

移動体データ形式「OGC MF-JSON形式」の取組みについて

(2) OGC MF-JSON に関連した日立のケーススタディと周辺技術のご紹介

2022/7/12 (株)日立製作所 研究開発グループ 先端AIイノベーションセンタ **刑部 好弘**



氏 名

刑部 好弘 (Yoshihiro Osakabe)

所属

(株)日立製作所 研究開発グループ デジタルサービス研究統括本部 先端AIイノベーションセンタ 知能情報研究部

担当業務

- ① 材料科学向けの人工知能技術 (Materials Informatics)
- ② 地理空間情報向けの人工知能技術 (GeoAI)

▶ 人流解析やAIS解析などの移動体データ分析に従事。 **OGCには2019から参加。**

バックグラウンド

電子工学(修士)+情報科学(博士)

⇒ 超伝導工学と非ノイマン型アーキテクチャの研究

趣味・その他

剣道歴27年(会社の剣道部にも所属), Jazz鑑賞など

本日ご紹介するコンテンツ



MF-JSON形式データを扱うツールのご紹介



移動体データ向けのPostGIS拡張

"可視化"の一例として日立の事例をご紹介



フィジカルディスタンス誘発システム



MobilityDBのご紹介



OGC MF-JSONをサポートするツールが増えています



Python: MovingPandas (GeoPandas拡張)

— Free and Open Source GIS Ramblings

Movement data in GIS / Projects / Publications / About

By underdark

Leave a comment

GIS



First time, we talked about the OGC Moving Features standard in a post from 2017. Back then, we looked at the proposed standard way to encode trajectories in CSV and discussed its issues Since then, the Moving Features working group at OGC has not been idle. Besides the CSV and XML encodings, they have designed a new JSON encoding that addresses many of the downsides of the previous two. You can read more about this in our 2020 preprint "From Simple Features to Moving Features and Beyond".

Basically Moving Features JSON (MF-JSON) is heavily inspired by GeoJSON and it comes with a bunch of mandatory and optional key/value pairs. There is support for static properties as well as dynamic temporal properties and, of course, temporal geometries (yes geometries, not just

I think this format may have an actual chance of gaining more widespread adoption



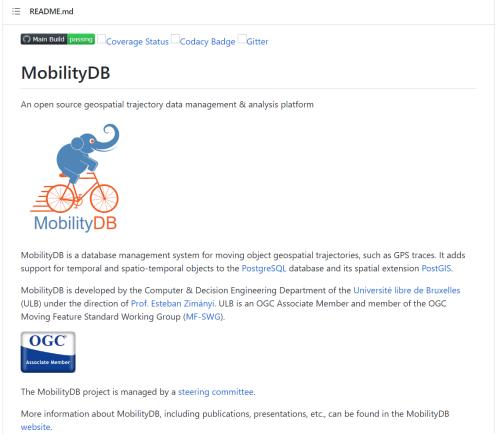
Figure 2. ISON Objects in the Moving Features ISON forma



Image source: http://www.opengis.net/doc/BP/mf-json/1.0

Inspired by Pandas.read_csv() and GeoPandas.read_file(), I've started implementing a read_mf_json() function in MovingPandas. So far, it supports basic MovingFeature JSONs with

PostgreSQL: MobilityDB (PostGIS拡張)



MobilityDBの概要



- ✓ 移動体データ向けのデータベース技術
 - 移動体データ向けのデータ型を提供
 - 移動体データ分析向けの組込み関数なども提供
 - SQLのクエリオプティマイザーも独自実装しているので高速
- ✓ PostgreSQLとPostGISの拡張機能なので、既存のエコシステムと共存する
- ✓ OGC標準に準拠、MF-JSONはもちろん、その他のデータ形式も積極的にサポート
 - 開発チームの一員であるMuhmoud Sakr氏は現在OGC Moving Features SWG のco-chairとなっており、2019年頃からMF-JSON仕様策定にも参画している
- ✓ Université Libre de Bruxellesのメンバーにより開発されている
 - OSSとしてPostgreSQLライセンスのもと公開されており、利用しやすい

なぜPostGISでは不十分なのか?



- ✓ Trajectory自体はLinestringによって表現できるが、時刻の情報は持たせられない
- ✓ そこで通常、PostGISでは「LinestringM」を使って、Measure(M)として時刻を入れる
 - ➤ ただしMeasure(M)は時刻の入力を想定しておらず、Mでソートするような機能がない。
 - このことにより、2分探索のようなアルゴリズムの実装が難しくなっている。
- ✓ さらに、temporal point, temporal numbers, temporal Booleans, temporal strings等、時系列データを想定した各種データ型も扱いたい

移動体データに特化したデータ型が必要 → MobilityDB

moid	tripid	tstart	xstart	ystart
1	2	2007-05-28T08:36:47	13.43593	52.41721
1	2	2007-05-28T08:36:49	13.43605	52.41723
1	2	2007-05-28T08:36:51	13.43628	52.41727
1	2	2007-05-28T08:36:53	13.43652	52.4173
1	2	2007-05-28T08:36:55	13.43676	52.41734
1	2	2007-05-28T08:36:57	13.437	52.41737



moid integer	tripid integer	astext text						
6	163	[POINT (2997192.88890412 5839689.91506735)@2007-05-28 06:00:00.001+00, POI						
6	165	[POINT (2985654.50641456 5848965.14626724)@2007-05-28 16:09:12.824+00, POI						
8	235	[POINT (3010311.09650771 5836055.09743228)@2007-05-28 07:19:01.864+00, POI						
8	237	[POINT(2997958.79103681 5837131.44898043)@2007-05-28 16:05:50.982+00, POI						
8	241	[POINT (2997958.79103681 5837131.44898043)@2007-05-29 17:11:03.19+00, POIN						
8	247	[POINT (3010311.09650771 5836055.09743228)@2007-05-30 07:02:57.848+00, POI						

MobilityDB内部では、Trip単位でデータ格納 形式は[POINT(x_y)@Time, …]

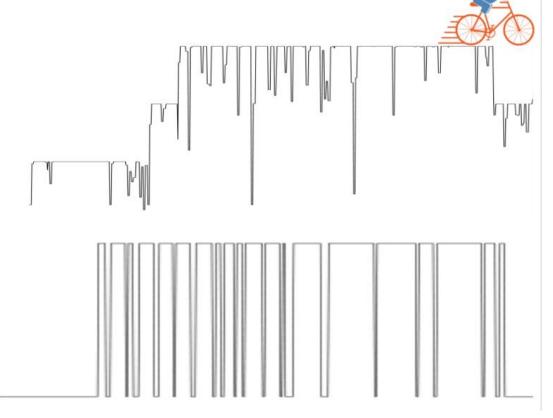
MobilityDBの想定ユースケース



Trip単位での計算・集計が容易に実行できる

tfloat: speed(Trip)

tbool: speed(Trip) > 90



MobilityDBの想定ユースケース



Trip単位ではない同時多発的事象も同様に扱い、集計することができる

Instant: UK road accidents 2012-14

https://www.kaggle.com/daveianhickey/2000-16-traffic-flow-england-scotland-wales

InstantSet: foursquare check-ins

https://support.foursquare.com/



MobilityDBの主要コンポーネント



- ✓ Time types
- ✓ Temporal types
- ✓ Query functions
- ✓ GiST, SP-GiST indexes
- ✓ Aggregation functions

詳しい説明は公式ドキュメントをご参照ください。 各種カンファレンスでの発表資料・動画など豊富です。

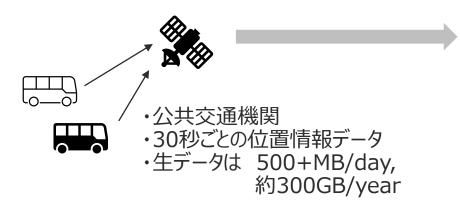


https://mobilitydb.com/publications.html

MobilityDBのデータ保存効率



MobilityDB開発チームによる実験





PostGIS → 100億レコード/day

44384	2015-04-06 06:38:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:38:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:39:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:39:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:40:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:40:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:41:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:41:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:42:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:42:30	POINT(37.3826816 55.7937783)

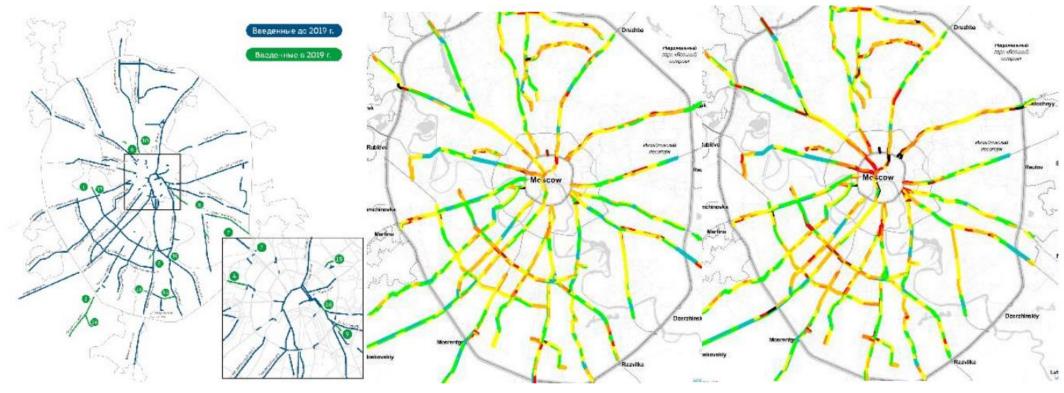
MobilityDB → 1万5,000レコード/day

44384 [POINT(37.3826816 55.7937783)@2015-04-06 06.38:00+03, POINT(37.3826816 55.7937783)@2015-04-06 0_4399 [POINT(37.6126166 55.7274032)@2015-04-06 07:14:29+03, POINT(37.6118683 55.7274732)@2015-04-06 0_4399 [POINT(37.6127783 55.7265099)@2015-04-06 05.32:14+03, POINT(37.6127783 55.7265099)@2015-04-06 0_5:32:14+03, POINT(37.607475 55.71504)@2015-04-06 0_5:35:1760, POINT(37.6078283 55.7158566)@2015-04-06 05:35:17+03, POINT(37.607475 55.71504)@2015-04-06 05:35:1760, POINT(37.6124233 55.7264416)@2015-04-06 05:07:57+03, POINT(37.6124233 55.7264416)@2015-04-06 0_5:07:57+03, POINT(37.60777 55.7153983)@2015-04-06 04:47:167762 [POINT(37.6093483 55.7190449)@2015-04-06 16:58:07+03, POINT(37.6094966 55.7188982)@2015-04-06 1-67762 [POINT(37.609216 55.719099)@2015-04-06 10-447:30+03, POINT(37.609215 57.7190516)@2015-04-06 10-47762 [POINT(37.609266 55.729099)@2015-04-06 10-447:30+03, POINT(37.609215 57.7190516)@2015-04-06 10-47762 [POINT(37.609266 55.729099)@2015-04-06 10-447:30+03, POINT(37.60921 55.7190516)@2015-04-06 10-47762 [POINT(37.609266 55.729099)@2015-04-06 10-471:30+03, POINT(37.60921 55.7190516)@2015-04-06 10-47762 [POINT(37.609266 55.729099)@2015-04-06 10-471:30+03, POINT(37.60921 55.7190516)@2015-04-06 10-471:30+03, POINT(37.60921 55.7190516)@2015

5MB/day (2GB/year)

Data analysis – velocity maps

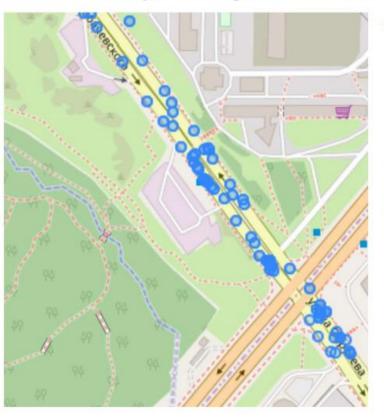
Moscow bus lanes







Velocity map - PostGIS



```
WITH segments AS (
     SELECT
'LINESTRING(37.44654993 55.72910825,...,
            37.44820638 55.72529983) '::geometry
       AS line
SELECT
   gl uuid, t, gl point, ST X(gl point) as lon,
ST Y(gl point) as lat, route,
    ST DistanceSphere(gl point, line) AS r FROM
   tracks tr, segments s
   WHERE t>'7:00' AND t< '9:00'
   AND ST DistanceSphere (gl point, line) < 20
   ORDER BY gl uuid
```

We have got a set of points, we make tracks from them and choose those that go in right direction.





Velocity map - MobilityDB



We just have got a velocity.

```
WITH segments as (SELECT
 ST Buffer (
  'LINESTRING (37.44654993 55.72910825, ..., 37.44820638
55.72529983) ':: geography,
  20)::geometry as shape,
 degrees (ST Azimuth (
      'POINT(37.44654993 55.72910825)'::geometry,
      'POINT(37.44820638 37.44820638)'::geometry)) as azimuth),
pbox as (
SELECT atPeriod(tline, period '[2015-04-06 07:00,2015-04-06 09:00]') AS
tline
     FROM mtracks),
lines as (SELECT unnest(sequences(atGeometry(tline, shape))) as track
                     FROM pbox, segments
                     WHERE intersects (shape, tline))
SELECT
     ST Length (getValues (track) :: geography) /
     extract('epoch' from (endTimestamp(track)-startTimestamp(track)))
     as v
FROM lines, segments WHERE
ABS (degrees (ST Azimuth (startValue (track), endValue (track))) -azimuth) < 60
```

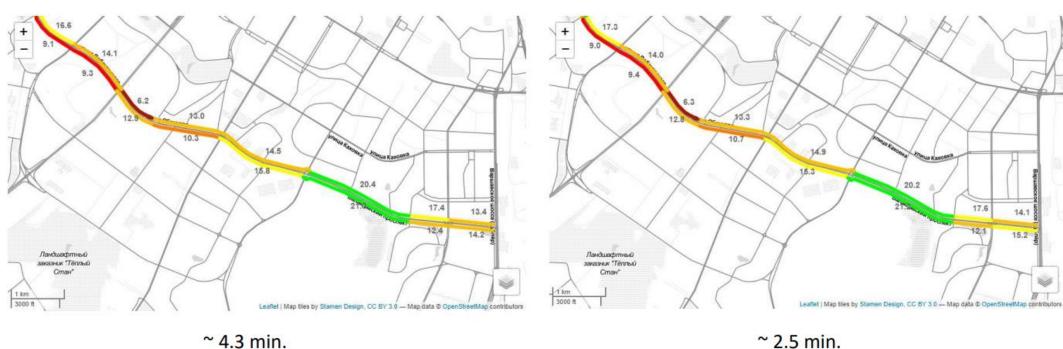




Velocity map calculation - comparison



MobilityDB



~ 4.3 min.

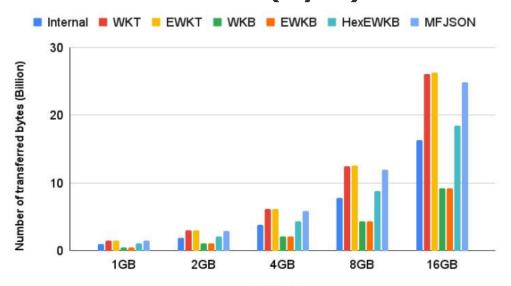


MobilityDBのベンチマーク: データ転送





転送したデータ量(byte)の比較



転送時間の比較

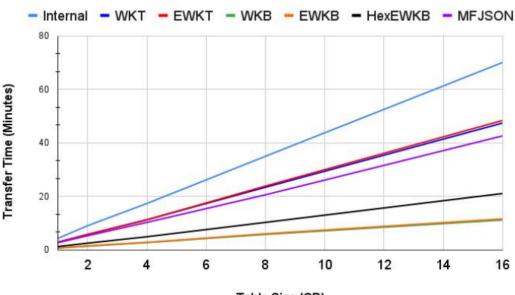


Table Size

Table Size (GB)

移動体データの可視化: Pythonとの連携



















可視化: HoloViz

MobilityDB-python

MobilityDB-python is a database adapter to a asyncpg adapters for PostgreSQL and uses the

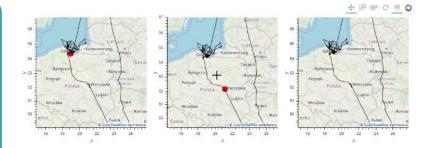
Install

pip install python-mobilitydb

Requirements

- Python >= 3.0
- MobilityDB

- ✓ MobilityDBチームが開発するOSS
- ✓ psycopg2とasyncpgを使って MobilityDB on PostgreSQLから データをクエリできる。
 - SQLAlchemyもサポート予定
- ✓ MovingPandasを使えばさらに便利に
- ✓ Django等のWebアプリ開発も容易に





出典:

MobilityDB-python | https://github.com/MobilityDB/MobilityDB-python/MovingPandas | https://anitagraser.github.io/movingpandas/HoloViz | https://holoviz.org/

ショートサマリ: MobilityDB



- ✓ 移動体データ向けのPostgreSQL/PostGIS拡張機能
 - OGCの活動に参加しているチームが開発するOSS
 - 他のエコシステムとの連携も積極的に進めている印象
 - FOSS4GやPgConfにも参加しアウトリーチ活動にも積極的
- ✓ 可視化に特化しているわけではないが、可視化する材料を手早く用意するには便利
 - MF-JSONを直接可視化するならSTINUUM (https://github.com/aistairc/mf-cesium)
 - Pythonを経由するならば、MovingPandasとの連携が有力
 - HoloVizなど、Pythonベースの可視化ツールは開発が活発
 - Keplerでの可視化もあり得る



日立での事例紹介

COVID-19をきっかけに、研究所内の"自由研究"としてスタート



YouTubeにて 「日立 フィジカルディスタンス」 でぜひ検索してください



サイトトップ > ニュース&イベント > 人の歩行に連動する空間演出の楽しさでフィジカル・ディスタンシングを誘発するシステムを開発

人の歩行に連動する空間演出の楽しさでフィジカル・ディスタンシングを誘発するシステムを開発

人の集まる公共空間における感染を抑止しつつ、自粛疲れの軽減を促す

2020年9月23日 株式会社日立製作所

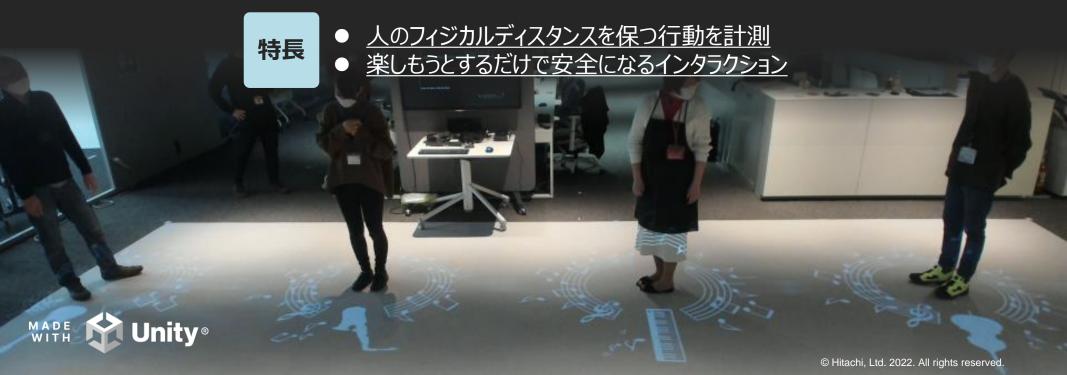
日立は、新型コロナウイルス感染症(COVID-19)の感染拡大防止ソリューションとして、施設利用者が自分の歩行に合わせて足元に投影される映像を楽しむことで、フィジカル・ディスタンシング*1 (身体的距離の確保) に繋がる行動を自然に誘発する新たな体感型映像システムを開発しました。本システムはリアルタイムに人の位置を計測できるLiDAR*2やTOFカメラ*3などの高精度なセンサーを用いて、個人を特定することなく人同士の距離を検知しており、適切な距離を保たないと映像が見えなくなるなど、フィジカル・ディスタンシングと連動するように動作します。これにより、駅や空港、大型商業施設、アミューズメントパークなどの人の集まる公共空間におけるウィルスの感染拡大を抑止することでサービス業の事業継続性を改善しつつ、足元に投影される映像から自ずと楽しさや気づきを得ることで施設に訪れる人々の自粛疲れを軽減する効果も期待できます。今後、日立はウィズコロナ、アフターコロナの時代を見据え、人流計測・解析結果の人へのフィードバック手法の最適化に取り組み、行動誘発ソリューション事業を拡大していきます。





フィジカルディスタンス誘発システムの プロトタイプ 開発と簡易検証

「楽しさ」と「フィジカルディスタンス」をつなぐ仕組み







感染拡大防止

COVID-19の感染拡大から自粛が求められている「フィジカルディスタンス」を保つ社会の維持は長期戦



個人のQoL維持

社会の経済活動維持

外出時は常に緊張を強いられ、"自粛疲れ"を起こしてしまう

自粛によって生活娯楽関連サービスは大きな打撃を受ける



従来の対策

レジや受付カウンタなどの列になる場所に 1~2m間隔の足元サインを張り付け 必要に応じ、係員が徹底を声がけ

現状の課題

施設側の人的コスト 行動規制による利用者の心的ストレス

- 適切に距離を保持し続ける緊張を強いられる
- 家族やグループが列に並んだ場合、その後ろに並ぶ人に とって適切な距離が保てなくなってしまう
- 適切な距離を保持しなくなる人がいることで、不要なトラブルが発生する





23

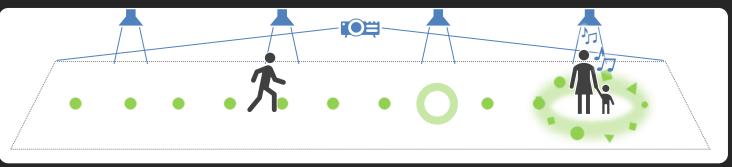
指向性スピーカ

TOFセンサー

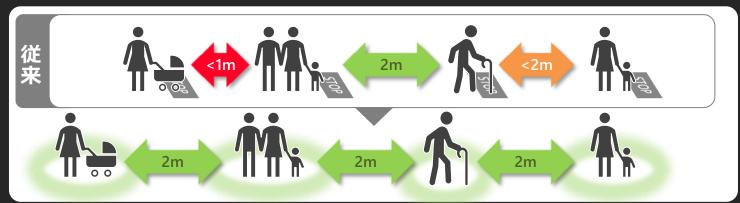
© Hitachi, Ltd. 2022. All rights reserved.

本提案

● 立ち止まるべき場所に保持し続けたくなる音と映像による演出



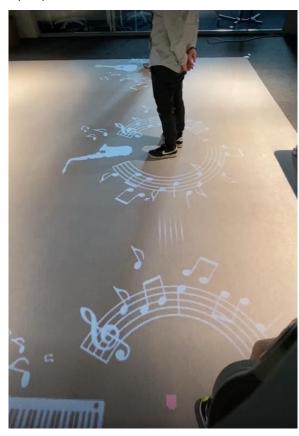
● グループで同じエリア内に立ち止まった場合、取るべき距離をダイナミックに変更する



デザイン仕様(例)



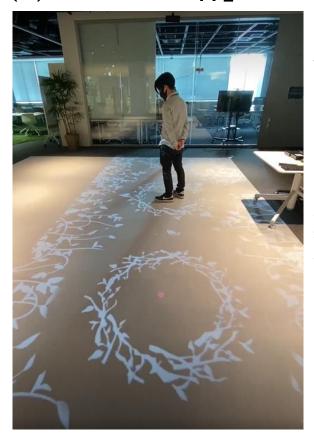
(A) コンセプト「ミュージック」



複合商業施設のカフェでの導入を想定したデザイン。

映像は五線譜の楽譜や楽器のイメージ、音響はジャズミュージックに合わせて、立ち止まった位置に表示された楽器イメージの演奏がセッションのように聴こえてくる。

(B) コンセプト「森」



自然の中のミュージアムでの導入を想定したデザイン。

映像のデザインイメージは森の茂道。音響は風のせせらぎの中、立ち止まった位置に表示された鳥や虫のイメージに合わせた生き物の音色が聴こえてくる。

動画コンテンツ①



動画コンテンツ②



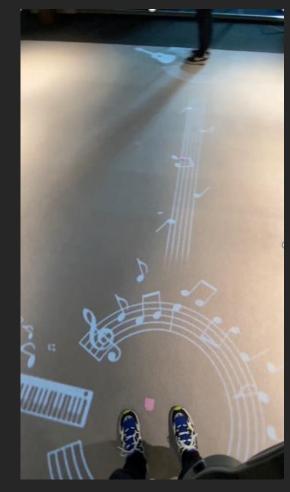
動画コンテンツ③



動画コンテンツ① ※公開資料用に代替のスクリーンショットを提示いたします。



五線譜に沿って入場します



適切な距離で立ち止まると…

動画コンテンツ② ※公開資料用に代替のスクリーンショットを提示いたします。

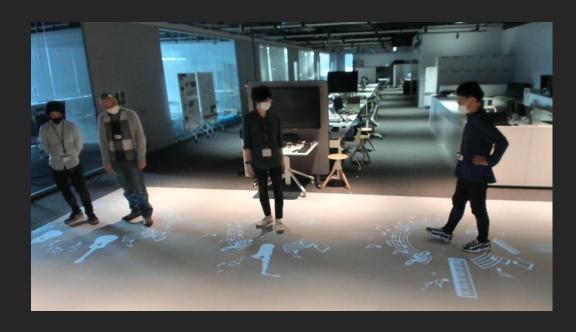


森の小道に沿って進む



適切な距離で立ち止まると…

動画コンテンツ③ ※公開資料用に代替のスクリーンショットを提示いたします。

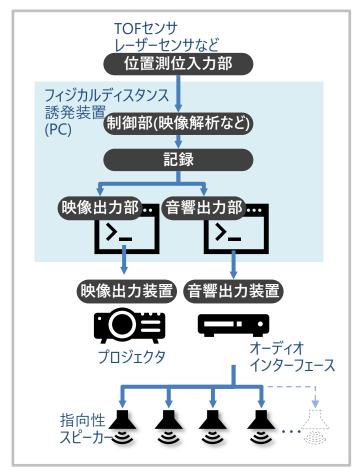




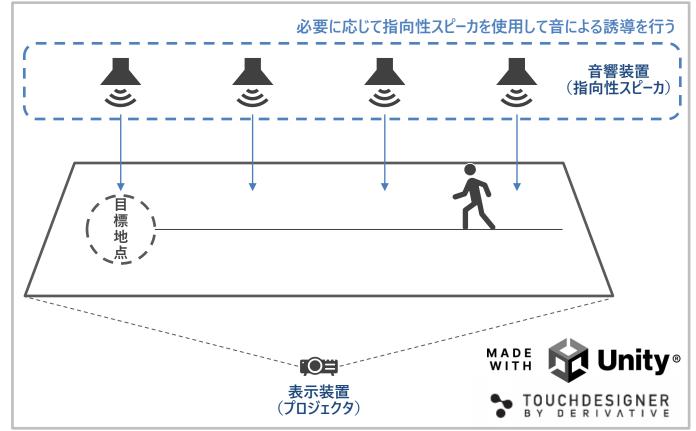
待機スポット内の人数が変わってもシームレスに追従して判別します

システム構成



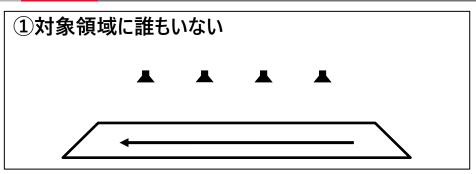


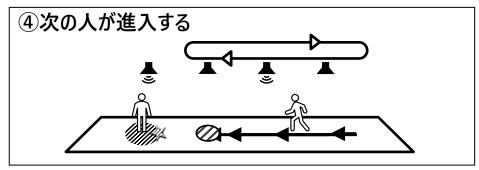
センサの出力結果からプロジェクタで情報を表示して、人の行動変容を促す

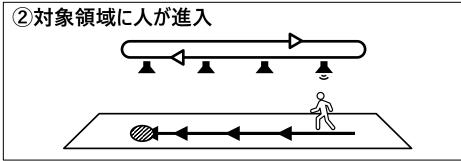


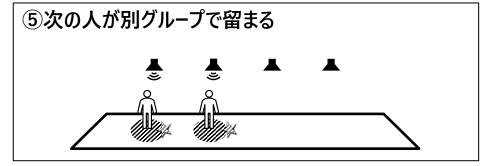
動作仕様

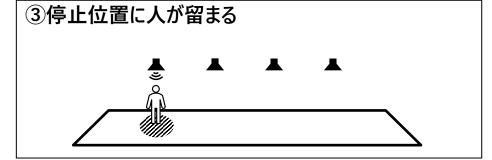












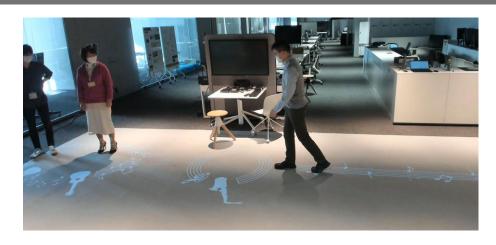


簡易実験



被験者 7 名にシステムの案内方法は事前説明せず、「案内に従い、カウンターに進んで下さい」という指示のみによって体験してもらい、下記 3 つの期待する行動を観察によって判定する簡易実験を実施。

- (1) 行列の配置, 進行方向
- (2) 立ち止まるべき位置
- (3) 進行すべきタイミング



被験者	(1)行列の配置,	進行方向	(2)立ち止まるべき位置	(3)進行すべきタイミング
Y.K.	0		0	0
M.I.	0		0	0
N.K.	0		0	0
T.H.	0		×	0
S.S.	0		×	0
M.K.	0		0	0
K.M.	0		0	0

□ (1) (3) は100%(7/7名)、(2) は70% □ 以上(5/7名)誘発が確認できた。体験 ② 後のインタビューから「(普段は)列で、 □ じっとしておくことはつらいけど、素敵な ■ 音が聴けるなら、ここにいようと思え □ る」など、心的ストレス軽減を示唆する □ コメントも得られた。

まとめ



- □ 待ち時間の演出とフィジカルディスタンスの行動誘発を促す手法を提案
 - ① 測距センサーによる人流計測・解析技術を用いて個人を特定することなく人同士の距離を検知
 - ② 行動と連動して投影映像と音響を変化させたマルチモーダルなインタラクション
- □ 2種類のシチュエーションを想定したプロトタイプを制作
 - ① 複合商業施設のカフェの注文カウンター前
 - ② ミュージアムの入場受付カウンター前
- □ 簡易実験より、フィジカルディスタンスを保つ行動誘発と心的ストレス軽減を両立できる効果が期待
 - ① 被験者7名中5名(70%以上)に対して、期待した行動を促すことができた
 - ② 被験者からは心的ストレス軽減を示唆するコメントも得られた
- □ 今回の簡易実験では、有意差を示す十分な評価はできていない。今後はサンプル数を増やした検証、また家族などの感染リスクの低いグループを配慮した行動誘発機能についての効果検証を実施する必要がある

Hitachi Social Innovation is

POWERING GOOD

HITACHI Inspire the Next