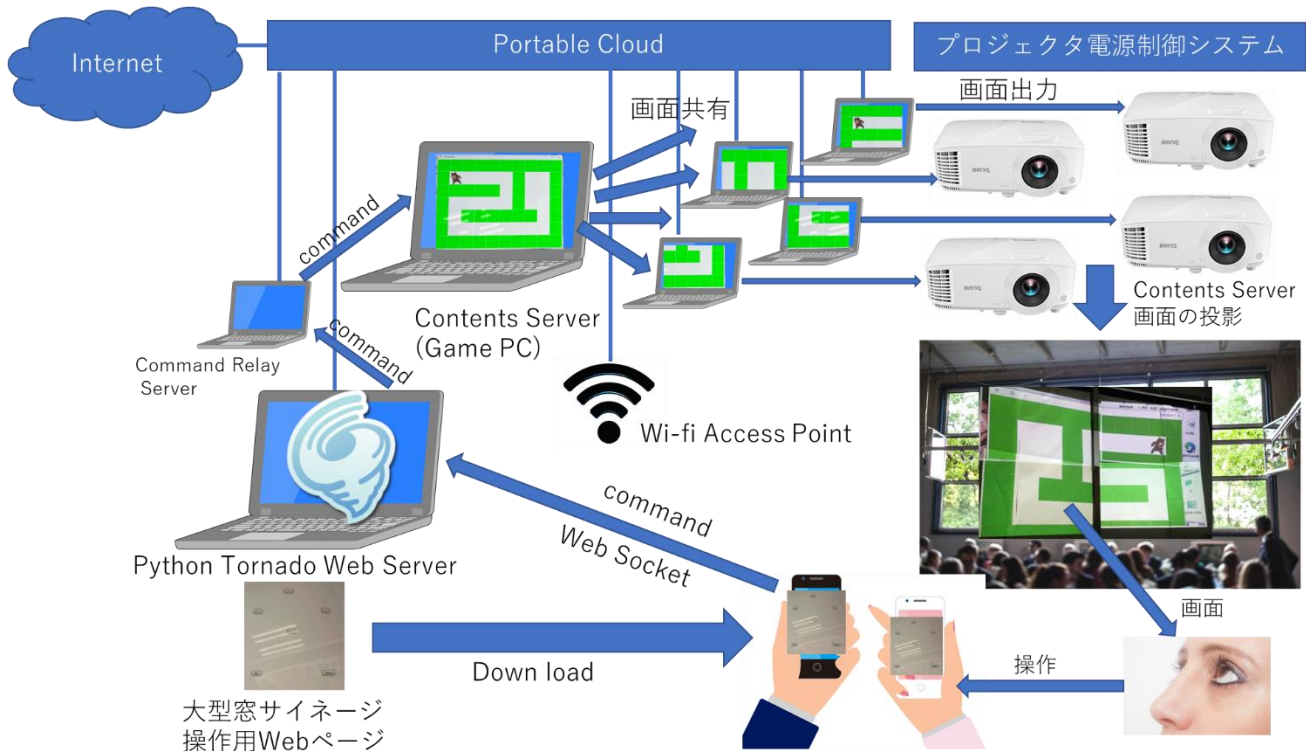


図 2 システム概要

2. システム概要

本システムは、Portable Cloud[4]を利用して作成された大型窓サイネージシステムに不特定多数の利用者との双方向性機能を付け加えたものである。図 2 にシステム概要を示す。

サイネージシステムの近くにいる不特定多数の利用者のスマホにその内容を操作する Web ページを表示され、利用者がその Web ページのボタンをクリックすることにより、クリックされたボタンに対応したコマンドがコンテンツを持っている Contents Server に送られる。それにより、不特定多数の利用者によるコンテンツの制御が行われ、双方向性のあるコンテンツが実現できる。

Contents Server は Portable Cloud の一台のクライアントパソコンとして、Portable Cloud に接続され、その画面が、Portable Cloud の画面共有システム (Distributed Web Screen Sharing system, DWSS) [3]により、ポータブルクラウドに接続された複数の表示用パソコンに送信される。DWSS は、授業や会議などで、講師のパソコン画面を、多数の受講者のパソコンやスマホに転送するために作成されたが、ここでは、Contents Server の画面を多数の表示用パソコンに送信するために利用している。

表示用パソコンでは受信した画像の一部を表示し、その表示がプロジェクタによって窓に貼られた障子紙に投影さ

れる。複数の表示用パソコンの画面が窓の障子紙に投影されて、全体として大きな 1 枚の画面になる。複数の表示用パソコンとプロジェクタを利用することにより、大きな画面に明るい表示を行うことを可能としている。

不特定多数の利用者のスマホに、コンテンツ操作 Web ページを表示するため、Portable Cloud のキャプティブポータルを利用している。大型窓サイネージシステム付近に表示された、このシステムの Wi-Fi のアクセスポイントの SSID とパスフレーズなどの情報を入力すると、それぞれのスマホに、操作 Web ページが表示される(一部実装中)。

操作 Web ページは、Python Tornado Web Server を利用して利用者のスマホに表示されるようになっており、利用者が Web ページのボタンをタップすると、そのボタンに対応したコマンドが Web Socket を使って、Python Tornado Web Server に送信され、そのコマンドは、Command Relay Server を通じて、Contents Server に送信される。このコマンドを受け取った Contents Server は、コマンドに従った処理を行う。

プロジェクタの電源管理を行うため、Wiki IoT システムを利用した制御が行われる。

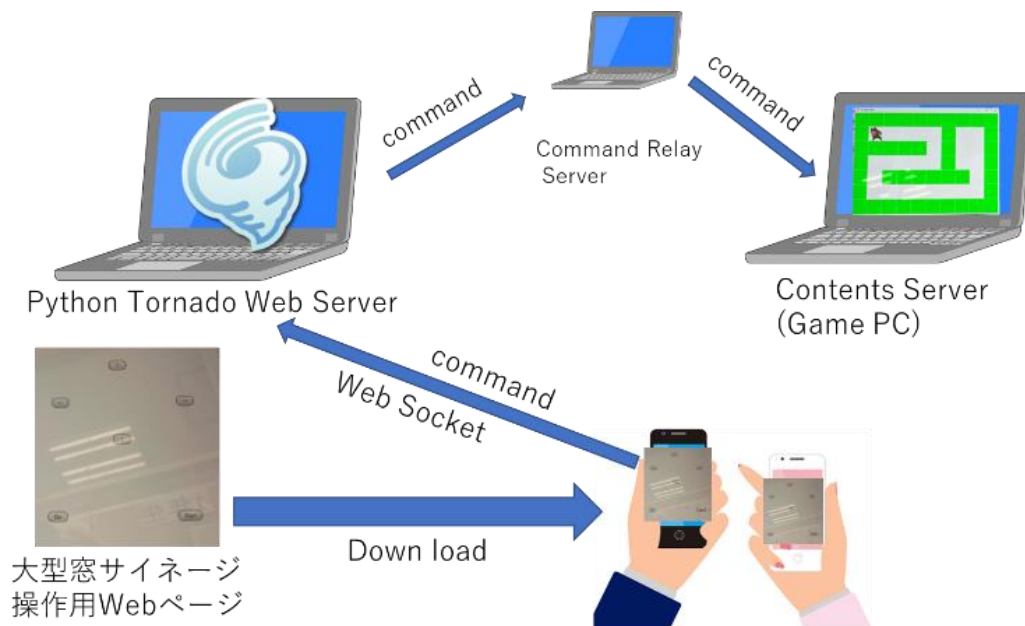


図 3 本システムで新たに開発した部分

3. システムの詳細

本システムは、既に紹介している大型窓サイネージシステムに、Python Tornado Web Server を使って作成された操作 Web ページ表示システム、コマンド中継サーバ、Contents Server を加えて構成している。図 3 に、本システムで新たに開発した部分を示す。以下にこれらを説明する。

3.1 操作 Web ページ表示システム

本システムでは、大型窓サイネージの近くにいる不特定多数の利用者のスマートフォンに、本システムのコンテンツの操作 Web ページを表示し、その Web ページを操作することで、コンテンツを実時間で制御することを可能にしている。

利用者は大型窓サイネージシステムの近くに表示されている Wi-Fi アクセスポイントの SSID とパスワードをスマートフォンに入力し、指定された URL の Web ページを開くことで、操作 Web ページを自分のスマートフォンに表示することができる。なお、ポータブルクラウドのキャプティブポータル機能をうまく使うことで、URL を入力しなくても、その Wi-Fi アクセスポイントに接続するだけで、直接スマートフォンの Web ブラウザにゲームコントローラを表示させることも可能である。

この操作 Web ページは Python の Tornado を使うことで実現している。Tornado は Python で作られた Web フレームワークであり、非同期通信を行う Web Socket ライブラリを利用することができる。

これを使って実時間で Web ページが双方向に動的に変化する Web サイトを実現することができる。

操作 Web ページのボタンをタップすると、そのボタンに結びつけられたコマンドが HTML 5 の WebSocket を通じて、Tornado のサーバに送信される。

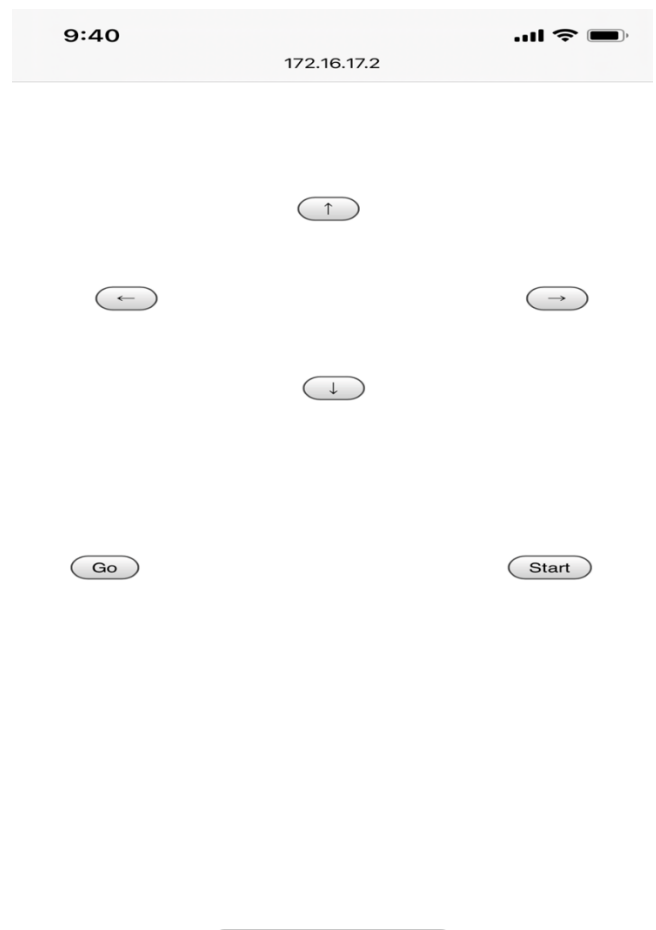


図 4 操作 Web ページの例

図 4 に、操作用 Web ページの例を示す。たとえば、このページの上矢印のついたボタンをタップすると、“up”という文字列がコマンドとして Tornado の Web サーバに、Web Socket を使って送信される。

Tornado の Web サーバは、スマホの Web ページからコマンドを受け取ると、このコマンドをそのままコマンド中継サーバに転送する。

3.2 コマンド中継サーバ

コマンド中継サーバは Tornado のサーバで受け取られた、スマートフォンから送信されたコマンドを Contents Server に中継する。

コマンド中継サーバは TCP を使ったチャットサーバであり、Tornado で作成した Web サーバと Contents Server はコマンド中継サーバのクライアントである。片方から入力されたコマンドがそのままもう片方に送信される。このことによって、スマートフォンの操作用 Web ページの操作で入力されたコマンドが Contents Server に転送される。

3.3 Contents Server

Contents Server は、大型窓サイネージに表示される Contents(画像)の元画像を作る Server である。また、Contents Server は IP ネットワークから TCP でコマンドを受け取り、そのコマンドに従って画像を制御することができる。

Contents Server は、TCP でコマンドの文字列を受け取ることが可能で、そのコマンドを解釈実行できるものであれば、どんなコンピュータ上のどんな OS やプログラミング言語で実装されたものでも利用可能であるとする。

今回、Contents Server として、簡単な迷路を探索するゲームを Python を使って実装した。このゲームを開始すると、初期画面が表示される。このとき、スマホに表示された操作用 Web ページの Start ボタンをタップすると、最初

の迷路が表示される。迷路が表示された状態で操作用 Web ページの上下左右の矢印ボタンをタップすることにより、迷路の上のキャラクタが迷路の中を移動する。キャラクタが迷路内のすべての移動可能箇所(27 か所)を移動すると、その迷路内の移動を終了する。その後、Go ボタンをタップすると、次の迷路が表示される。このようにして3つの迷路すべてを移動するとゲームが終了する。

4. 大型スクリーンを使った実験

本システムを使って、実際にプロジェクトを4つ使う大型スクリーンにゲーム画面を表示し、Portable Cloud に接続されたスマートフォン（ここではスマートフォンの代わりに、タブレット端末）に表示された制御用 Web ページを使って、ゲームを遊ぶことができた。

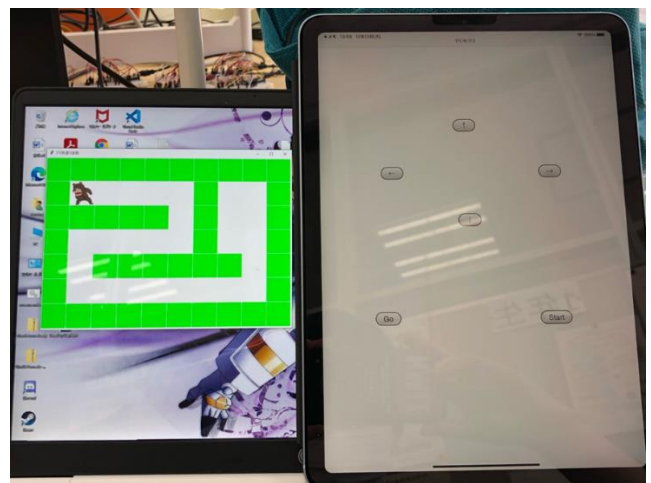


図 5 大型スクリーンを使った実験に使用した Contents Server(左)とタブレット端末の画面(右)

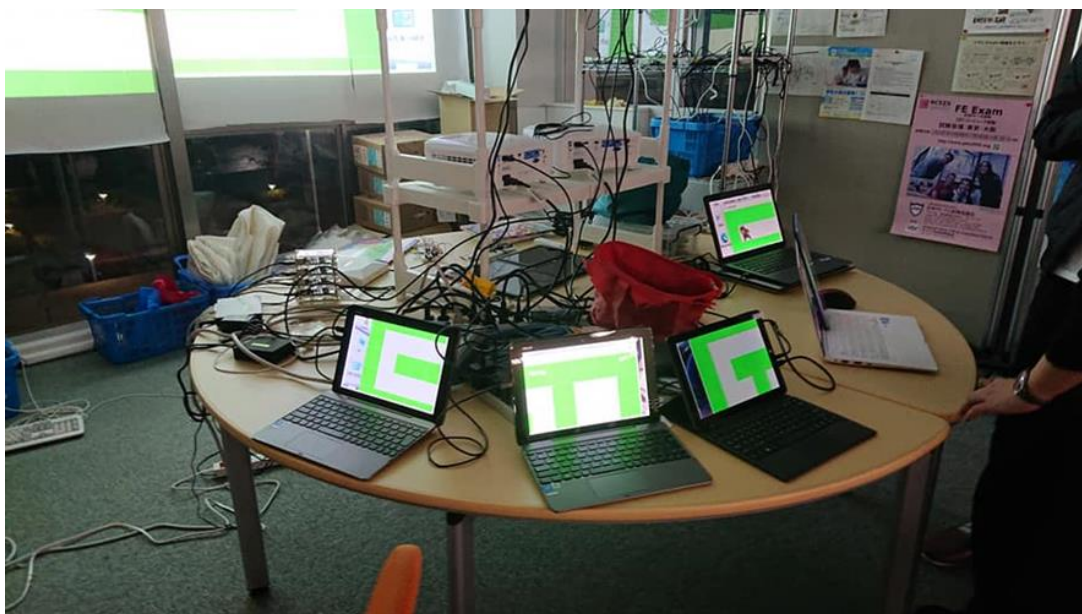


図 6 大型スクリーンを使った実験に使用した表示用パソコンの画面

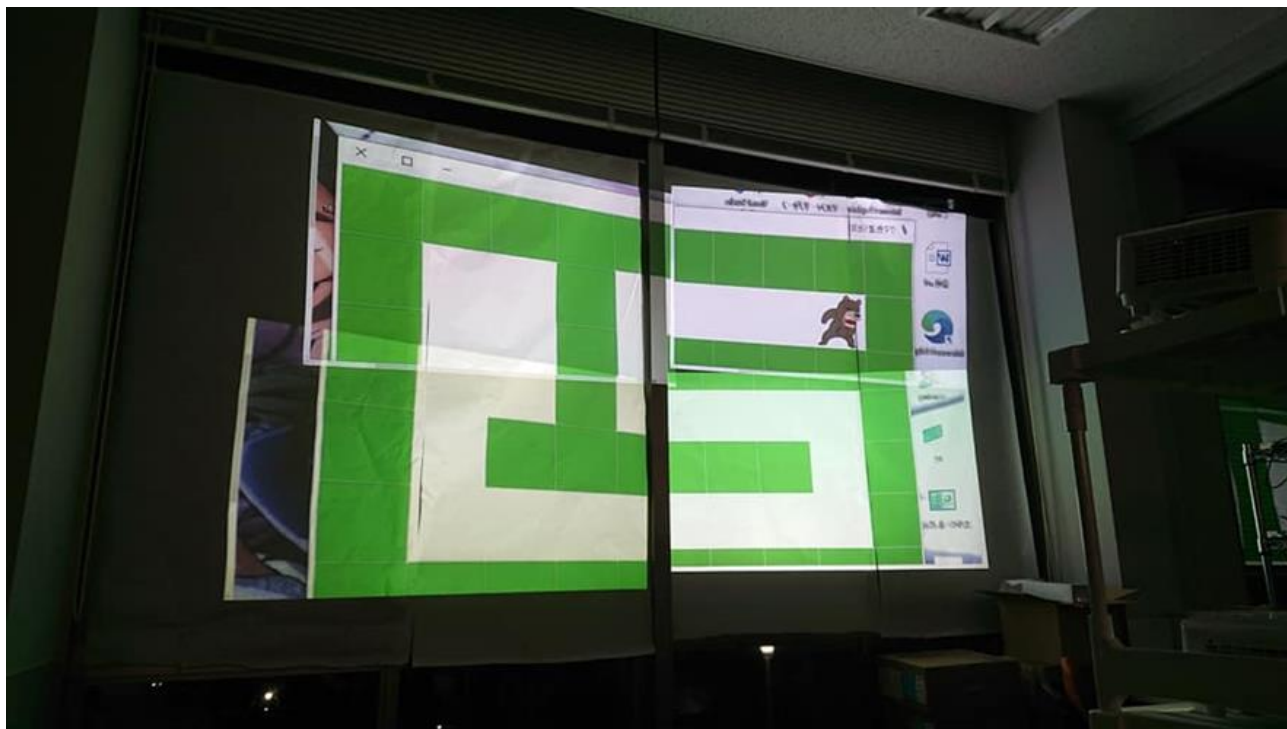


図 7 大型スクリーンを使った実験の大型スクリーンに表示された画面

5. 関連研究

5.1 デジタルサイネージシステムにおける画面デザインの研究

遠藤はデジタルサイネージの画面デザインや評価方法に適用できる内容、画面デザインの違いによる利用者の評価や反応、的確に情報を伝達しやすい画面デザインの特徴などについて評価実験を行っている[6].

デザイン・パターンの検討においては、ピクトグラムのデザイン・パターンが画面デザインとして適しており、写真については、被験者の評価決定が難しいと推測されるなどの結果が示されている。

我々のシステムのコンテンツについては、これらの結果が有用に使える可能性がある。

5.2 大型サイネージシステムを使ったゲームシステムの開発

福山大学の中川は、ポータルクラウドを使って作られた大型窓サイネージシステムと、Tornadoを使って作った Web 上のゲームシステムを組み合わせ、大型サイネージシステムの画面を使って遊ぶことができるゲームシステムを作っている[7].

本システムは、中川のゲームシステムのゲームを実行する部分とコントローラ部分を分離したものである。このことにより、同じコマンドを入力できるゲームであれば、様々なゲームを汎用的に実行することができる。

5.3 大画面描画可能な PIKAPIKA 作成支援ツール

福山大学の村尾は PIKAPIKA 名づけられた大画面と人間のジェスチャーによるインタラクティブな描画システムの、描画範囲の広さとユーザビリティについて検証している [8].

PIKAPIKA も大画面表示を行った双方向型システムであるが、入力装置に Kinect を使い、人のジェスチャーで入力を行っている。これに対し、本システムの入力装置では利用者のスマートフォンに表示されたコントローラで入力を行っている。

5.4 巨大壁面スクリーンを利用した一斉遊学支援システム

柳澤と梶本は教室の壁全面をスクリーンとして、その一部を電子黒板化し、大学講義などの一斉授業を支援するシステムについて述べている[9]. このシステムと我々のシステムは、その利用目的が異なることや、利用環境が異なるなどの違いがあるが、大きな画面を用いた双方向性のあるシステムということ類似している。柳澤と梶本のシステムではメインスクリーン 1 とサブスクリーン 2 の合計 3 つのスクリーンを 3 台のプロジェクタを使って一つの壁に投影しているが通常のプロジェクタでは明度が足りない問題が指摘されている。これに対して我々のシステムではプロジェクタを複数利用して 1 つの画面に合成することで、その問題の解決を図っている他、昼間の投影は仮定していない。また、柳澤と梶本のシステムでは、利用者の入力に

については電子黒板の機能が利用されているのに対して、本システムは不特定多数の利用者のスマートフォンを利用している。

5.5 MIT のビル窓を使ったテトリス

マサチューセッツ工科大学ではビルの窓を使った大画面テトリスを作成している[10]。このテトリスはビルの、9×19個の窓でドットを使い、照明をコントロールすることで遊んでいるようにみせている。

MIT のテトリスのコントローラは専用のゲームコントローラを使っており、不特定多数の利用者のスマホによって操作できるものではない。

6. おわりに

大型窓サイネージシステムの表示を、不特定多数の利用者が、それぞれの利用者の所持するスマートフォンによって、制御可能にするシステムと、そのシステムを使って、大型窓サイネージを使った簡単なゲームシステムを開発できたことについて述べた。

今後はこのシステムを人通りが多い街中などで使い、その効果の評価などを行いたい。

本システムは不特定多数の利用者が利用する。このため、情報セキュリティに関して特別な配慮が必要になる。この問題については今後の課題とする。

謝辞 大型幅広デジタルサイネージシステム開発にご協力いただいた福山大学人間文化学部長の田中教授と福山大学の学生諸君に感謝します。

参考文献

- [1] 中村 伊知哉, 「デジタルサイネージの動向」, 情報管理 55(12), 891-898, 2013
- [2] 山之上卓, 田中 始男, 原 裕樹, 中川 裕貴, 矢嶋 辰伍, 増野 治哉, 森重 太智, 「簡便な大型幅広デジタルサイネージシステムとその自動運用システム」, 情報処理学会 研究報告インターネットと運用技術 (IOT), 2019-IOT-47(10), 1-8 (2019-09-20).
- [3] 山之上 卓, 小荒田 裕理, Katagiri Taiki, 小田 謙太郎, 下園 幸一, 「HTML5 技術を利用した授業や会議向けデスクトップ画面実時間配信システムとその管理システムの試作」, 研究報告インターネットと運用技術 (IOT), 2014-IOT-26(11), 1-8 (2014-06-21)
- [4] Takashi Yamanoue, Soshi Tetaka, Kentaro Oda, Kochi Shimozono, "Portable Cloud Computing System - A System which Makes Everywhere an ICT Enhanced Classroom", Proceedings of the 42th annual ACM SIGUCCS conference on User services, Salt Lake City, Utah, US, 4-7 Nov., 2014.
- [5] 山之上卓, 樋高想士, 小林幸司, 小荒田裕理, 片桐太樹, 小田謙太郎, 下園幸一, 「ポータブルクラウドの試作」, 情報処理学会研究報告, Vol.2013-IOT-22, No.12, (2013).
- [6] 遠藤潤一, 茂登山清文, 中村 純, 「情報提供を目的としたデジタルサイネージの画面デザイン評価」, 図学研究 第43巻4号 (通巻126号) 平成21年12月, pp.23-30, 2009.
- [7] 中川裕貴, 「大型サイネージシステムを使ったゲームシステ

ムの開発」, 福山大学工学部情報工学科 2019 年度卒業論文, 2020.

- [8] 村尾佳晃, 「大画面描画可能な PIKAPIKA 作成視点ツールの提案」, 福山大学工学部情報工学科平成 27 年度論文, 2016.
- [9] 柳沢 昌義, 梶本奈都未, 「巨大壁面スクリーンを利用した一斉遊行支援システムの開発」, 日本教育工学会第 27 回大会講演論文集, pp.505-506, 2011.
- [10] MIT Hack Gallery, "Tetris on the Green Building", http://hacks.mit.edu/Hacks/by_year/2012/tetris/, 2012. As of 2021/1/20.