



第35回 全国高等専門学校

プログラミングコンテスト

まほろばの地で古都ロケ

花開

開催部門 ①課題部門「ICTを活用した環境問題の解決」をテーマにした作品
②自由部門 自由なテーマで独創的な作品
③競技部門 与えられたルールによる対抗戦「シン・よみがえれ世界遺産」

予選 書類による審査 令和6年6月22日(土)

本選 令和6年10月19日(土)～20日(日)
なら100年会館

高専プロコン公式サイト

<https://www.procon.gr.jp>

X (旧Twitter) : @KosenProcon(公式アカウント)
#procon35(ハッシュタグ)

第16回NAPROCK

国際プログラミングコンテスト

<https://naprock.jp/IntProcon>

主管校 独立行政法人国立高等専門学校機構 奈良工業高等専門学校

お問合せ 第35回全国高等専門学校プログラミングコンテスト事務局
奈良工業高等専門学校 学生課学生係
0743-55-6034 E-mail jimu35@procon.gr.jp

主催/一般社団法人 全国高等専門学校連合会 共催/特定非営利活動法人 高専プロコン交流育成協会(NAPROCK)

後援 / 文部科学省、総務省、経済産業省、警察庁、デジタル庁、奈良県、奈良県教育委員会、奈良市、奈良市教育委員会、大和郡山市、大和郡山市教育委員会、奈良県商工会連合会、一般社団法人ソフトウェア協会、一般社団法人情報処理学会、一般社団法人電子情報通信学会、一般社団法人教育システム情報学会、国立研究開発法人情報通信研究機構、一般社団法人デジタル地方創生推進機構、一般社団法人人工知能学会、NPO 法人 IT ジュニア育成交流協会、公益財団法人奈良県地域産業振興センター、日本弁理士会関西会、NHK、朝日新聞奈良総局、毎日新聞奈良支局、日刊工業新聞社、奈良テレビ放送、奈良新聞社、奈良高専後援会、奈良高専同窓会、奈良高専地域イノベーションコンソーシアム

第35回全国高等専門学校 プログラミングコンテスト

主催	一般社団法人 全国高等専門学校連合会
共催	特定非営利活動法人 高専プロコン交流育成協会(NAPROCK)
後援	文部科学省、総務省、経済産業省、警察庁、デジタル庁、奈良県、奈良県教育委員会、奈良市、奈良市教育委員会、大和郡山市、大和郡山市教育委員会、奈良県商工会連合会、一般社団法人ソフトウェア協会、一般社団法人情報処理学会、一般社団法人電子情報通信学会、一般社団法人教育システム情報学会、国立研究開発法人情報通信研究機構、一般社団法人デジタル地方創生推進機構、一般社団法人人工知能学会、NPO法人ITジュニア育成交流協会、公益財団法人奈良県地域産業振興センター、日本弁理士会関西会、NHK、朝日新聞奈良総局、毎日新聞奈良支局、日刊工業新聞社、奈良テレビ放送、奈良新聞社、奈良高専後援会、奈良高専同窓会、奈良高専地域イノベーションコンソーシアム
産学連携協賛	アバナード(株)、(株)NSD
特別協賛	(株)セゾンテクノロジー、(株)日立製作所、ピクシブ(株)、チームラボ(株)、(株)シーエーシー、(株)トヨタシステムズ、さくらインターネット(株)、ネクストウェア(株)、(株)FIXER、(株)ブロードリーフ、(株)jig.jp、(株)NTTデータ フィナンシャルテクノロジー、(株)アイ・エス・ビー、LINEヤフー (株)、アドバンスドプランニング(株)、(株)インフォコム西日本、(株)サイエンスアーツ、(合) DMM.com、(株)クレスコ、コスモリサーチ(株)、(株)デンソー、(株)インフォコム東日本、(株)サイバーエージェント、KDDIアジャイル開発センター(株)、(株)アクセスネット、(株)ビッツ、CompTIA 日本支局、NTTコム エンジニアリング(株)、国立研究開発法人理化学研究所 計算科学研究センター
一般協賛	(株)インテリジェント ウェイブ、(株)ワコム、アイフォーコム(株)、富士ソフト(株)、アイ・システム(株)、オープンテクノロジー(株)、TDCソフト(株)、(株)タブチ、(株)ワイズ、(株)クレスコ・ネクシオ、(有)ユニバーサル・シェル・プログラミング研究所、(株)CRI・ミドルウェア、(株)アイスタイル、(株)TOKIUM、(株)日本総合研究所、(株)朝日ネット、(株)オブティム、(株)RKKCS
募集部門	パソコン等で実行可能なソフトウェア環境のもとで以下の3部門で競う 1. 課題部門 「ICTを活用した環境問題の解決」をテーマにした作品 2. 自由部門 自由なテーマで独創的な作品 3. 競技部門 与えられたルールによる対抗戦「シン・よみがえれ世界遺産」
応募資格	国公立高専の学生(専攻科生を含む)
募集期間	令和6年5月20日(月)～27日(月)
審査	1. 予選(書類による審査) 日時 令和6年6月22日(土) 会場 奈良工業高等専門学校 2. 本選(プレゼン・デモ等による審査、競技は対抗戦) 日時 令和6年10月19日(土)～10月20日(日) 会場 なら100年会館(奈良市三条宮前町7番1号)
表彰	次の各賞を授与します。 課題・自由部門(各々) 最優秀賞 各1点(賞状および副賞) 優秀賞 各1点(賞状および副賞) 特別賞 各数点(賞状および副賞) ※ 最優秀賞受賞者には文部科学大臣賞、情報処理学会若手奨励賞、電子情報通信学会若手奨励賞が授与される。 競技部門 優勝 各1点(賞状および副賞) 準優勝 各1点(賞状および副賞) 第三位 各1点(賞状および副賞) 特別賞 各数点(賞状および副賞) ※ 最優秀賞受賞者には文部科学大臣賞、情報処理学会若手奨励賞、電子情報通信学会若手奨励賞が授与される。

The 16th NAPROCK International Programming Contest, Nara, Japan (NAPROCK PROCON 2024)

INTRODUCTION

NAPROCK has co-sponsored the College of Technology Programming Contest (Kosen Procon) since 2008, and started to sponsor the NAPROCK international programming contest since 2009. This year, the NAPROCK 16th international programming contest is held with the 35th Kosen Procon. This contest aims to promote flexibility of thinking through programming at a very high level. At the contest, students from kosen or universities compete with each other by utilizing knowledge and ideas in information processing technology they have learned everyday of their lives. It is required for them to make full use of the latest and evolving information processing technology.

CONTEST INFORMATION

DATE: Oct. 19 (Sat.) – 20 (Sun.) , 2024

VENUE: Nara Centennial Hall (7-1 Sanjo-Miyamaecho, Nara, Japan)

PARTICIPANTS: KOSEN students (who participant in KOSEN Programming Contest) ,
and students in foreign universities/KOSENs/institutes

EXAMINATION METHODS:

Themed Section and Original Section:

Both of the presentation and demonstration will be examined by the judges. The examination standard includes originality, technical capabilities of systems development, usefulness, ease of use, manual/ documentation preparation, speech and presentation skills, and so on. The operation manual and program source list will also be reviewed.

Competition Section:

Each team competes according to the answers to the designated problems.

AWARDS:

In the Themed Section and the Original Section, the following prizes will be awarded.

Grand Prize: 1 team, Second Prize: 1 team, Special Prize: several teams

In the Competition Section, the following prizes will be awarded.

Champion: 1 team, First Runner-up Prize: 1 team, Special Prize: several teams

PROCON WEB SITE:

NAPROCK Official site: <https://www.naprock.jp/>

NAPROCK Facebook page: <https://www.facebook.com/naprock2008>

KOSEN PROCON official site: <https://www.procon.gr.jp/>

SPONSORS and SECRETARIAT:

Main sponsor: Nourishment Association for Programming Contest

KOSEN Co-sponsor: Technical College Association

Supervising college: National Institute of Technology, Nara College

Judging Committee: KOSEN PROCON judging committee

Foreign Participants: VNU University of Engineering and Technology (Vietnam)

Hong Kong Vocational Training Council (Hong Kong)

Singapore Polytechnic (Singapore)

King Mongkut's Institute of Technology, Ladkrabang (Thailand)

KOSEN-KMITL (Thailand)

Mongol Koosen College of Technology (Mongolia)

New Mongol College of Technology (Mongolia)

Secretariat: NAPROCK

ご挨拶

大会会長挨拶

全国高等専門学校連合会会長
釧路工業高等専門学校長

大塚 友彦



第35回全国高等専門学校プログラミングコンテストが、全国高等専門学校連合会主催、NPO 法人高専プロコン交流育成協会（NAPROCK）の共催のもと、盛大に開催できますことを大変喜ばしく思います。

高等専門学校（高専）は15歳から5年一貫の早期エンジニア養成を行っている高等教育機関で、現在、全国で国公立合わせて58高専あり、高専連合会はこれらの連合組織です。体育大会や各種コンテストを主催して高専生に活躍の場を提供し、コミュニケーション力等の人間力の醸成や技術者教育の補完を担っております。

デジタル化の進展により、異なる専門分野の技術を融合させて付加価値の高い「ものづくり」や「ことづくり」が要請される時代において、イノベティブなエンジニアにとって情報通信技術の理解と修得は今や必須です。本コンテストは、高専生が日頃の学修成果を生かし、情報通信技術によってアイデアを実現する力を競うことで、創造性・独創性を育てるプロジェクトの一つです。本コンテストを通じ、高専生の若く力強いエネルギーや柔軟な発想力を広く社会に発信します。本コンテストは、1990年度（平成2年度）より開催され、今年で35回目となります。年々技術的レベルも向上しており、産業界や学会等から高い評価を受けています。

本選大会では、作品の独創性、有用性・実現可能性を審査し、予選を勝ち抜いたチームや個人が会場に集まり、パフォーマンスを競い合います。本年度は、「まほろばの地で古都ロジー開花」と題し、例年どおり課題部門、自由部門および競技部門の3部門を開催します。

コロナ禍の経験を経て、さらなる発展を遂げてきた情報通信技術を、高専生たちは課題解決の「知恵」として応用し、見事に生み出された作品を最高のプレゼンテーションとデモンストレーションで日本中を魅了してくれることでしょう。皆様にも高専生の若さ溢れる感性・創造性・技術力、引いては、今や世界の「KOSEN」と言われる日本の高専教育の神髄を感じていただけるものと思います。どうか応援よろしくお願いいたします。

最後に、ご後援いただきました文部科学省、総務省、経済産業省、警察庁、デジタル庁、奈良県、奈良県教育委員会、奈良市、奈良市教育委員会、大和郡山市、大和郡山市教育委員会、奈良県商工会連合会、奈良県地域産業振興センター、関連の学協会・団体、報道機関等、協賛いただきました企業等の方々に深く感謝申し上げますとともに、ご多忙にもかかわらず審査委員をお引き受けいただいた先生方、企画・運営された実行委員会の皆様、主管校である奈良高専の近藤校長先生はじめ教職員の皆様、そして今回の高専プロコンにご支援ご協力いただきました全ての皆様に心より御礼申し上げます。

NAPROCK 理事長挨拶

桑原 裕史



奈良高専を開催校として古都奈良で、文部科学省をはじめ多くの皆様から後援・協賛をいただき、第35回全国高等専門学校プログラミングコンテストが開催できますことを誠に喜ばしく存じます。今年、日本列島は数度も大きな地震に襲われました。この災害でお亡くなりになられた方々に謹んでお悔やみを申し上げますとともに、被災されました皆様に心からお見舞いと一日も早い復興を心よりお祈り申し上げます。このような状況にありながら、残念な事に近い将来大規模地震発生の恐れが指摘されており、自然災害への備えを怠ってはなりません。さらに、環境破壊や地球温暖化等の環境問題も我々の生活にとって早急な解決が求められる大きな課題です。今回、課題部門のテーマは新しく「ICTを活用した環境問題の解決」と設定されました。この問題解決には様々な切り口が考えられると思いますが、高専学生が得意とする独創性と実現力が発揮できる課題であり、自由部門、競技部門とともに素晴らしい作品の競演となることを期待し学生諸君の奮闘を拝見したいと存じます。

同時開催される NAPROCK 主催の国際大会につきましても16回を数えることになり、海外より多数のチームを迎えて実施できることを誠に嬉しく思います。この大会がIoT技術の習得を目指す世界の学生諸君の切磋琢磨の場となることを期待する次第です。

さて、これらの大会の開催は、全国高等専門学校連合会、プロコン実行委員会の皆様方のご尽力と今回主管校を引き受けていただきました近藤科江校長をはじめとする奈良高専の教職員のお力・ご負担によるものと存じ、改めまして心より感謝申し上げます。また、審査委員の皆様方には予選時から厳しいスケジュールでお取り組みいただき、そのご苦勞に対し深く感謝申し上げます。

最後になりましたが、本大会が長きに渡り継続的に開催できますのは、ご協賛いただく企業の方々およびご後援下さる皆様のご理解・ご尽力の賜物と存じます。多大なるご援助に心より感謝申し上げますとともに、高専生および高専の、ひいては高度化する情報社会の進展のために一層のご支援をお願い申し上げ、挨拶とさせていただきます。

第35回全国高等専門学校 プログラミングコンテスト 実行委員長挨拶

奈良工業高等専門学校長
近藤 科江



第35回全国高等専門学校プログラミングコンテストの開催にあたり、ご挨拶申し上げます。

本大会は、JR奈良駅直結のなら100年会館での開催です。大会テーマは「まほろばの地で古都ロジー開花」とし、日本最古の文化が花開いた奈良の地で未来を拓くプログラミングの祭典を催して、「温故知新」の気持ちで高専生に日頃の学習成果を大いに発揮して、新たなテクノロジーに繋がるパフォーマンスをして頂きたいとの期待を込めております。課題部門では、「ICTを活用した環境問題の解決」をテーマにして、世界的な異常気象の原因として対策が急がれている環境問題に挑んでもらいます。独自に製作した作品を競い合う自由部門では、高専生の斬新なアイデアが光ります。競技部門では、舞鶴大会のテーマ「よみがえれ世界遺産」を、世界遺産の宝庫である奈良の地で再び、いにしへの偉業に思いを馳せて「シン・よみがえれ世界遺産」をテーマに掲げ、少ない手数で如何に早く文化財修復を成し遂げられるかを競います。いずれもプロコン創設以来めざしてきた高専生の「独創性や創造性の育成」を重視し、SDGsにも繋がる素晴らしい内容になっており、高専生がどのように取り組んでくれるのか楽しみです。

前回の福井大会に引き続き、第16回NAPROCK国際プログラミングコンテストの同時開催も行われます。タイ、シンガポール、モンゴル、ベトナム、香港から12チームが参加します。本大会を介してプロコンの意義が海外でも展開され、会場での交流が高専生の視野を世界に広げてくれる機会になることを期待いたします。

最後に、本大会の開催にあたりまして、協賛、後援を頂きました企業・団体の皆様、大会の企画・運営にご支援頂きました高専プロコン交流育成協会(NAPROCK)の皆様、審査委員の皆様、当地における大会開催にあたりまして、多大なるご理解とご協力を頂きました文部科学省、高専連合会、奈良県、奈良市、大和郡山市をはじめ関係者の皆様に心よりお礼申し上げます。

大会日程

10月19日(土)

- 8:30～8:50 参加者連絡会議
- 9:30～10:00 開会式
- 10:10～17:14 プレゼンテーション審査(課題部門・自由部門)
- 10:30～17:00 デモンストレーション公開(課題部門・自由部門)
- 12:00～17:00 競技部門(ファーストステージ・セカンドステージ)
- 17:30～19:00 学生交流会

10月20日(日)

- 8:15～8:30 参加者連絡会議
- 9:00～12:15 デモンストレーション審査・マニュアル審査(課題部門・自由部門)
- 8:30～14:00 競技部門(敗者復活戦・ファイナルステージ)
- 14:15～14:45 特別講演
- 15:00～16:30 閉会式

名刺クエストと学生交流会

第35回奈良大会においても協賛企業と参加学生・教員の交流促進のため名刺クエストを実施いたします。本年度も学生交流会の企画として学生同士の名刺交換も実施します。

従来の企業と学生・教員間の名刺交換を「企業部門」と位置づけ、昨年に引き続き学生同士の名刺交換を行う「学生部門」を設置し、交換した名刺の数を競います。企業部門と学生部門のそれぞれ上位10チームに賞品を進呈します。

また、参加者同士の交流を促進するため、10月19日(土)17:30～19:00に大ホールにおいて学生交流会を実施いたします。交流としてクイズ大会などを行い、豪華賞品も用意しておりますので、是非参加ください。

特別講演

日 時: 10月20日(日) 14:15～14:45

講演題目: 「奈良高専プロコン OB プロコンから得た学びと卒業後の10年を振り返る」

講師: ・合同会社 DMM.com デジタルコンテンツ開発本部 矢野 完人 氏
・TerraDrone 株式会社 開発本部 小川 夏輝 氏
・サイボウズ株式会社 開発本部 上岡 真也 氏

奈良高専に在籍中は情報処理研究会(通称 Joken)に所属し、高専プログラミングコンテストに幾度か出場していた3名。専攻科、奈良先端科学技術大学院大学と同じ進路を進み、学生生活を共に過ごす。大学院修了後はそれぞれ別の道を歩む。

東証プライム上場の B2B 向け SaaS 企業、「なんでもやっている」を謳う非上場メガベンチャー、ドローン技術で世界を変えようとするスタートアップ企業と、三者三様な IT 企業に所属する3名が、高専プロコンから得た学びを振り返りつつ、未来のエンジニアに向けてこれまでの10年と IT 業界の今を語る。

審査委員

審査委員長	大場みち子	京都橘大学 工学部 情報工学科 教授
審査副委員長	杉田 泰則	長岡技術科学大学 技学研究院 電気電子情報系 准教授
審査委員	綾塚 祐二	株式会社クレスコ 技術研究所 所長
	荒川 弘毅	株式会社サイエンスアーツ 開発部
	有馬 三郎	株式会社セゾンテクノロジー 執行役員 CTO
	飯岡あゆみ	株式会社インテリジェント ウェイブ
	石川 順平	株式会社FIXER Microsoft Top Partner Engineer
	稲見 昌彦	東京大学 総長特任補佐・先端科学技術研究センター 副所長・教授
	犬塚 正	株式会社トヨタシステムズ 事業企画推進部 部長
	井上美智子	奈良先端科学技術大学院大学 先端科学技術研究科 教授
	今井 一範	アドバンスドプランニング株式会社 代表取締役
	上原 隆平	北陸先端科学技術大学院大学 先端科学技術研究科 副学長・研究科長・教授
	内原 浩介	アパナード株式会社 グループマネージャ
	奥村 充男	株式会社デンソー 先端技能開発部 モビリティ実験室 室長
	加邊 徹	株式会社アイ・エス・ビー 管理本部 人事部シニアマネージャー
	香山 瑞恵	信州大学工学部 電子情報システム工学科 教授
	北岡 教英	豊橋技術科学大学 情報・知能工学系 教授
	木村 倫也	NTTコムエンジニアリング株式会社 サービスネットワーク部 サービスネットワーク部門 サービスフロントユニット 主査
	久保 慎一	ネクストウェア株式会社
	熊野 弘幸	株式会社シーエーシー R&D本部 サービスプロデューサー
	木暮 圭一	KDDIアジャイル開発センター株式会社 代表取締役社長 / CEO
	笹岡賢二郎	一般社団法人 ソフトウェア協会 専務理事
	佐藤 崇	株式会社ビッツ 取締役
	舘山 北斗	株式会社サイバーエージェント 機械学習エンジニア
	田中 務	株式会社インフォコム西日本 代表取締役社長
	田村 哲也	チームラボ株式会社 取締役
	鳥越 直寿	株式会社インフォコム東日本 代表取締役社長
	中西 弘明	株式会社ブロードリーフ 特販・契約基盤開発課 エンジニアリーダー
	長谷川 泰	NHK メディア技術局 コンテンツテクノロジーセンター デジタルソリューション 専任部長
	福原 一仁	株式会社NTTデータ フィナンシャルテクノロジー 日銀システム事業部 副事業部長
	三池真優子	株式会社NSD 執行役員
	道井 俊介	ピクシブ株式会社 執行役員 CTO
	宮地 力	NPO法人 音楽の未来を拓くスマイル協会 理事
	矢野 完人	合同会社DMM.com 動画配信事業部 配信基盤グループ マネージャー
	山地 駿徹	株式会社jig.jp エンジニア
	吉田 育代	フリーランスライター
	鷺北 賢	さくらインターネット株式会社 さくらインターネット研究所 所長

(五十音順 敬称略)

高専プログラミングコンテスト発展の経緯

第35回全国高専プロコンは今年度、奈良高専が主管となり奈良県奈良市において本選を開催致します。今大会は、課題部門40テーマ、自由部門57テーマ、競技部門58チームの応募があり、予選を通過した課題・自由部門各20テーマ、競技部門58チームの参加により本選が実施されます。これも、様々な制約の中でシステムの構想から開発に至るまで努力された関係各位のご尽力の賜物であり、深く感謝申し上げます。

さて、これまでの高専プロコン発展の経緯について説明致します。本コンテストの主催団体は一般社団法人全国高等専門学校連合会（以下、「高専連合会」）です。旧連合会の組織に高等専門学校情報処理教育研究委員会があり、情報処理教育に関わる全国高専の教員の代表が、種々の調査研究や催し物の立案を行っていました。平成元年8月、この委員会の常任委員会で全国の高専生を対象としたプログラミングコンテストの開催が採択され、この会を母体として本コンテストの実行委員会が編成されました。本コンテストは、情報処理技術の発展に対する期待の高揚や教員・学生の交流の機会拡大などの狙いもありましたが、高専が持つ若くて力強いエネルギーや発想の柔軟性を世の中に紹介したいという強い願いもありました。

第1回大会は、1年間の準備期間を経て京都国際会館で開催されました。全国41高専の応募から、予選審査を経て、自由部門10テーマ、課題部門6テーマが本選に臨みました。初回の本選は、盛大な中にもアカデミックな香りが満ちあふれ、反響も極めて良いものでした。応募作品の一部はソフトウェアハウスからアプローチを受けるなどの実績も得られました。当初、課題・自由の2部門でスタートした本コンテストですが、第5回大会から競技部門を設け、3部門体制で実施しています。高専全体のイベントとして定着するとともに、運営面でも変化が現れてきました。第4回大会から開催校（主管校）が本選の運営担当として設けられ、最近では、募集から本選開催にわたりコンテスト運営の中心となっています。

技術教育に主眼をおく高専においては、創造性・独創性を涵養する教育への取り組みが強く求められています。創造性教育のプロジェクトの一つとして、高専プロコンにも大きな期待が寄せられています。その目的を果たすため、作品の独創性を審査で重視するとともに、初回からプレゼンテーション審査とデモンストレーション審査の両方を課し、学生に対し表現力の涵養を図って参りました。主催団体である高専連合会も、教育プロジェクトとしての高専プロコンの役割を重視し、下部機関として全国高専プログラミングコンテスト実行委員会を独立して発足させ、高専プロコンのさらなる充実を図っています。

第2回大会からは文部省からもご後援を賜り、第4回大会からは念願の文部大臣賞（現 文部科学大臣賞）を、第6回大会からは競技部門を含む全部門で文部大臣賞をいただけるようになりました。第25回大会からは総務省、経済産業省、第27回大会からは警察庁、第33回大会からはデジタル庁の後援を賜っております。また、18回課題・21回自由・22回課題・25回課題部門の最優秀作品が第3～7回ものづくり日本大賞（内閣総理大臣賞）を連続受賞しました。その他にもプロコンの教育効果に対する高い評価を各界からいただいています。さらには、大会運営に関わるプロコン委員が、（公）日本工学教育協会から工学教育賞を、（一社）情報処理学会からは教育賞を頂戴しております。

一方、高専が社会に対して貢献していくためには、産業界との連携も重要な課題の一つです。高専プロコンは第1回大会より（社）日本パーソナルコンピュータソフトウェア協会（現（一社）ソフトウェア協会）から後援をさせていただき、絶大なご援助をいただいております。第1回大会は6社からスタートした大会協賛ですが、最近では50社を超える多くのご支援をいただいております。また、マスコミ各社からもご後援を頂戴しております。さらに高専プロコンを支援する特定非営利活動法人（NPO法人）高専プロコン交流育成協会が平成20年7月に東京都の認可を受け、第19回大会から共催団体として加わりました。これも、後援団体ならびに協賛企業はじめ各方面からのご支援があって実現したものと深く感謝しております。

プロコンの国際化も進んでいます。第8回大会ではオーストリアへ、第10回大会では韓国へ課題部門の最優秀賞受賞チームの派遣が行われました。そして、第15回大会ではベトナムのハノイ工科大学をはじめオープン参加で受け入れ、これまで、ベトナム、モンゴル、中国、台湾、タイ、マレーシア、シンガポール、香港の7つの国と地域から延べ100チーム以上を本選に迎えています。また、第20回大会より、NAPROCK国際プログラミングコンテストを同時開催しています。さらに、第30回大会では、同時開催ではなく海外（ハノイ：ベトナム）での国際大会の実施が計画され、課題部門と競技部門の上位入賞チームが参加予定でしたが、新型コロナウイルス感染症感染拡大の影響により現地開催は中止となりました。なお、第35回大会は国際大会と同時開催となり、海外より12チームの参加が予定されております。

最後に、次世代の日本を支える高専生のエネルギーと皆様のご支援を糧として、高専プロコンを核とした益々の発展的な展開を目指して引き続き努力したいと考えております。

高専プログラミングコンテスト開催実績

回数	開催年	開催地	主管校	予選会場
第1回	平成2年	京都市		フォーラム8
第2回	平成3年	大分市		サンプラザ
第3回	平成4年	仙台市		東京文化会館
第4回	平成5年	名古屋市	豊田高専	都立高専
第5回	平成6年	富山市	富山商船高専	東京高専
第6回	平成7年	函館市	函館高専	東京高専
第7回	平成8年	北九州市	北九州高専	東京高専
第8回	平成9年	長岡市	長岡高専	東京高専
第9回	平成10年	明石市	明石高専	東京高専
第10回	平成11年	呉市	呉高専	都立高専
第11回	平成12年	津市	鈴鹿高専	都立高専
第12回	平成13年	鶴岡市	鶴岡高専	都立航空高専
第13回	平成14年	金沢市	石川高専	都立航空高専
第14回	平成15年	八王子市	東京高専	育英高専
第15回	平成16年	新居浜市	新居浜高専	都立高専
第16回	平成17年	米子市	米子高専	都立高専
第17回	平成18年	ひたちなか市	茨城高専	都立高専(品川)
第18回	平成19年	津山市	津山高専	都立高専(品川)
第19回	平成20年	いわき市	福島高専	サレジオ高専
第20回	平成21年	木更津市	木更津高専	田町CIC
第21回	平成22年	高知市	高知高専	サレジオ高専
第22回	平成23年	舞鶴市	一関・舞鶴高専	舞鶴市総合文化会館
第23回	平成24年	大牟田市	有明高専	都立産技高専（品川）
第24回	平成25年	旭川市	旭川高専	都立産技高専（品川）
第25回	平成26年	一関市	一関高専	関東ITソフトウェア健保会館（市ヶ谷）
第26回	平成27年	長野市	長野高専	都立産技高専（品川）
第27回	平成28年	伊勢市	鳥羽商船高専	都立産技高専（品川）
第28回	平成29年	周南市	大島商船高専	都立産技高専（品川）
第29回	平成30年	徳島市	阿南高専	都立産技高専（品川）
第30回	令和元年	都城市	都城高専	都立産技高専（品川）
第31回	令和2年	オンライン	苫小牧高専	オンライン
第32回	令和3年	オンライン	秋田高専	オンライン
第33回	令和4年	高崎市	群馬高専	群馬高専
第34回	令和5年	越前市	福井高専	都立産技高専（品川）
第35回	令和6年	奈良市	奈良高専	奈良高専

全国高専プログラミングコンテスト応募状況一覧 第31回～第35回

凡例：◎競技部門優勝、○競技部門準優勝 ※競技部門の応募作品は、審査の結果、予選通過。第31回は競技部門中止。

学 校 名	第31回			第32回			第33回			第34回			第35回		
	課題	自由	競技	課題	自由	競技	課題	自由	競技	課題	自由	競技	課題	自由	競技
函 館	2	1		1			2	1		1	1		1		1
苫 小 牧						1			1			1			1
釧 路					2							1			1
旭 川		1		1	1	1		2	1		1	1		1	1
八 戸	1	1		1	1	1	1		1	1	1	1	1	1	1
一 関	1	1		1	1	1	1	1		1					1
仙 台（名 取）						1			1			1			1
仙 台（広 瀬）	1	1		1	1	1		1	1	1	1	1	1	2	1
秋 田		1			1	1		1				1			1
鶴 岡						1			1			1			
福 島	1	1		1	1	1	2			1	2	1	1		1
茨 城				1	1	1	1	1			1			1	
小 山	1	2		2	2	1	1	2	1	2	2	1	2	2	1
群 馬	1			2		1	1	1	1	1	2		1	2	1
木 更 津		1		2	1				1			1		1	1
東 京	1	1		1	1	1	1	2	1	1	1	1		2	1
長 岡					1	1	1	1	1		1			1	1
長 野				1		1			1	1		1	1	1	1
富 山（本 郷）	1	1		1	1	1	1		1	1	1	1	1	1	1
富 山（射 水）	2	1		1	2	1	1	1	1		1	1		1	1
石 川		1		1	1				1	1	1	1		2	1
福 井	2	2		2	2	1	2	2	1	2	2	◎1	2	2	1
岐 阜	1	1		1	1					1	1				1
沼 津	1	1			2	1		1	1			1		1	1
豊 田	1	1			2	1	1	2	1	2	2	1		1	1
鳥 羽 商 船	2	2		2	2	1	2	2	1	2	2	1	2	2	1
鈴 鹿					2				1	1		1			1
舞 鶴	1	2		2	2	1	2	2	1	2	2	1	2	2	1
明 石									1			1		1	
奈 良	2					1			1	1	1	1	1		1
和 歌 山						1		1	1			1			1
米 子	2	1		2	2	1	1	1	1	1	1	1	1	1	1
松 江	2	2		1	2	1	1	2	1	1	2	1	2	1	1
津 山	1	1		2	2	1	2	1	1	1	1	1	2		1
広 島 商 船	1	1		1		1			1	1	1	1		1	1
呉	1	1		2	1	1	1	1	1	1	1	1	1	1	1
徳 山	1				1	1	1			1		1		1	1
宇 部	1	1		1	1	1	1		1	1		1	1		1
大 島 商 船				1	1	1	1	1	1	1	1	1	2	1	1
阿 南	1	2		1	2	1	2	1	1	2	2	1	2	1	1
香 川（高 松）	1	1		1	1	1	1		1	1	1	1	1	2	1
香 川（詫 間）	1				1	1	1	1	1	1	1	1	2	2	1
新 居 浜	1					1			1		1	1		1	1
弓 削 商 船	1				1	◎1	1	1	○1	1	1	1	1	1	1
高 知											1	1	1	2	1
久 留 米	1	1		1	1	1			1		1	1		2	1
有 明	2	2		1	1	1	2		1	1	2	1		2	1
北 九 州				1											
佐 世 保						1			1			1		2	1
熊 本（熊 本）	2	1			2	1	2	2	1			○1		1	1
熊 本（八 代）	1	2			2	1		1	1	1	2	1	2	1	1
大 分						1			1			1			1
都 城								1			1	1		1	1
鹿 児 島	1					1	1		1	1		1			1
沖 縄	1			1	1		1	2		1	2		2	2	1
都 立（荒 川）						1						1			1
都 立（品 川）				1		1	1		1	1		1			1
大 阪 公 大						○1			◎1			1			1
神 戸 市 立	1			1	1	1	1	1	1	2	2	1	1	1	1
サ レ ジ オ						1			1		1	1			1
国 際						1						1	1	1	
近 畿 大 学	1					1				1		1			1
札 幌 市 立															
神 山										1	2		2	2	1
	46	39	0	43	54	51	41	41	46	45	52	55	40	57	58
	85			148			128			152			155		

凡例：◎文部科学大臣賞・最優秀賞、○優秀賞

11

第 34 回 全国高専 プログラミングコンテスト 本選結果

■ 課題部門

賞の名称	タイトル	学校名	指導教員	作成学生
文部科学大臣賞 最優秀賞 情報処理学会 若手奨励賞 電子情報通信学会 若手奨励賞	転生将棋 －新感覚中盤トレーニング－	熊本(八代)	小島 俊輔	宮本 健太郎、稲富 将大、辻中 煌希、 岩谷 奏慈郎、栗林 雄大
優 秀 賞	FishCam －遊漁船業のオンライン安全確認・釣果共有 システム－	鳥羽商船	江崎 修央	永井 玖愛、椿 理紗子、奥村 茉奈、 國米 一心、北仲 悠人
特 別 賞	BLAST OF BRAIN	大島商船	重本 昌也	西川 陸、神崎 友希、中野 美乃夏、 平松 莉奈、清水 蒼馬
特 別 賞	UPUPUP	香川(詫間)	宮武 明義	平尾 百智、大野 暉人、尾崎 真心、 中山 広夢、杉村 晃紳
特 別 賞	CYBER WARS －セキュリティ初学者の夜明け－	阿南	太田 健吾	折上 泰生、中田 東吾、木下 聡大、 日下 遥斗、久保田 隼輔
特 別 賞	ANNECT	函館	小山 慎哉	今野 佑星、吉田 海翔、川尻 千遥、 片野 遥恭、大田 陽

■ 自由部門

賞の名称	タイトル	学校名	指導教員	作成学生
文部科学大臣賞 最優秀賞 情報処理学会 若手奨励賞 電子情報通信学会 若手奨励賞	わんもあ －砂と鏡で創るもう一つの世界－	香川(詫間)	金澤 啓三	山田 美羽、天竺 寛貴、横井 優樹、 小原 崇靖、大西 俊輔
優 秀 賞	ばどろーる －安心・安全なカヤック支援システム－	鳥羽商船	江崎 修央	多米 希花、白川 琥大、松葉 勇希、 中森 立樹、塚本 真己也
特 別 賞	WashBoard －学生寮の洗濯管理をスマートに－	松江	渡部 徹	木島 陽斗、宮脇 楽久、大崎 翔太、 長谷川 友音、松谷 颯太
特 別 賞	μ sight －ひとりでも合奏がしたい! AR 合奏練習アプリ－	神山	正木 忠勝	金谷 尚興、鈴木 結衣、宮野 柊太、 山口 空、メリット キア
特 別 賞	SeQuick －気軽な学習ゲームで、セキュリティ人材を増 やそう－	久留米	原田裕二郎	山福 桜綺、田中 伶佳、梅林 直生、 近藤 碧
特 別 賞	継承の玉華 －VR で拓く花火職人道－	大島商船	北風 裕教	嶋 諒大、初崎 雛希、成田 健志郎、 神所 恭佑、河本 竜慎

■ 競技部門

賞の名称	タイトル	学校名	指導教員	作成学生
文部科学大臣賞 優勝 情報処理学会 若手奨励賞 電子情報通信学会 若手奨励賞	蟹高専	福井	西 仁司	前田 学、高橋 知也、辻 大知
準 優 勝	ゴリゴリズム	熊本(熊本)	藤井 慶	西村 淳志、中村 春喜、永松 日月
第 三 位	ボゴソート	豊田	三浦 哲平	宇井 颯汰、鈴木 健太、片野 蓮香
特 別 賞	そこにAIはあるのか?	徳山	力 規晃	児玉 隆之介、棟近 祐希、 山崎 秀太
特 別 賞	ゼスミナ	近畿大学	坂東 将光	滝野 慎之助、西半 涼賀、楓 悠斗
特 別 賞	武田進言	都立(荒川)	鈴木 弘	高野 陽大、豊田 アディール、 三澤 進生

NAPROCK 15th INTERNATIONAL PROGRAMMING CONTEST

■ Themed section

賞	作品タイトル	学校名	指導教員	作成学生
Grand Prize	転生将棋 －新感覚中盤トレーニング－	熊本（八代）	小島 俊輔	宮本 健太郎、稲富 将大、辻中 煌希、 岩谷 奏慈郎、栗林 雄大
Second Prize	FishCam －遊漁船業のオンライン安全確認・釣果共有 システム－	鳥羽商船	江崎 修央	永井 玖愛、椿 理紗子、奥村 茉奈、國米 一心、 北仲 悠人
Special Prize	Z-crafto	モンゴルコーセン 技術カレッジ	Jambaldorj Oyundari	Tserenlkham Batsuren、 Khosbileg Bilegsaikhan
Special Prize	Virtual band	タイ高専	Thanyawarat Pawasopon	Napatsakorn Kamutchat、 Pattanun Maneekeolphan、 Thanakrit Nilsuwanwong

■ Original section

賞	作品タイトル	学校名	指導教員	作成学生
Grand Prize	わんもあ －砂と鏡で創るもう一つの世界－	香川（詫間）	金澤 啓三	山田 美羽、天竺 寛貴、横井 優樹、小原 崇靖、 大西 俊輔
Second Prize	ばどろーる －安心・安全なカヤック支援システム－	鳥羽商船	江崎 修央	多米 希花、白川 琥大、松葉 勇希、中森 立樹、 塚本 真己也
Special Prize	家畜数え	モンゴルコーセン 技術カレッジ	Jambaldorj Oyundari	Odbayar Dandar、Amarsaikhan Galbadrakh
Special Prize	Artlet	新モンゴル高専	Shur-Erdene Buy- annemekh	Enkhsaikhan Batsaikhan、 Myagmarsuren Nyamkhuu
Special Prize	Gift-O	シンガポール ポリテクニク	Kwee Yin Wong	Lucius Chee Zihan、Lim Jia Xuan、 Celine Chan
Special Prize	Virtual Reality Therapeutic Application for Stress Reduction	キングモンクット 工科大学 ラカバン校	Samart Moodleah	Panuwat Kongchansawang、 Santakorn Wongsiripa
Special Prize	Authentication Via Discord Bot	タイ高専	Thanyawarat Pawasopon	Pattarachanon Harnkumnedpong、 Jaruphich Buakhunnen、 Kamathorn Mueangmun

■ Competition section

賞	作品タイトル	学校名	指導教員	作成学生
Champion	蟹高専	福井	西 仁司	前田 学、高橋 知也、辻 大知
First Runner-up Prize	ゴリゴリズム	熊本（熊本）	藤井 慶	西村 淳志、中村 春喜、永松 日月
Special Prize	Silver	新モンゴル高専	Shur-Erdene Buy- annemekh	Bilguuntushig Amarsaikhan、 Dulguun Zolzaya
Special Prize	VTC	香港 VTC	Choy Shu Sang	Lau Wing Yin、Chau Ka Tai、 Lam Long Hei Jason
Special Prize	CoreMind	モンゴル科学技術 大学	Khuder Altanger- el	Gal-Erdene Chinbat、 Ulbayar Luvsanjargal
Special Prize	HaUI_Miuxinhhh	ハノイ工業大学	Ma Thi Chau	Nguyen Tien Duy、Nguyen Duy Minh Quan

■ 企業賞

賞	作品タイトル	学校名	指導教員	作成学生
ピクシブ企業賞	Joint Draw －気軽な合作－	松江	杉山耕一郎	近藤 菜々美、上湊 翔太
jigjp 企業賞	Learn Mate －学生の学生による学生のための連絡アプリ－	有明	松野 良信	猿渡 優衣、古川 蒼太郎、坂口 凜華、 緒方 太一、中川 ひかる
セゾン情報システムズ企業賞	I-POST －世界一シンプルなコミュニケーションを－	舞鶴	森 健太郎	西村 陸杜、加川 直澄、伊勢 巧、 池田 翔太郎、Ganbaatar Bayartulga
さくらインターネット企業賞	NABASE	津山	宮下 卓也	森中 智己、瀬良 瑞葉、鈴木 冬馬、満 瑛洋、 石原 希光
シーエーシー企業賞	biblio connect	神山	正木 忠勝	丹那 伊織、伊藤 楽大
Blueship 企業賞	SeQuick －気軽な学習ゲームで、セキュリティ人材を増 やそう－	久留米	原田裕二郎	山福 桜綺、田中 伶佳、梅林 直生、近藤 碧
NSD 企業賞	VibraSymphony －全ての人にリアルな VR 体験を－	石川	越野 亮	荒木 建貴、箕崎 仁、鍛冶 光、 橋本 月冨、野崎 大翔
AGEST 企業賞	SeQuick －気軽な学習ゲームで、セキュリティ人材を増 やそう－	久留米	原田裕二郎	山福 桜綺、田中 伶佳、梅林 直生、近藤 碧
チームラボ企業賞	服薬守くん－メディカルサポーター－	小山	小林 康浩	藤澤 颯介、青木 優哉、田代 幸助、 癸生川 大斗、長澤 陽生
トヨタシステムズ企業賞	WashBoard －学生寮の洗濯管理をスマートに－	松江	渡部 徹	木島 陽斗、宮脇 楽久、大崎 翔太、 長谷川 友音、松谷 颯太
ブロードリーフ企業賞	μ sight －ひとりでも合奏がしたい! AR 合奏練習アプリ－	神山	正木 忠勝	金谷 尚興、鈴木 結衣、宮野 柊太、山口 空、 メリット キア
アパナード企業賞	無双職人～チームから追放された俺が闇の力 で世界最強に!～	神戸市立	朝倉 義裕	新村 康太、小新堂 叶、崎谷 優
DMM.com 企業賞	そこに AI はあるのか?	徳山	力 規晃	児玉 隆之介、棟近 祐希、山崎 秀太

課題・自由部門について

●課題部門の概要

課題部門では、与えられた課題テーマに沿った独創的なコンピュータソフトウェア作品を募集しています。

今大会は「ICTを活用した環境問題の解決」をテーマとしました。地球温暖化による気象変動は、記録的な集中豪雨、大型台風など、大きな災害をもたらしています。このような年々深刻化する環境問題に対して取り組まなければいけない活動、たとえば、省エネ、食品ロスの削減、公共交通機関の利用促進、森林保護や植林などは、ICTを積極的に活用することでより効果的に実現できる可能性があります。高専生の独創的な発想が、持続可能な地球環境を実現するための一助となることを期待しています。

今大会では、課題部門に40作品の応募をいただき、6月に行われた予選審査において、書類選考によって20作品が選抜されました。これに海外からタイ高専、モンゴルコーセンを加えた23作品が本選に参加します。予選審査では、作品の独創性と課題との適合性が重点的に審査されるため、システムが完成していない設計コンセプトの段階での応募が可能となっています。本選ではこれらのアイデア段階の設計コンセプトがいかにして具体的な作品として実現されたかも審査の重要な項目の一つとなります。

本選では、次の4つのステージで審査されます。

- 1) 学会形式のプレゼンテーションによる審査
- 2) 実際に完成したシステムを動作させて説明するデモンストレーションによる審査
- 3) 操作マニュアルの適正度のチェック
- 4) ソースリストのチェック

いずれのステージでも、独創性をはじめとして有用性・技術力・操作性などが総合的に審査されます。また、プログラミング能力のみならず、プレゼンテーション能力やマニュアル記述力など、総合的に評価する点が、高専プロコンの大きな特色となっています。

●自由部門の概要

自由部門では、参加者の自由な発想で開発された独創的なコンピュータソフトウェア作品を募集しています(第10回～第12回大会は、コンテンツを主体としたコンテンツ部門として実施されました)。

近年、スマートフォンやタブレット端末が普及し、最近ではウェアラブルコンピュータのような新しいデバイスも登場してきました。また、クラウドコンピューティング、オープンデータやビッグデータの利活用、サイバーセキュリティの必要性などインターネットを取り巻く環境も大きく変化してきています。自由部門では、このような社会的背景において、既成の枠にとらわれない自由な発想で考案された独創的な作品を期待しています。過去には、自由部門の優秀作品が情報処理推進機構(IPA)の未踏ソフトウェアに採択されたり、マイクロソフトが主催する学生ITコンテストImagine Cupで優秀な成績を残す等、本コンテストの枠を超えて高い評価を得ており、アイデア創生と熟成の場として大きな役割を果たしていることがうかがえます。

今大会では、自由部門に57作品の応募をいただき、6月に行われた予選審査において、書類選考によって20作品が選抜されました。これに海外からタイのキングモンクット工科大学ラカバン校、シンガポールポリテクニク、新モンゴル高専、タイ高専KMITLを加えた25作品が本選に参加します。予選審査では、課題部門と同様の方法で作品の独創性が重点的に審査されました。

本選審査は、課題部門と同様にプレゼンテーション、デモンストレーション等により、学生のプレゼンテーション能力や作品の完成度等を含めて総合的に優秀な作品が選抜されます。

競技部門について

●競技部門の概要

競技部門は、第5回大会から導入されました。課題部門や自由部門と異なり、各チームの直接対決により勝敗を決します。競技内容は、コンピュータを用いた時間競争、精度競争、最良解探索競争等で、毎年異なるテーマで実施されています。

今年の競技部門では、画像の修復を行います。過去の大会では解を求めるだけでなく、実際に巨大迷路を使用したり巨大パズルを動かしたりして実演する競技や、ネットワークに接続されたコンピュータを利用した競技などが実施されました。そのため、解を求めるアルゴリズムが優れているだけでなく、問題の入力から解答の表示、更にミスへの対処、あるいは自動的なネットワーク通信等、あらゆる面で優れた、完成度の高いシステムが要求されてきました。競技部門には全国高専から58チームが予選を通過し、本選でプログラミング技術を競います。また、NAPROCK 国際大会を同時に開催しますが、海外から新モンゴル高専（モンゴル）、モンゴルコーセン（モンゴル）、タイ高専 KMITL（タイ）、香港 VTC（香港）、ハノイ工業大学（ベトナム）の5チームが参加し、合計63チームで争われます。

●今大会の競技内容

「シン・よみがえれ世界遺産」

今年の競技部門では、文化財の修復をモチーフにした競技を行います。奈良県は日本の世界遺産の宝庫といえる地域の1つです。法隆寺地域の仏教建造物は1993年に日本で最初に世界遺産（文化遺産）に登録されました。ほかにも1998年には古都奈良の文化財が、また2004年に紀伊山地の霊場と参詣道も登録されています。しかしこれらの貴重な文化遺産には天災や人災、また長い間の風雨によりひどく損傷したり当時の色彩を失いかけていたりしたものが多数あります。これらの文化財を保護すると同時に、大規模な修復をおこなって細部に至るまで復元する活動が各地で行われてきました。我々はこれら先人の知恵と努力により復元された珠玉の品々を実際に目にする事ができます。

はじめに複数のピースから構成されるボード（修復前の文化財）と、複数の抜き型（修復道具）が用意されます。抜き型をボードに適用すると抜き型と一致したピースが抜け、抜けたピースを上下左右いずれかの方向から詰めて補います。うまく抜き型を適用して、ボードを最終盤面（修復後の文化財）にします。できるだけ少ない手数かつ早く修復前の文化財を修復後の文化財に「修復」したチームが勝利する競技です。

	0	1	2	3	4	5	6
0	1	0	1	1	2	2	1
1	2	3	1	1	0	0	2
2	3	0	2	1	1	1	1
3	3	0	0	2	2	3	1
4	2	2	3	2	0	2	2
5	3	3	1	0	3	2	3

図1. 元ボードと型抜き位置

	0	1	2	3	4	5	6
0	1	0	1	1	2	2	1
1	2	3	1	1	0	0	2
2	3	0	1	1	1	1	2
3	3	0	2	3	1	0	2
4	2	2	0	2	2	2	3
5	3	3	1	0	3	2	3

図2. 左寄せ型抜き

課題部門本選参加作品

■ 「ICT を活用した環境問題の解決」

発表 番号	タイトル	学校名	指導教員	作成学生
1	ラビビ -Rubbish navigation-	香川 (詫間)	宮崎 貴大	三好 湊稀 (1年)、藤原 瑞月 (1年)、来田 久一郎 (1年)、 高畑 葉響 (1年)、村口 友星 (1年)
2	SDCs -100年続くまちづくり-	舞鶴	井上 泰仁	山本 瑞樹 (3年)、平田 爽馬 (4年)、河本 泰尚 (3年)、 松崎 伶音 (3年)、高岡 優羽 (3年)
3	SEA-VIS -水中ドローンとAIによる調査システム-	沖縄	中平 勝也	山川 響樹 (4年)、西田 皓太 (4年)、當間 一代 (4年)、 仲宗根 由弥 (4年)
4	クラスdeECOウォッチ -環境教育支援システム-	宇部	久保田 良輔	中野 晃聖 (4年)、村谷 凌 (5年)、木村 友亮 (2年)
5	だすばす -農家と家庭をつなげる堆肥共有プラットフォーム-	国際	伊藤 周	江口 太一 (2年)、角 隼 (2年)、五百蔵 蒼天 (1年)、 藤原 聖也 (1年)、中澤 琉月 (2年)
6	Noverflow -「ボーイ捨て」を取り除く-	舞鶴	井上 泰仁	伊勢 巧 (4年)、池田 翔太郎 (4年)、小寺 恭平 (3年)、 森本 柊哉 (3年)、半林 駿典 (3年)
7	まもるん -役場と町民の災害時連携アプリケーション-	神山	春田 麻里	鷹田 葵 (2年)、井上 明 (2年)、梅原 琉 (1年)、 下島 夏 (1年)
8	りょーめん -持続可能な寮食支援システム-	弓削商船	長尾 和彦	井上 裕太 (5年)、黒河 天晴 (5年)、岡田 右京 (3年)、 久保 竜壮 (2年)、福田 煌輝 (1年)
9	EcoMeow	タイ高専	Krittantik Srithanasarn	Nuntanut Poonpayap、Thad Choyrum
10	Smart faucet	モンゴル コーセン	Khurelbat Khishigt	Ganjuurdalai Enkh-Uchral、Ariunbold Zorigtsaikhan
11	Share CARE	タイ高専	Teerapong Orachon	Natthanon Weerapong、Punnawith Sutsukon
12	地賛地頌 -詠もうぜ地元！君の言葉で-	仙台 (広瀬)	安藤 敏彦	司東 晃大 (2年)、村木 祐太 (3年)、千葉 翔太 (3年)、 伊藤 颯汰 (3年)
13	節電だヨ！全員集合 -節電の実践と情報提供で意識を高めるシステム-	阿南	吉田 晋	木下 陽平 (4年)、松本 琉希 (4年)、廣島 承悟 (4年)、 北山 秀磨 (4年)、天羽 優太 (4年)
14	Triplean -インバウンド対応・清掃支援システム-	鳥羽商船	江崎 修央	白川 琥大 (4年)、松葉 勇希 (4年)、北仲 悠人 (3年)、 山下 温斗 (3年)、鮎川 颯 (1年)
15	オトセバ -音によるゴミ分別システム-	大島商船	重本 昌也	神崎 友希 (5年)、西川 陸 (5年)、小松 咲絵 (4年)、 岡本 羽琉夢 (4年)、倉増 風紗 (3年)
16	Excess Informer -買い過ぎ防止システム-	小山	大内 翔平	菅原 晃一郎 (3年)、横井 蛭汰 (5年)、藤沼 心吾 (2年)、 黒田 若葉 (2年)、小堀 咲陽 (2年)
17	プラたべる	大島商船	重本 昌也	縄田 嵐 (2年)、倉本 悠雅 (2年)、車 優月 (2年)、 横道 玲旺 (2年)、宮崎 蘭 (2年)
18	TerraMeal	鳥羽商船	中井 一文	前田 遥 (4年)、田中 映音 (4年)、岡 駿希 (4年)、 糸川 颯人 (1年)、西根 明日翔 (1年)
19	アルCO -地球を守る、優しいアプリ-	奈良	上野 秀剛	崎本 俊治 (4年)、日下 優人 (4年)、池本 拓冬 (4年)、 森岡 大和 (3年)、増井 崇 (3年)
20	金継ぎVR	熊本 (八代)	小島 俊輔	武藤 淳之助 (4年)、山下 遼太 (5年)、盛高 雄寛 (2年)、 脇田 龍之介 (4年)、吉里 彩心 (1年)
21	バイオ炭治郎 -炭素貯蓄編-	松江	村橋 究理基	山崎 巧実 (2年)、岩成 広樹 (2年)、辻 蓮治 (1年)
22	育てるエコタウン -誰でも楽しく環境活動-	熊本 (八代)	小島 俊輔	岩谷 奏慈郎 (2年)、廣野 達徳 (2年)、緒方 亮太 (5年)、 高見 勇斗 (1年)、高山 透也 (5年)
23	POOI -デジタル投票式ゴミ箱-	阿南	太田 健吾	泉 創太 (3年)、井口 蒼生 (3年)、久保田 昇輔 (3年)、 山上 遥己 (3年)、阿瀬川 祥永 (5年)

課題部門 プレゼンテーション審査 タイムテーブル

審査日時 10月19日(土) 10:10 - 16:44

会場 中ホール

発表持ち時間 発表時間 8分 質疑応答 4分(海外チーム 6分) 交代 1分

発表順	発表予定時間	タイトル	学校名
1	10:10 ~ 10:22	ラビビ - Rubbish navigation -	香川(詫間)
2	10:23 ~ 10:35	SDCs - 100年続くまちづくり -	舞鶴
3	10:36 ~ 10:48	SEA-VIS - 水中ドローンと AI による調査システム -	沖縄
4	10:49 ~ 11:01	クラス deECO ウォッチ - 環境教育支援システム -	宇部
11:01 ~ 11:09 休憩 8分			
発表順	発表予定時間	タイトル	学校名
5	11:09 ~ 11:21	だすぼす - 農家と家庭をつなげる堆肥共有プラットフォーム -	国際
6	11:22 ~ 11:34	Noverflow - 「ボイ捨て」を取り除く -	舞鶴
7	11:35 ~ 11:47	まもるん - 役場と町民の災害時連携アプリケーション -	神山
8	11:48 ~ 12:00	りょーメシ - 持続可能な寮食支援システム -	弓削商船
12:00 ~ 13:00 休憩 60分			
発表順	発表予定時間	タイトル	学校名
9	13:00 ~ 13:14	EcoMeow	タイ高専
10	13:15 ~ 13:29	Smart faucet	モンゴルコーセン
11	13:30 ~ 13:44	Share CARE	タイ高専
13:44 ~ 13:53 休憩 9分			
発表順	発表予定時間	タイトル	学校名
12	13:53 ~ 14:05	地賛地頌 - 詠もうぜ地元! 君の言葉で -	仙台(広瀬)
13	14:06 ~ 14:18	節電だヨ! 全員集合 - 節電の実践と情報提供で意識を高めるシステム -	阿南
14	14:19 ~ 14:31	Triplean - インバウンド対応・清掃支援システム -	鳥羽商船
15	14:32 ~ 14:44	オトセパ - 音によるゴミ分別システム -	大島商船
14:44 ~ 14:53 休憩 9分			
発表順	発表予定時間	タイトル	学校名
16	14:53 ~ 15:05	Excess Informer - 買い過ぎ防止システム -	小山
17	15:06 ~ 15:18	プラたべる	大島商船
18	15:19 ~ 15:31	TerraMeal	鳥羽商船
19	15:32 ~ 15:44	アル CO - 地球を守る、優しいアプリ -	奈良
15:44 ~ 15:53 休憩 9分			
発表順	発表予定時間	タイトル	学校名
20	15:53 ~ 16:05	金継ぎ VR	熊本(八代)
21	16:06 ~ 16:18	バイオ炭治郎 - 炭素貯蓄編 -	松江
22	16:19 ~ 16:31	育てるエコタウン - 誰でも楽しく環境活動 -	熊本(八代)
23	16:32 ~ 16:44	POOI - デジタル投票式ゴミ箱 -	阿南

プレゼンテーション審査終了

課題部門デモンストレーション審査&マニュアル審査 タイムテーブル

審査日時 10月20日(日) 9:00～12:01

会 場 中ホールホワイエ

審査時間 7分(説明2分, 質疑応答5分, 移動を含む)

審査時間	課題部門			
	A 班	B 班	C 班	D 班
9:00～9:07	1	7	13	19
9:07～9:14	2	8	14	20
9:14～9:21	3	9	15	21
9:21～9:28	4	10	16	22
9:28～9:35	5	11	17	23
9:35～9:42	6	12	18	1
9:42～9:49	7	13	19	2
9:49～9:56	8	14	20	3
9:56～10:03	9	15	21	4
10:03～10:13	休憩 10分			
10:13～10:20	10	16	22	5
10:20～10:27	11	17	23	6
10:27～10:34	12	18	1	7
10:34～10:41	13	19	2	8
10:41～10:48	14	20	3	9
10:48～10:55	15	21	4	10
10:55～11:02	16	22	5	11
11:02～11:09	17	23	6	12
11:09～11:19	休憩 10分			
11:19～11:26	18	1	7	13
11:26～11:33	19	2	8	14
11:33～11:40	20	3	9	15
11:40～11:47	21	4	10	16
11:47～11:54	22	5	11	17
11:54～12:01	23	6	12	18

注意事項

- ① A 班, B 班, C 班はデモンストレーション審査
D 班はマニュアル審査を示す
- ② 1～23 はプレゼンテーション審査の発表順の作品を表す
- ③ 色が付いているセルは, 海外チームを表す

1

ラビビ Rubbish navigation

香川
(託問)

三好 湊稀 (1年) 藤原 瑞月 (1年)
来田 久一郎 (1年) 高畑 葉響 (1年)
村口 友星 (1年) 宮崎 貴大 (教員)

1. はじめに

近年、世界的に問題となっているゴミのポイ捨て。この問題に対して様々な取り組みを行っている市区町村。環境省の調査では、ポイ捨て問題を解決するため、多くの市区町村が条例を制定するなどの対策を行っています。しかし、誰もがポイ捨てをしてしまう状況が続いています。ポイ捨てする理由はいくつかありますが、その1つに“すでにゴミが落ちているからポイ捨てに抵抗感がなくなる”という心理面での要因があります。その点に着目し、清掃活動を活性化させることでポイ捨てが軽減できるだろうと考えました。

そこで私たちは、マップやAIの機能を組み合わせて、身近な学校生活や地域の中で『どんなところにどんなゴミが多いのか』、『ゴミ箱がどこにあるのか』などがわかり、ポイ捨てに対する意識改善を目標とした Web アプリ「ラビビ」を提案します。

2. 概要

「ラビビ」は清掃活動を活発化させるため、ゲーム感覚でゴミ拾いを進められる Web アプリです。

2.1 ゴミ拾いの概要

ポイ捨てされたゴミを見つけた時、ユーザーは

1. ポイ捨てされたゴミの発見報告
2. ポイ捨てされたゴミの回収報告

の2つの行動を取ることができます。ゴミを発見したユーザーは必ずゴミを拾わないといけないわけではなく、事情によりそのゴミを拾えない場合はボタン一つで発見した報告をすることができます。ゴミを拾いたいユーザーはゴミの発見報告を確認することで効率的にゴミ拾いをすることができます。ゴミの回収報告はゴミの写真を撮ってから回収し、ゴミを捨てたことを報告します。「ラビビ」は発見報告や回収報告のデータを集め、ゴミの分布状況をマップ上に表示します。それによりユーザーはどこにどんなゴミが多いのかなど

を確認でき、より一層効率的にゴミ拾いができると共に、ポイ捨てに対する意識の改善に繋がります。

2.2 ポイントシステムの概要

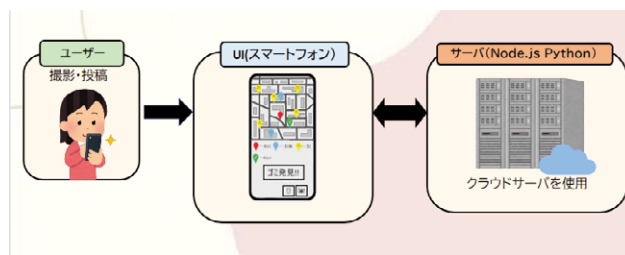
「ラビビ」ではユーザーの行動によってアプリ内で利用できるポイントを獲得できます。ゴミ拾い報告で送られる写真は AI によってゴミの種類や個数が判別され、ゴミの種類や量に応じてポイントを獲得できます。他にもゴミ拾い中に歩いた距離に応じてポイント



を獲得できます。ユーザーの「ラビビ」を利用する意欲を高めるために、「スクラッピー」と呼ばれるキャラを獲得、コレクションすることができます。「スクラッピー」は取得したポイントによる取得や回収したゴミの総量などのランキング特典によって取得が可能です。

3. システム構成

「ラビビ」は Web アプリなのでユーザーは Web ブラウザでゴミやゴミ箱の位置を確認可能で、発見報告やゴミの写真などのデータはサーバーで保存します。



4. おわりに

「ラビビ」によって、身近な場所のポイ捨てゴミが減少し、きれいな、住み続けたい町となることを目指します。

2

SDCs

100 年続くまちづくり

舞鶴

山本 瑞樹 (3 年) 平田 爽馬 (4 年)
河本 泰尚 (3 年) 松崎 伶音 (3 年)
高岡 優羽 (3 年) 井上 泰仁 (教員)

1. はじめに

昨今、地球温暖化を代表とする様々な環境問題について世間の関心が高まりつつあります。しかし、問題を認識していても規模が大きくなればなるほど実感が湧きにくく、問題意識が低いという現状があります。

こうした状況を打破するためには、若い年代への環境教育が不可欠と言え、より意識を高めるための施策やコンテンツ・システムが必要となるでしょう。

そこで私たちは、若い年代の中でも特に小学校高学年に焦点を当てた体験型環境学習システム「SDCs」を提案します。

2. システム概要

SDCs は、環境問題を**触って・見て・考える**ことによって環境に対する意識を高めることのできる教育システムです。

システム中では、まちがあとどのくらい持続するかの指標である持続バロメータを 100 年以上にすべく、まちづくりを行います。

2.1 触る・見る・考える

触る

物理ブロックの駒を使ってまちを実際に形成します。直感的な操作でまちを形成することができます。

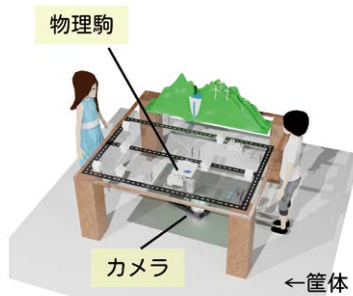


図 1 筐体イメージ

見る

実際に配置された駒を元にまちを映像化し出力します。環境状態により周囲が変化するため、より理解度が向上します。

考える

ただまちを作っていくのではなく、システム上の指示に従って、自分で考えながらまちづくりを行うことで、直面している問題について考え、環境意識向上に繋げることができます。

2.2 システム使用の流れ

本システムは、何もないところから**まちを作る→まちを発展させる→環境問題を解決する**という大きく分けて 3 つのステップによって構成されています。

実際に何もないところからまちを発展させ、環境問題がどのようにして発生するのか、その過程でどのように問題を解決すべきか考えながら学ぶことができるようになっています。

3. 持続バロメータ

本システムでは、100 年後にそのまちが存続しているかの指標を示す持続バロメータがあり、存続年数が 0 年になると、まちが減んでしまいます。

まちを 100 年以上存続させるために、発生した問題に取り組みます。

4. システム構成

本システムは、現実空間と仮想空間の両方を使い、より没入感のあるまちづくりを行うことができます。

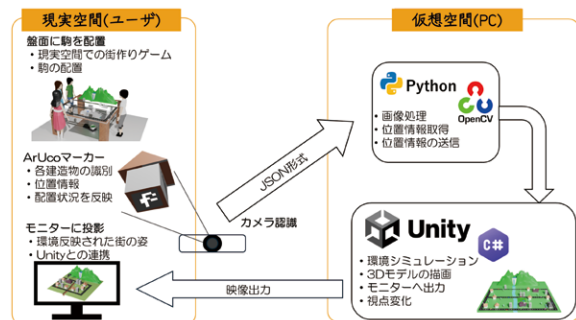


図 2 システム構成

5. 終わりに

まちづくりを通して、発生する環境問題を解決することができる SDCs は環境問題への意識を高め、自ら考えながら行動する第一歩になることを願っています。

3

SEA-VIS

水中ドローンと AI による調査システム

沖縄

山川 響樹（4 年） 西田 皓太（4 年）
 當間 一代（4 年） 仲宗根 由弥（4 年）
 中平 勝也（教員）

1. はじめに

世界 50 万種の海洋動物のうち 4 分の 1 はサンゴ礁域に生息しており、定期的な調査が必須である。従来の調査法は、膨大な人件費や調査精度に問題点がある。この問題を解決する新しい海洋生態系調査システム「SEA-VIS」を提案し、開発と実証実験を行った。

2. SEA-VIS の構成

SEA-VIS は、画像系 AI とロボティクスの 2 つの技術を組み合わせて正確かつ大規模な海洋調査を実現する。

2.1 魚 AI

水槽の金魚を対象とし、yolov8 の画像検出タスクで機械学習をおこなった結果、平均 98% の精度を得られ、基本性能を確かめた。最後に、水族館などで収集した約 3000 枚の教師画像から 40 種類の魚を平均 94% の精度で判別できる魚 AI を完成させた。



図 1. 魚 AI の実行画面

2.2 サンゴ AI

サンゴ AI の開発で必要となるサンゴの良質な教師画像はインターネットには少ない。まずは、3D 仮想環境を用いて良質なサンゴの仮想画像を取得し、基本性能を明らかにした。最後に、yolov8 の画像セグメンテーションタスクで機械学習をおこなった結果、実際の珊瑚礁での取得した画像データを使って検証し平均精度 80% で判別できるサンゴ AI を完成させた。

2.3 ドローン制御

海上/海中ドローンから適切なタイミングと場所で海中画像を取得するため、ドローンを正確に移動させ

る自動制御方式を開発した。海上/海中のドローンは陸上からは直接目視できないため、海中ドローンの位置特定が重要課題である。そこで上空の画像から、AR マーカーを用いてドローンの位置特定を行う。図のようにマーカーの座標から移動方向と目標方向ベクトルを作成し、二つの間の角度が小さくなるようにモーターを制御することで自動制御方式を完成させた。

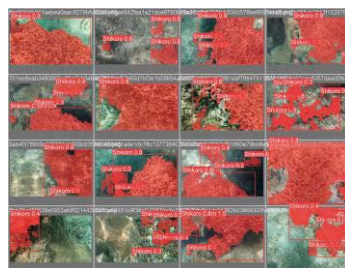


図 2. サンゴ AI の実行画面

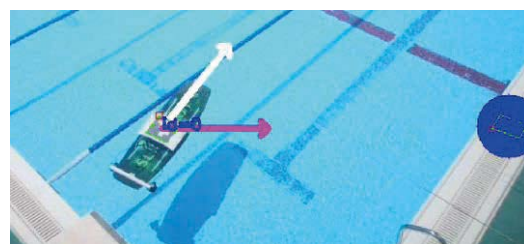


図 3. プールでの水上実験

2.4 実証実験

2024 夏、沖縄各地のサンゴ礁では大規模な白化が起きている。そこで瀬底島のサンゴ礁の状態を SEA-VIS で調査した。サンゴ AI を利用して計算した白化したサンゴの被度は 55% となった。専門家からも未発表の検査を行うことができ、SEA-VIS の大きな成果である。



図 4. 白化サンゴの検出画面（赤 白化サンゴ）

4

クラス deECO ウォッチ 環境教育支援システム

宇部

中野 晃聖（4年） 村谷 凌（5年）
木村 友亮（2年） 久保田 良輔（教員）

1. はじめに

小学校では、新学習指導要領に基づき様々な教科で環境教育が行われており、児童は、資源などの有効利用や環境保全に対する主体的な態度を継続的に学ぶ必要があります。しかし、教育現場では、「授業時間の確保」や「適切な教材やプログラムの準備」などに課題が山積しており、解決が難しい状況です。

私たちは、教室に置くだけで環境教育を支援することができるシステム「クラス deECO ウォッチ」を開発し、その解決に取り組めます。

2. 提案システムの概要

本システムでは、CO₂をどのくらい削減できたかを示す「CO₂削減ポイント」（事前に設定）を導入し、これをクラスごとに集計し、表示します。本システムは、図1に示すように、教室デバイスと個人用アプリ、データサーバーで構成されます。教室デバイスと個人用アプリではそれぞれ CO₂ 削減ポイントが算出・表示され、データサーバーにも送信されます。データサーバーでは、クラスごとの CO₂ 削減ポイントが保存され、個人用アプリで閲覧することができます。

教員は、システム設置後は、必要に応じて児童に適宜アドバイスするだけで良いので、授業時間の確保や教材準備の必要がありません。また、児童は、自身の日々の行動がどれほど CO₂ の削減に寄与できているかを具体的に確認できるので、達成感が得られやすく自己点検や目標設定しやすいという特徴があります。

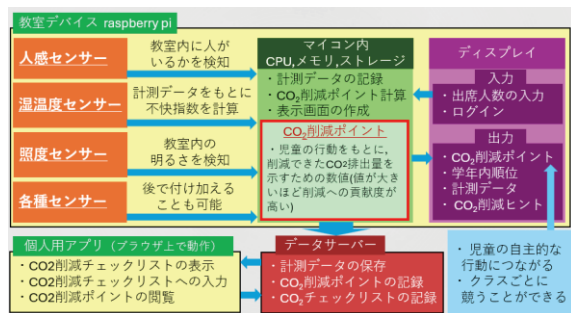


図1：クラス deECO ウォッチの概要と構成

3. 機能

3.1 教室デバイス

設置された教室内の湿・温度、照度などを計測する各種センサーとマイコンを用いて、教室での CO₂ 削減に結びつく行動を自動で検知し、CO₂ 削減ポイントを算出・集計し、記録します。

ディスプレイでは、図2に示すように、当日の CO₂ 削減ポイントや気温、湿度、天気、不快指数、クラス順位、CO₂ 削減へのヒントが表示されます。

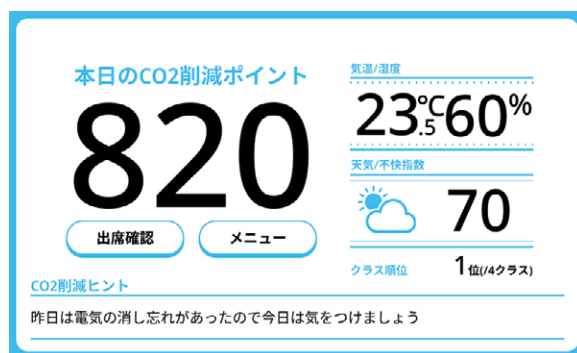


図2：教室デバイスのディスプレイ出力例

3.2 個人用アプリ

事前に作成されたCO₂削減チェックリストを表示し、そのチェックリストの行動後にチェックを入力することで、児童が所属するクラスの CO₂ 削減ポイントに加算されます。また、クラスの CO₂ 削減ポイント数を閲覧することもできます。

4. 提案システムの応用の可能性

4.1 様々な学校や教室で利用できる

教室デバイスは、オフラインでも教室内の CO₂ 削減ポイントを算出・表示できるので、通信環境がない場所でも使用できます。

4.2 計測データの活用

計測された実データは、教室デバイスやデータサーバーから取り出すことが可能であるため、理科や算数などの学習にも活用することができ、児童の意欲的な学習や深い学びに繋げることができます。

5

だすぽす

農家と家庭をつなげる堆肥共有プラットフォーム

国際

江口 太一（2年） 角 隼（2年）
 五百蔵 蒼天（1年） 藤原 聖也（1年）
 中澤 琉月（2年） 伊藤 周（教員）

1. はじめに

家庭から出る燃えるごみの約4割を占め、その処理が環境に与える影響が問題視されてきた生ごみ。そんな中、近年では生ごみをコンポスト（堆肥）にするという考え方が広まってきている。だが、手間やコストがかかり、作った堆肥を活用する人が少ないことから、堆肥づくりの継続率は約3割に留まっている。そこで、我々は一般家庭が生ごみ簡単にコンポストとして有効活用できるシステム「だすぽす」を開発した。

2. 概要

だすぽすはスマートコンポスターとモバイルアプリケーションで構成されるシステムである。コンポスターは自動化により生ごみを堆肥化する手間を省き、アプリはコンポスターとの同期を行う他、作られた堆肥を必要とする農家や家庭菜園を行う人とコンポスターユーザーをつなぐ、情報・物々交換のためのプラットフォームとなる。

3. 使用方法

3.1 コンポスターユーザーの場合

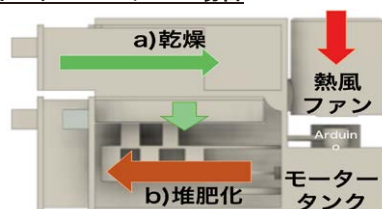
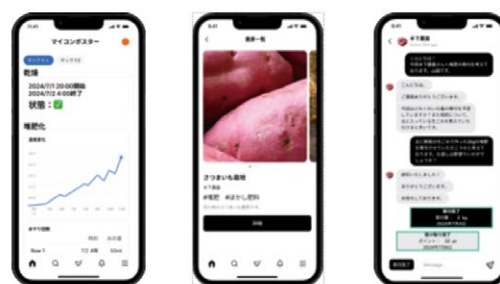


図1 コンポスターの機能

- ・生ごみをコンポスターの乾燥ボックスに入れ、自動乾燥を開始する（図1-a）
- ・一定の時間で乾燥された生ごみが自動で堆肥化ボックスに落ち、堆肥化を開始する（図1-b）
- ・堆肥化の間、温度・湿度をセンシングして自動で水分調整、土のかき混ぜが行われる
- ・堆肥化が進み、内部の温度が下降した段階で、アプリから堆肥化完了のお知らせが来る（図2-a）
- ・堆肥を求める農家をアプリで検索、チャットを通して堆肥譲渡先を決め、堆肥譲渡量を登録後堆肥を送る



a)進捗確認画面 b)農家検索画面 c)チャット画面
 図2 アプリのUI

(図2-b,c)

- ・堆肥譲渡量に応じて農家からポイントを受け取り、貯まったポイントで農家からの返礼品を受け取る

3.2 農家ユーザーの場合

- ・農家のプロフィール、求める堆肥、譲渡可能な返礼品をアプリに登録する
- ・コンポスターユーザーからのチャットを通して受け取りたい堆肥かを確認、堆肥譲渡が決定する
- ・受け取った量の堆肥と登録された堆肥譲渡量を確認し、ポイントを付与する
- ・コンポスターユーザーからのポイントと返礼品の交換希望に応じて返礼品を送る

3. システム構成

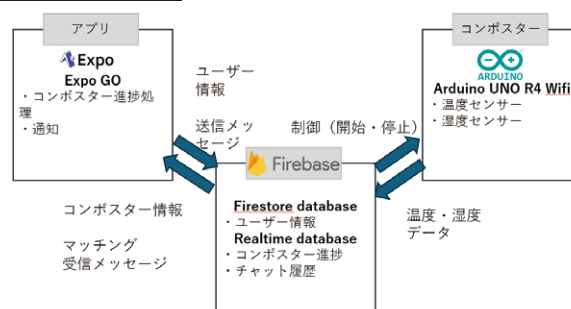


図3 システム構成図

4. おわりに

だすぽすは、この1つのシステムで各家庭が生ごみを活用して気軽に環境問題の解決に取り組みようとする他、農家のより良い農作物づくりを助けることも目指している。

6

Noverflow

「ポイ捨て」を取り除く

舞鶴

伊勢 巧（4年） 池田 翔太郎（4年）
小寺 恭平（3年） 森本 柊哉（3年）
半林 駿典（3年） 井上 泰仁（教員）

1. はじめに

街中を歩いていると、路上にポイ捨てされたゴミが目に入ったことのある人は多いと思う。また、駅内やコンビニ等に設置しているゴミ箱からゴミがあふれている状態を目にしたことがある人も多いだろう。これは、路上に設置しているゴミ箱が家庭ゴミ（ここでは分別されていないゴミを指す）の持ち込み、管理の難しさなどの理由から減少傾向にある為である。

これらの事からペットボトル、プラスチック容器の使用量は増加しているが、それらを処分する機会は、失われつつあるということが言える。

そこで私たちは、利用者は確実にゴミを処理する事ができ、ゴミ箱の管理者の負担にならず、家庭ゴミ等の関係のないゴミを捨てる事ができないゴミ箱、「Noverflow」を提案する。

2. 概要

「Noverflow」は管理者側のゴミ箱型ハード、利用者側のスマホアプリから構成される。主な特徴として、ゴミ箱内の空き容量を計測し、ゴミの偏りを防ぐ。開閉口が認証式の為、対象外のゴミは捨てられない。が挙げられる。

2.1 機能

ハードは空き容量の計測、ゴミの種類判別を行い、アプリは最適なハードへの誘導、認証用のQRコードの表示（QRコードは時間制限付）、環境への貢献度の表示（環境への貢献度とは一人当たりの環境への働きを指し、プラスチックを例に挙げると一人当たりの年間のプラスチックの使用量から一か月当たりの使用量を算出し、撮影されたゴミの重量の総量との比較を行い、貢献度を算出する。）を行う。

2.2 動作手順

スマホにて捨てたいゴミの種類を選択し、種類から周辺の最適なゴミ箱に誘導、QRコードの表示。

ハードにてQRコードで認証完了の後、投入口が開き、ゴミを判別層に投入可能に。判別層内のゴミの種類を画像処理にて判別。正しければ集積層に。処理完了。

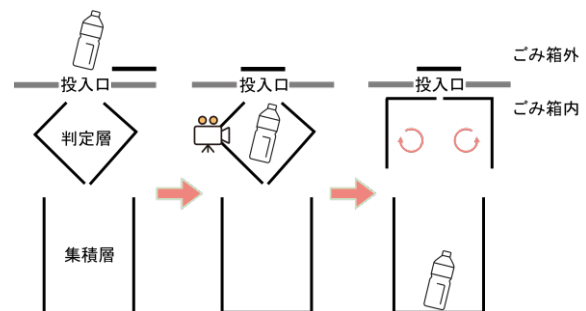


図1 ハードを横から見た図

3. システム構成

ゴミ箱型ハードは Raspberry Pi で、画像認識は OpenCV を使用して作成、アプリは androidstudio で作成した。また、それらのデータのやり取りのためのサーバーは Firebase、django を用いて作成する。

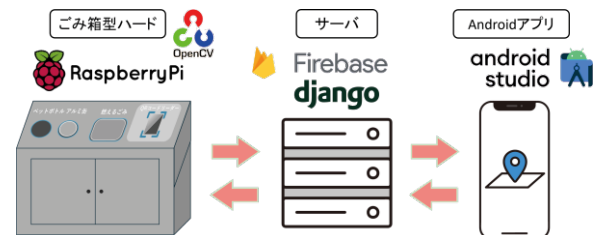


図2 システム構成

4. おわりに

「Noverflow」によって路上で目に入るゴミが減り、気持ちよく街中を歩ける社会を目指します。

7

まもるん

役場と町民の災害時連携アプリケーション

神山

薦田 葵（2年） 井上 明（2年）
梅原 琉（1年） 下島 夏（1年）
春田 麻里（教員）

1. はじめに

昨今、地球温暖化による異常気象の増加に伴い、自然災害への対応が求められています。特に過疎化と高齢化が進む山間地域では、災害時に住民が適切に避難することが困難であり、役場と住民が連携して災害対策を行うことが重要です。そこで私たちは、住民と役場が協力して災害時に迅速かつ安全に避難できるようサポートするアプリケーション「まもるん」を開発しました。このアプリは、安否確認、避難誘導、危険箇所共有などの機能を備え、地域住民の安全を守ることを目指しています。

2. 機能概要

「まもるん」は災害時に町民が迅速かつ安全に避難をするために、役場と連携して町民をアシストするアプリケーションです。災害時に自分の様子を役場に伝えることができる「安否確認」、利用者に安全な避難経路を示す「避難誘導」、危険な場所を町民や役場と共有できる「危険箇所共有」の三つの機能を兼ね備えています。

2.1 安否確認

災害時に選択肢から選ぶ形式で、高齢者でも簡単に役場へ自分の状態を知らせることができます。役場側は町民の名前、安否情報、位置情報を知ることによって救助隊に状況を伝えることができ、安全かつ迅速な救助に繋がられます（図1）。

地図情報		安否情報	
五十音順 年齢順	名前	安否情報	位置情報
	名前	安否情報	位置情報
	名前	安否情報	位置情報
	名前	安否情報	位置情報
	名前	安否情報	位置情報
	名前	安否情報	位置情報
	名前	安否情報	位置情報

図1.安否確認役場側UI

2.2 避難誘導

役場が町民を安全かつ迅速に避難することができますようにサポートすることができます。

利用者の位置情報や災害の情報をもとに避難するのに最適な道を提案し、町民に最適な避難経路を提示します。役場側ではどこに危険なものがあるかを設定し町民にアプリを通して伝えることができます。

2.3 危険箇所共有

危険箇所を発見、送信し共有することで、迅速な情報共有や減災に繋げることができます。

町民は町内で災害が起きた際に危ない場所（木が倒れそうなど）の情報を見つけた際写真を取って位置情報と共に送信します。町民と役場側どちらともから危険な場所を知ることができ危険箇所を知り合うことができます。

3. システム構成

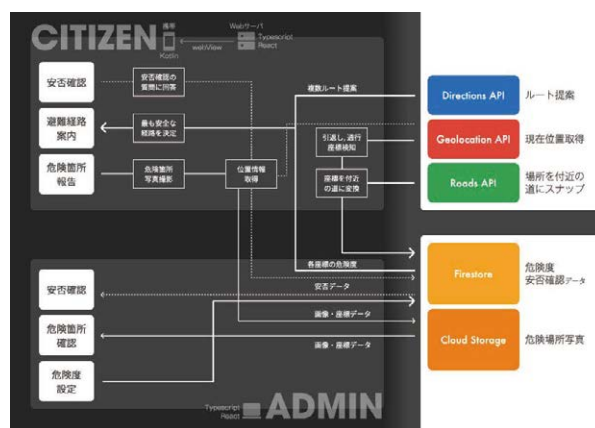


図2.システム構成

4. 終わりに

「まもるん」は町民側と役場側のどちらからでも危険な箇所を知ることができます。このアプリを通じて、町民の皆様が災害への備えや安全意識をさらに高めるきっかけとなることを期待しています。

8

りょーめし

持続可能な寮食支援システム

弓削商船

井上 裕太（5年） 黒河 天晴（5年）
岡田 右京（3年） 久保 竜壮（2年）
福田 煌輝（1年） 長尾 和彦（教員）

1. はじめに

寮食堂では、作りすぎや食べ残しによるフードロスが問題となっています。そのため、利用者数の予測や料理の量の調整、利用者数を増やすなどして、フードロスを削減する必要があります。そこで、私たちは、フードロスの削減と寮食堂のサービスを向上させる「りょーめし」を開発しました。

2. 概要

本システムでは、食堂の利用者数を予測することによって、作り過ぎの削減を行います。また、学生証を利用して個人を識別することにより、アレルギーや量の調節などの対応をしやすいです。本システムは、学生用と食堂用に分かれています。学生用では、献立表示や食堂を利用できるかどうかを調べる事前アンケート、フィードバックを得るための食後アンケートの機能があります。食堂用では、表示する献立とアレルギー情報の入力や検食の結果の入力ができます。さらに、食堂のカウンターでのアレルギー対応の表示や予測した利用者数を基にした提供数の決定も行います。

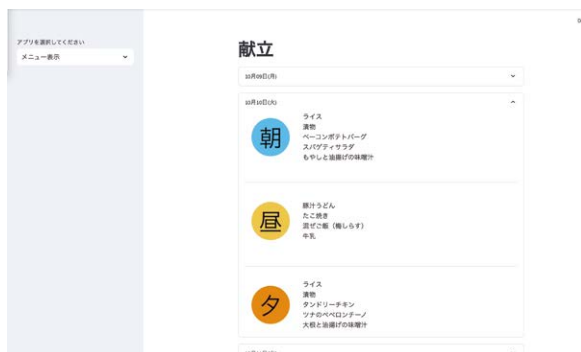


図1 学生用の献立表示ページ

2.1 利用者数の予測機能

過去の献立や利用者数、曜日、天気などの統計データを用いて、それぞれのデータがどのくらい食堂の利用率に影響しているかどうか重み付けをして機械学習を行います。データベースに登録されている献立や曜

日、天気などのデータを取得し、利用者数を予測します。また、事前アンケートの結果を考慮することで予測をより高精度にできるようにしています。

2.2 学生証による個人の識別機能

ICカードリーダーを食堂に設置し、学生証をスキャンすることで学籍番号を読み取ります。学籍番号からデータベース内の学生データを取得し、個人の識別を行います。個人を識別することで、それぞれの学生データを基に食堂側の端末に顔写真やアレルギー情報を表示できます。欠食申請をしている学生や、二度目の利用のような不正利用を検知することもできます。さらに、学生証が読み取られた回数から、食堂の利用者数の集計も可能です。

3. システム構成

図2は、りょーめしのシステム構成を示したものです。過去データや事前アンケートの結果、学生などの情報はデータベースに登録されます。食堂側のアプリから入力された献立データもデータベースに登録されます。これらのデータは、利用者数の予測や献立表示にデータベースから取り出して使用されます。

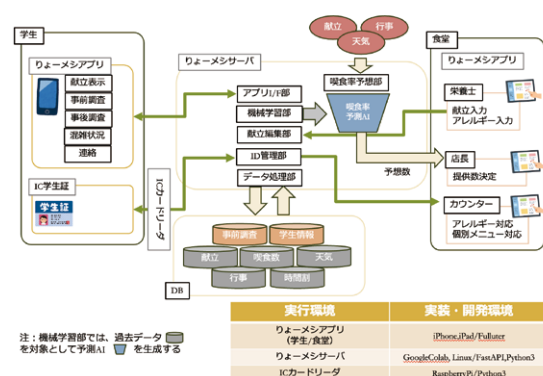


図2 システム構成図

4. まとめ

りょーめしは食堂の利用者数の予測や、学生証を用いた個人の識別機能によって、食べ残しや作り過ぎによるフードロスの削減とともに、寮食堂のサービスを向上させます。

Introduction

The rising levels of CO₂ due to human activities are accelerating climate change, leading to more frequent natural disasters and environmental degradation. In response, we propose "EcoMeow", a software system designed to predict CO₂ emissions and energy consumption using advanced machine learning techniques and real-time cloud computing. Additionally, the system features an interactive cat mascot to engage and educate users on global warming and sustainability. Combining prediction models, real-time data, and educational content through an interactive mascot into one platform makes EcoMeow different from others.

Methodology

1. Software operations and Machine Learning Predictions

- Data Sources: EcoMeow will use Python scripts to scrape and clean data from sources like Worldometer, NASA, and IPCC reports, focusing on CO₂ levels, energy consumption, population growth, deforestation rates, and other environmental indicators.
- Machine Learning Models: The software will employ Polynomial Regression and Random Forest Regression, chosen for their computational efficiency and ability to balance underfitting and overfitting, making them suitable for predicting CO₂ emissions and energy consumption.
- Environment: The model is deployed on Oracle Cloud Infrastructure, using a virtual machine with 4 ARM cores and 24 GB RAM. Testing will be done in a controlled cloud environment, running continuously to capture various environmental data patterns over a 24-hour period.
- Technical Innovations: Containerization will ensure a modular, scalable, and efficient software stack, with real-time data analysis and predictive modeling displayed for future planning (e.g., 2030, 2050).

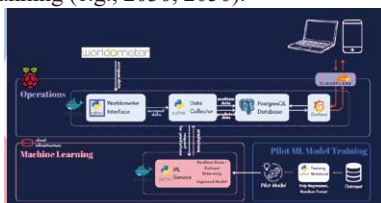


Figure 1, A simplified flowchart of the software stack.

2. Character Design

- Eko Eko Chan: A user-friendly cat mascot designed to engage the public and make the software more accessible.
- UX/UI Design: The design focuses on creating a seamless user experience, with intuitive interfaces and visually appealing elements.
- User Interaction: The mascot's role in interacting with the data and guiding users through the platform enhances the system's educational impact and ease of use.



Figure 2 & 3, Eko Eko Chan and Logo for EcoMeow.

3. Real-Time Monitoring

- Docker for containerization, PostgreSQL for database management, and d3.js for dashboard visualization (Grafana is used as a temporary visualization tool for demonstration).
- The software will predict and display critical parameters, including total and real-time CO₂ emissions, population growth, fossil and renewable energy consumption, deforestation rates, and projected global temperatures for set future dates.
- A user-friendly dashboard will present these predictions, making them accessible to the public and organizations. The dashboard will allow users to interact with the data and understand the implications of different environmental trends.



Figure 4, The prototype of the dashboard using Grafana.

Expected Outcomes

- Monitoring and Accuracy: Continuous monitoring compares predictions with actual data, showing over 90% accuracy for CO₂ emissions, global warming, and energy consumption.
- Informed Decision-Making: Supports organizations and governments in creating effective climate strategies based on reliable data.
- Public Engagement: Enhances awareness and promotes sustainability through interactions with a mascot chatbot, encouraging active public participation.

Conclusion

"EcoMeow" leverages machine learning and real-time data to deliver accurate environmental predictions, supporting sustainability planning and CO₂ management. Its interactive mascot engages the public, promoting sustainability. Designed for governments, organizations, and the public, EcoMeow plays a crucial role in global efforts against climate change.

10 Smart faucet

モンゴル コーセン

Ganjuurdalai Enkh-Uchral
Ariunbold Zorigtsaikhan
Khurelbat Khishigt (教員)

1. Introduction

Water is fundamental to our existence. Yet, as humans, we often waste it without much thought. Water is the root of all life, and preserving it starts with us—specifically, with how we manage water usage in our homes.

However, there's a challenge: our habits. Changing how we use water is difficult because our brains are not accustomed to such shifts. This is where the concept of a “smart faucet” comes into play. With water consumption at unsustainable levels, we need simple solutions to reduce usage—not eliminate it, but make it more efficient and responsible.

2. Development progress

2.1 Briefly

We’ve developed an app that engages users to be more conscious of their water usage. At the center is a cute water droplet character that interacts with users, offering tips and reminders to manage their water consumption more responsibly. The app is designed not only to inform but also to help users build lasting water-saving habits. By making conservation fun and interactive, the app encourages small daily changes that add up, ultimately helping to preserve water and promote sustainability in an enjoyable, user-friendly way.

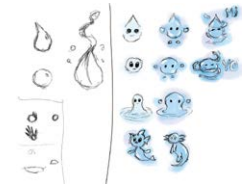
2.2 System

We are diving into learning Android Studio, Arduino Nano, React Native, and Blender for the first time. It's an exciting journey as we explore these tools, expanding our skills and pushing our boundaries to bring our ideas to life.

- Visual studio
- Android studio
- React-Native
- Arduino Nano
- Blender

2.3 About our main character

The character is a cute little droplet that shares helpful information like this:



- A breakdown of your daily water usage
- Comparing your water consumption to the average
- Smart tips for using water efficiently in your daily routine

3. Conclusion

With our application, we aim to encourage people to adopt new water-saving habits that will lead to a significant reduction in water usage. By helping individuals become more mindful of their consumption, we hope to make a positive impact on the environment and contribute to a sustainable future for our world.

11

Share CArE

タイ高専

Natthanon Weerapong
Punnawith Sutisukon
Teerapong Orachon (教員)

※出場辞退のため掲載無し

12

地賛地頌

詠もうぜ地元！君の言葉で

仙台
(広瀬)司東 晃大 (2年) 村木 祐太 (3年)
千葉 翔太 (3年) 伊藤 颯汰 (3年)
安藤 敏彦 (教員)

1. はじめに

地産地消は地域経済を活性化させ、その地域に暮らす方々の雇用機会が増えたり、地域の経済面を安定させたりすることができます。そのため、現在の環境問題の一つである地域経済の停滞の解決につながります。

しかし、地産地消を意識している人の割合は高いとは言えず、課題解決のためにはより多くの方々にその地域のお店や特産物、そして地域の状態に興味を持ってもらう必要があります。そこで、地域の魅力を伝えあうことができ、言葉のリズムを楽しむことができる「詩」で地域の方々の心を動かし、地産地消を活発化する。それが、当システム「地賛地頌」です。

2. システム概要

2.1 地図上でお店を検索

ユーザーは地図上で付近のお店を検索することが可能です。地図にはその土地で地産地消を行っているお店やその土地の特産品がそれぞれ同時にピンで表示されます。ピンをタップすることでお店の情報や画像、さらにはお店側からの宣伝やお店に対するコメントなど、詳しい情報を閲覧することができます。宣伝やコメントは詩で表現され、システムが読み上げることができます。

2.2 詩を作成・投稿

お店の宣伝やコメントなどを投稿する際にユーザー

は詩を作成します。まず俳句・川柳、短歌、短詩から作成する詩の種類を選びます。次に読み上げた際の音程やイントネーションのパターンを選択し、デフォルトの設定から自分好みに調整します。そのため、字余りや字足らずも表現できます。

2.3 コミュニティーチャット

その地域内の生産者・飲食店・消費者の方がお互いに情報を共有することができます。例えば、生産者の方が収穫前の野菜や野菜の収穫風景について投稿したり、飲食店を経営されている方は地元で採れた野菜の新メニューについて投稿したりすることができます。

地域内でコミュニケーションをとることができ、地域で地産地消を取り組むきっかけにつながります。また、ユーザーがその地域の特産品やお店について知る機会にもなります。

3. システム構成

本システムはweb サービスになっています。地図やピンの表示には「Google Map API」を用いており、詩を詠みあげる際の音声データの入手には「VOICEVOX API」を用いています。ユーザーは詩の投稿などにおいて、本システムにデータを入力し、入力されたデータはサーバーに保存されます。保存されたデータは本システムを通して他のユーザーにお店の宣伝やコメントとして見るすることができます。

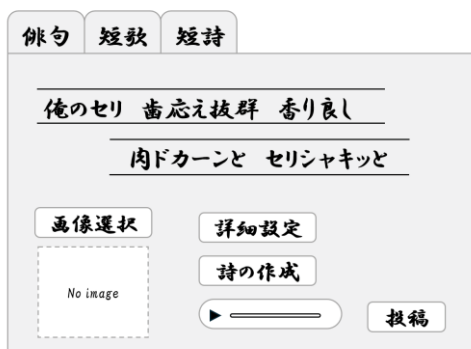


図 1 詩の作成画面

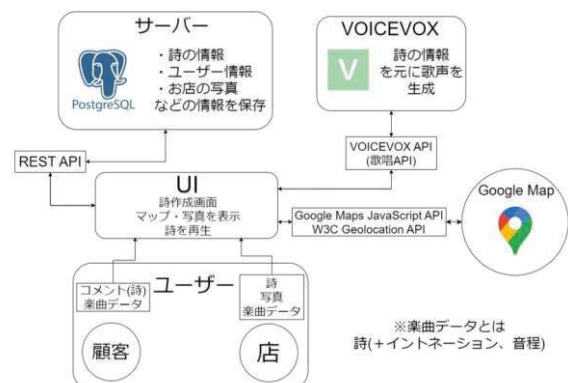


図 2 システム構成図

13

節電だヨ！全員集合

節電の実践と情報提供で意識を高めるシステム

阿南

木下 陽平（4年） 松本 琉希（4年）
 廣島 承悟（4年） 北山 秀磨（4年）
 天羽 優太（4年） 吉田 晋（教員）

1. はじめに

持続可能な未来のために地球温暖化を抑えることが重要視されています。地球温暖化を抑える方法の1つとして消費電力の削減、つまり節電が挙げられます。

消費電力の多い身近な施設である私たちの学校を調査したところ、この3年間で学校の全出費に対する電気代の割合が約2倍まで増えており、節電は大きな課題で、節電のために節電啓蒙ポスターの掲示などを行っていることが分かりました。しかし、節電の効果が見られません。そこで学生への節電の意識調査を実施すると、学生の節電意識が低いことが分かりました。そこで学校を対象とし、学生自ら節電を実践し、節電意識を高めることが出来るシステム「節電だヨ！教室集合」を提案します。

2. システム概要

本システムは図1のシステム構成に示すように、センサデバイスとWebアプリケーションを組み合わせた節電を見える化イベントとして取り組めるシステムです。

2.1. 節電状況の見える化

M5Stackと温度センサとM5Cameraを用いた画像認識AI技術を用いてクラスごとの空調、照明の使用状況と、教室にいる学生数を把握し、学生一人当たりの使用量をWebブラウザ上で可視化して表示して学生に節電状況を知ってもらいます。

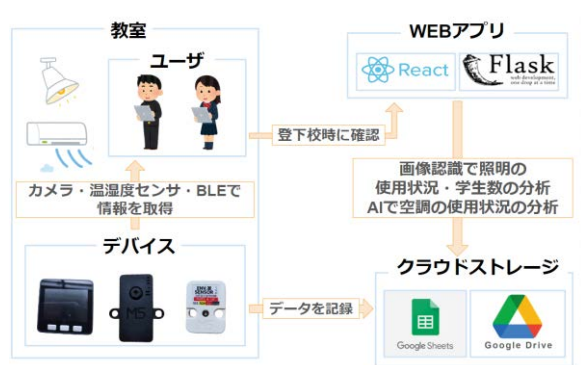


図1 システム構成

2.2. 競技性・イベント性

節電効果を可視化することで、クラス単位や学年単位で、みんなで協力する節電イベントが実施可能となります。夏や冬の消費電力が多い期間に節電イベントを開催し、学年対抗で節電を競うことで、節電意識を高めます。

2.3. 教室に集まる仕組み

放課後、各教室にそれぞれ少人数で残って空調を使い続ける問題を改善することを目的とし、少人数の教室の学生を指定の教室に集め、空調を使用する教室を減らす節電イベント開催を可能とする工夫をしたWebシステムを構築しています。

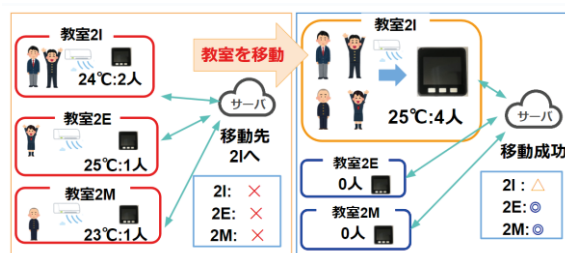


図2 教室移動の検出と評価例

3. 特徴

このシステムは節電状況の見える化に加えて、競技性・イベント性、集まる仕組みを実装しているのが特徴です。これらの仕組みから節電意識の向上を見込むことができます。

また、学校の設備は古く、数も多いので設備の更新が難しく、空調管理システムの設置費用も高額になります。本システムでは、M5Stackを使用することで費用を少なく、また古い空調システムにも設置することが出来ます。

4. まとめ

「節電だヨ！教室集合」は、目的を節電意識の向上とした、学校向けの節電システムです。見える化、イベント性、集まる仕組みを用いることで、この目的を達成します。

14

Triplean
インバウンド対応・清掃支援システム

鳥羽商船

白川 琥大（4年） 松葉 勇希（4年）
北仲 悠人（3年） 山下 温斗（3年）
鮎川 颯（1年） 江崎 修央（教員）

1. はじめに

訪日観光客は過去最高を記録するなど、観光産業の発展と地域自治体の活性化に大きく貢献しています。しかし、多言語対応や地域のインバウンド受け入れ態勢が不十分であり、一部地域ではオーバーツーリズムが問題となっています。そこで私たちは、今後も増加を続ける観光客向けに快適な旅行環境を提供するとともに、新たな地域のインバウンド受け入れを支援する「Triplean」を提案します。

2. システム概要

「Triplean」は、地方の公共トイレやゴミ箱などの設備を効率的に清掃・管理するシステムです。観光客向けにトイレ、ゴミ箱、喫煙所などの情報提供を行う「①施設検索アプリ」、地域住民から清掃員を募集する「②清掃募集アプリ」、汚れた施設を検出する「③施設状態分析機能」から成ります。なおこのシステムは、利用者である訪日観光客から得られる寄付を元に、清掃作業の対価を支払うビジネスモデルで成立させます。



図1 システム概要図

3. 実施する機能

3.1 施設検索アプリ

安心して観光するための施設（トイレ、ゴミ箱、喫煙所）の位置と状態をマップに表示し、各施設の混雑状況、清掃状況、詳細情報（洋式・和式、バリアフリー対応など）を多言語で提供します。施設利用後に評価と寄付を募り、国際的に利用可能な決済システムにより、簡単に寄付できる仕組みを整えます。集まった

寄付金は施設の維持管理に活用することで、観光客の快適性向上と地域の環境維持を両立させます。

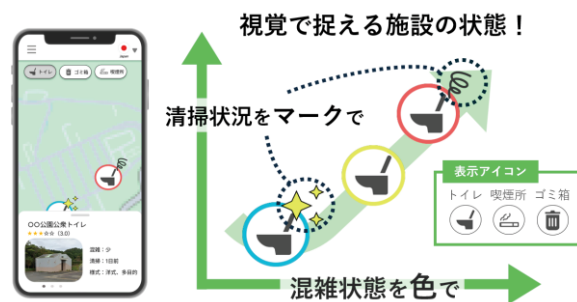


図2 施設検索アプリの画面

3.2 清掃募集アプリ

重量センサや距離センサを活用し、利用頻度や消耗品の状態を収集するだけでなく、寄付金額と評価データを分析して清掃需要を予測します。清掃員募集は地域住民の位置情報やユーザー清掃評価をもとに、効率的に仕事を割り振ります。地域住民は、アプリで清掃依頼を受け取り、清掃作業を受託します。地域住民の清掃評価は作業姿勢、作業時間、訪日観光客の満足度から総合的に算出され、次回以降の報酬額や仕事の割振に反映し、質の高いサービス提供を促進します。



図3 清掃募集アプリの画面

4. おわりに

このシステムにより、施設の持続的な維持管理を効果的に実現させます。観光客の満足度向上と地域経済の活性化を同時に達成し、快適で持続可能な観光地づくりに貢献します。

1. はじめに

環境問題の解決において、ゴミを適切に分別することが重要です。正確にゴミを分別するには、IoTを用いたゴミの自動分類が有効的です。

現在、画像認識を用いた自動分類手法がありますが、画像ではゴミの中身まで判別することはできません。中身の有無がわからず分別してしまうと、例えばスプレー缶のようなエアゾール製品は処分する際に、ガス爆発のような危険を及ぼす恐れがあります。

私たちは、ゴミ容器の中身があるときとないときで、叩いたときの“音”が違うことに着目しました。この音の違いをディープラーニングで判別してゴミを分別する新しいゴミ箱『オトセバ』を提案します。

2. システム構成

オトセバでは、ゴミを叩いた時の音をディープラーニングで解析して分別し、その情報をデータベースに保存して管理者がゴミ箱の状況を確認することができるシステムです(図1)。

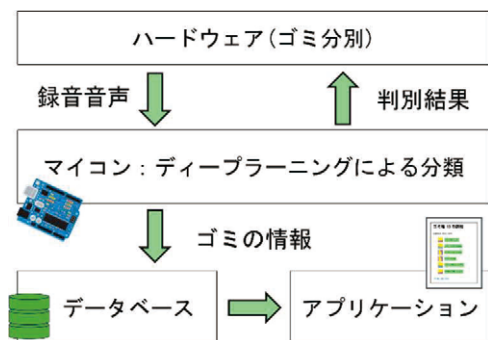


図1. システム概要

2-1. ハードウェア

本システムのハードウェアは録音装置と分別装置で構成されています(図2)。録音装置では、JetsonNanoを用い、ゴミが捨てられたことを赤外線距離モジュールで検知後、鉄の棒でゴミを叩いて音を出し、その音を読み取ってゴミの種類を判別します。分別装置では、Arduinoを使用してサーボモーターを制御し、板を回転させることで正確なゴミの分別を行います。

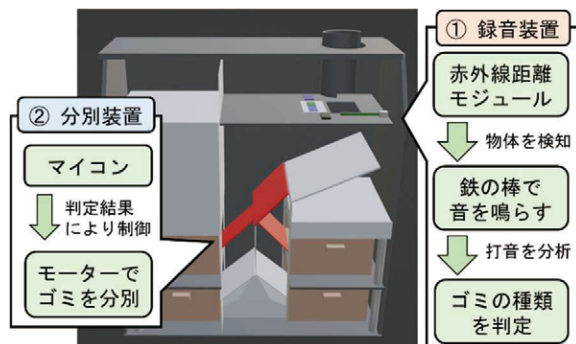


図2. ハードウェア構成

録音装置と分別装置は、JetsonNano と Arduino をシリアル通信することで接続します。

2-2. 音を用いた分類

音の分類手順を図3に示します。スプレー缶や瓶などのゴミを叩いた時の音を中身のありとなしの状態で音声データ(1s)をディープラーニングにより事前学習します。テスト用の音声データを用いて正しく分類できるか検証を重ねて、完成した分類器を JetsonNano に導入してゴミの分別を行います。



図3. 音を用いた分類

2-3. アプリケーション

アプリケーションでは Python を使用し、データベースに保存されるゴミ箱 ID の一覧が表示され、ゴミ箱 ID をタップするとそのゴミ箱の内部情報を確認することができます。

3. おわりに

『オトセバ』では、画像では見えないゴミの中身を、叩いたときの音を用いて分別することで、安全かつ正確にゴミを分別・回収できるシステムを目指します。

16

Excess Informer
買い過ぎ防止システム

小山

菅原 晃一郎 (3年) 横井 蛭汰 (5年)
藤沼 心吾 (2年) 黒田 若葉 (2年)
小堀 咲陽 (2年) 大内 翔平 (教員)

1. はじめに

食材を買ってきたが食べきれずに捨ててしまった経験はありませんか。買いすぎなど買ったものをそのまま捨ててしまう直接廃棄は食品ロスの内4割を占めています。また、食品ロスにより年間1500万tもの温室効果ガスが排出され地球温暖化の要因となっています。そこで私たちは買い物時、購入前に買いすぎを防止するシステム「Excess Informer」を提案し、家庭での廃棄や無駄な出費を減らすことに貢献します。

2. システム構成

Excess Informer では、買い物かごにスマートフォン、冷蔵庫にカメラを設置することで食材の種類、個数を検出し冷蔵庫内の食材リストを作成する管理機能、作成した食材リストを使用して買いすぎを防止する警告機能、賞味/消費期限を通知する通知機能、買いすぎ防止の貢献度を可視化するポイント機能の4つの機能を提供します。

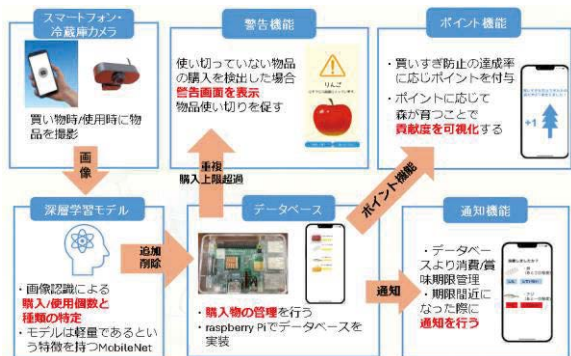


図1. Excess Informer のシステム構成

3. 機能

3.1 管理機能

スマートフォンの Excess Informer アプリで買い物を開始すると、カメラ映像より画像認識 AI を用いてカゴに入れる食材の種類と個数を検知して冷蔵庫内の食材リストを作成します。作成された食材リストはアプリ内で閲覧することができます。冷蔵庫にもカメラを設置し、食材の出入りを画像認識 AI より検知する

ことで食材の出し入れに応じて食材リストからの自動削除も可能となります。また、文字認識 AI により賞味/消費期限も同時に食材リストに情報として追加します。

3.2 警告機能

買い物時に Excess Informer に検知された食材がすでに冷蔵庫内食材リストに存在している場合、買いすぎを警告する警告画面を表示し、食材購入前に買いすぎを防止します。



図2. 警告画面例

3.3 通知機能

購入時に文字認識 AI を使用することで賞味/消費期限を取得し食材リストの情報として追加されます。取得した期限が近くなった食材に対してリマインドの通知を行います。通知により、買った食材の使い切りを促し食材の廃棄を減らします。

3.4 ポイント機能

警告機能により重複購入をやめた場合等買いすぎ防止に貢献した場合、ポイントを付与します。ポイントが貯まるとポイントに応じてアプリ内で木が生えるという形で貢献度の可視化を行います。可視化により環境貢献度を見える化し、システム利用継続を促します。



図3. 通知・ポイント機能画面例

4. おわりに

Excess Informer を使用することで、食品ロスの一因である買いすぎを購入する前の段階で防ぎます。家庭内では無駄な出費やゴミが削減され、その結果として環境問題の解決に貢献することができます。

17 プラたべる

大島商船

縄田 嵐 (2年) 倉本 悠雅 (2年)
車 優月 (2年) 横道 玲旺 (2年)
宮崎 蘭 (2年) 重本 昌也 (教員)

1. はじめに

昨今、マイクロプラスチックの環境への影響が問題視されています。マイクロプラスチックは、プラスチック製品が海中で劣化することにより発生する微小なプラスチックの総称であり、水中に浮かぶこれらを魚などが摂取することによる問題が多く報告されています。そこで私たちは、マイクロプラスチックの回収と教育活動を組み合わせることで、回収を「地域の負担」から「楽しい」ものへと変える新しいシステム『プラたべる』を提案します。

2. システム概要

回収装置を使用してマイクロプラスチックを効率的に回収し、そのデータを基にゲームを楽しむシステムです。装置では砂と水を攪拌し、マイクロプラスチックを分離・回収します。収集したプラスチックはディープラーニングで詳細に解析され、キャラクターの育成などに活用されます。ゲームと回収活動が連携し、楽しみながら環境保護に貢献することができます。

2.1. 育成



図1：プレイ画面

回収装置で集めたプラスチックの色に応じて、ゲーム内ポイントが増加します。このポイントでキャラクターを育成し、育成に使用したポイントの色によって攻撃力・防御力・体力が成長します。キャラクターが

一定のレベルに達すると、二段階の変化があり、新しい姿に変化します。変化先はプレイヤーが選べます。

3. システム構成

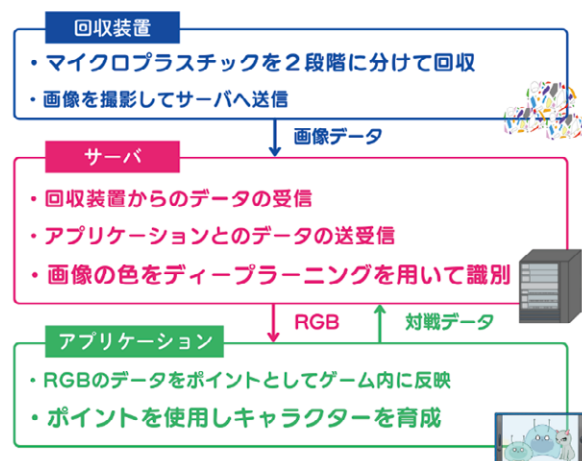


図2：システム構成

3.1. ハードウェア

回収した砂と水を装置に投入し、攪拌することでマイクロプラスチックを効率的に分離します。比重が軽いマイクロプラスチックは、攪拌によって浮き上がるため、簡単に回収することができます。さらに、沈んでいる微細な部分も、ろ過機能を使ってしっかりと取り出すことが可能です。また、装置に設置されたカメラとディープラーニング技術を活用し、画像内の色情報を高精度で解析します。これにより、画像内の R, G, B の値を正確に算出し、マイクロプラスチックの色を詳細に把握することができます。

4. まとめ

プラたべるは、海のプラスチックを回収しながら、小さな子どもたちでも楽しみつつ環境問題に取り組むことができるシステムです。楽しみながら環境保護に対する理解を深めることができ、ゲームを通じてその成果を実感できます。みなさんも、環境問題への意識を高めてみませんか。

18 TerraMeal

鳥羽商船

前田 遥 (4年) 田中 映音 (4年)
岡 駿希 (4年) 糸川 颯人 (1年)
西根 明日翔 (1年) 中井 一文 (教員)

1. はじめに

日本の焼却ごみの約 40%を生ごみが占めています。さらに、生ごみの重量の約 80%は水分であるため、焼却時のエネルギーロスが大きいです。また、化学肥料の価格高騰により、農業への影響が懸念されています。

そこで、持続可能な農業の実現のため、生ごみを堆肥化して完成した堆肥を活用するシステム「TerraMeal」を提案します。

2. システム概要

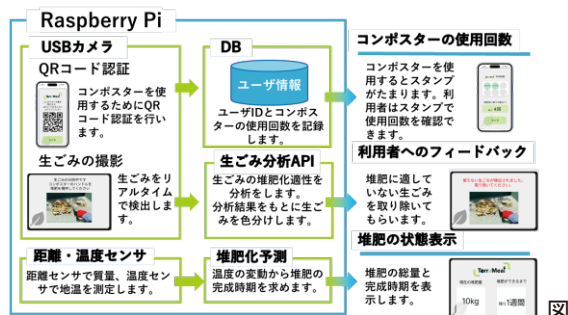
本システムは、画像解析を用いて利用者の家庭から出た生ごみを分別して堆肥を作り、利用者への堆肥の受け渡しを支援するコンポスターです。

これにより、生ごみを堆肥化する際に発生する、「堆肥に適さない生ごみを入れてしまうことで、植物や土壌に悪影響を及ぼす」という問題を解決します。

3. システム構成

本システムは USB カメラ、距離・温度センサ、Raspberry Pi で構成されています。

機能は主に 3 つあります。生ごみが堆肥に適しているか分析し、堆肥に適していない生ごみを取り除いてもらう機能、コンポスターで管理している堆肥の質量と堆肥化が終了するタイミングを表示する機能、QRコード認証でコンポスターの使用回数を記録する機能です。



1 システム構成図

4. 提供する機能

4.1 堆肥化・堆肥の受け渡し

自作のコンポスターで生ごみの堆肥化・堆肥の受け渡しを行います。

コンポスターには、堆肥の攪拌のための攪拌装置とハンドルがあります。生ごみを集める容器と、堆肥化させる容器の 2 つの容器があり、距離センサで測定した値で生ごみ台の角度を切り替えます。利用者が一定回数コンポスターを利用すると堆肥を受け取ることができます。容器を引き出しのようにスライドさせることで簡単に堆肥を取り出すことができます。普段は容器がロックされ、堆肥が取り出せないようになっています。堆肥の受け取り時期は温度センサで取得した値から、堆肥化が終了するタイミングを求めます。

モーター、距離・温度センサは Raspberry Pi で制御します。

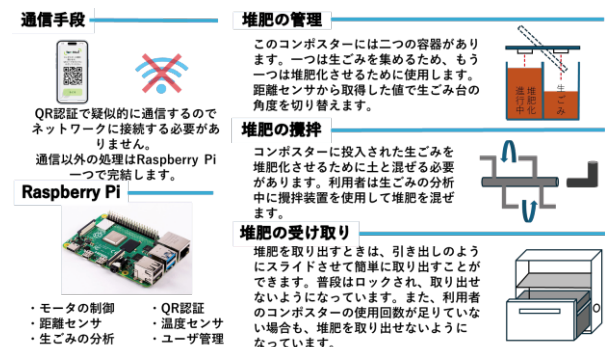


図 2 各機能の設計

4.2 生ごみの分別

生ごみ分析 API で生ごみの堆肥化適性を調べます。生ごみをカメラで撮影し、生ごみが堆肥に適しているか識別します。「堆肥に適している生ごみ」は緑色、「細かく切る必要がある生ごみ」は黄色、「堆肥に適していない生ごみ」は赤色に色分けされます。「堆肥に適している生ごみ」以外は利用者に取り除いてもらいます。

19 アル CO

地球を守る、優しいアプリ

奈良

崎本 俊治（4年） 日下 優人（4年）
池本 拓冬（4年） 森岡 大和（3年）
増井 崇（3年） 上野 秀剛（教員）

1. はじめに

2022年度では日本の自家用乗用車のCO2排出量は国全体のCO2排出量の8.3%を占めており、この半分は1Km以内の短距離にも自動車を使っている現状があります。このことから、車によるCO2排出の問題に関心を持ってもらうことが課題であると考え、「移動手段」という観点でCO2排出問題に対する関心を広めるために、私たちは「アルCO」を開発しました

2. 作品の概要

アルCOは、プレイヤーが歩いて貯めた歩数をゲーム内貨に変換され、それを用いて仮想の地球上で環境問題を解決していくゲームです。

ゲームを通じてプレイヤーに車から徒歩への移動への切り替えを促し、単距離の自動車での移動による排気ガスの抑制に貢献することができます。また、自分の成果をSNSに投稿することでモチベーションの向上を図ります。

3. システムの概要

3.1 歩数計APIを利用したデータ処理

本ゲームは、AndroidのAPIを使用してユーザーの歩数を計測し、取得された歩数データをユーザーの操作によってデータベースに送信します。このデータはゲーム内通貨に変換され、アイテムの購入に使用できます。

3.2 ゲーム進行の概要

プレイヤーは、並行世界の地球を救うためにアイテムを使用して、週単位で救済処置を行います。歩数データが送信されない（＝プレイヤーが長期間歩かない）場合、ゲーム内のパラメーターが悪化し、自然異常現象によるデバフ（海面の上昇、異常気象等）が発生することがあります。このデバフは一部アイテムの使用でも発生する可能性があり、悪化の速度が加速します。1週間後、パラメーターに基づいたエンディングが発生し、「救済成功」と「救済失敗」の2つの大きな結末

が存在します。デバフの状態により、エンディングの内容が変化します。



3.3 Xへの投稿機能

本システムには、ゲーム進行の結果をX（旧Twitter）に投稿する機能があります。ユーザーが投稿ボタンを押すと、アカウント認可が行われ、取得した認可データを使用してTwitter APIと連携し、投稿が完了します。

4. 最後に

アルCOでは利用者がゲームを通じてCO2の削減について関心を持ち、またそれがSNSを通じて様々な方面に拡散されていけるような設計を行いました。

このアプリを通じて一人でも多くの人がこの輪に入ってもらえることを目指します。



20 金継ぎ VR

熊本
(八代)

武藤 淳之助 (4 年) 山下 遼太 (5 年)
 盛高 雄寛 (2 年) 脇田 龍之介 (4 年)
 吉里 彩心 (1 年) 小島 俊輔 (教員)

1. はじめに

金継ぎとは、陶磁器の割れや欠けを漆で修復し、金粉で装飾する日本の伝統工芸です。(図 1)

近年、持続可能な社会の実現のため、資源の再利用や修理に対する関心が世界的に高まっています。私たちは、破損したものに新しい魅力を付加してモノを大切にする文化である金継ぎを世界に広め、環境保全に貢献するために「金継ぎ VR」を提案します。

2. システム概要

現実の金継ぎは、陶磁器が割れたときでないとやる機会がありませんし、漆を扱う難しさや完成までの期間の長さが、新しく始めようとする方の壁になってしまいます。「金継ぎ VR」では、VR ヘッドセットさえあれば手軽に何度でも金継ぎ体験ができます。

2.1 VR金継ぎ体験

本ゲームでは、実際に筆や皿を握っているような感覚を再現するため、ハンドトラッキングを採用しています。ハンドトラッキングとは、手の形を認識し、ゲーム世界に手を出現させる技術です。このゲームの場合、図 2 のように手の指の先端や関節の位置・角度データを取得し、それをもとに筆を握る手の形を検知したり、筆の位置・角度を決定しています。これにより、現実と同じように指先による繊細なコントロールが可能です。破片をくっつける場面では、皿を握る形を作ると、皿(の破片)が現れます。皿は、ボロノイ図を用いたアルゴリズムによって毎回ランダムに割れるので、金継ぎしたときにユニークな模様が出来上がります。



図 1：開発メンバーによる金継ぎ作品

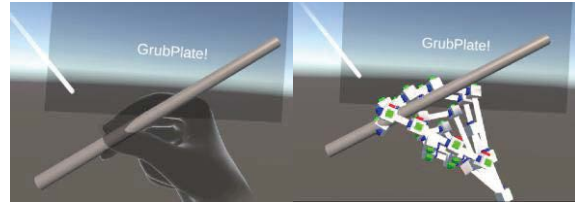


図 2：手の関節を認識し、筆の位置を決定する様子

また、破片に漆を塗るときは、間違った部分に塗ってもしっかりと反映されるので、緊張感が味わえます。

2.2 ランキング

プレイ毎に、漆を塗る際の正確性などからスコアを算出し、ユーザーはそれをランキングに登録できます。現状はスコアの登録のみですが、将来的にはスコアだけでなく他人の作品データの保存、鑑賞も行えるようにする予定です。

3. システム構成

図 3 に示したように、バックエンドは Docker 上に Flask と PostgreSQL で構築しています。ユーザー登録とランキング登録の API を実装しており、Unity 上のフォームからアクセスできます。

4. まとめ

「金継ぎ VR」は、一切の費用やごみを出さず、やろうと思ったその日から金継ぎについて手を動かしながら学び、体験できるゲームです。ユーザーは、金継ぎ体験を通してモノを大切に使う精神を学べます。

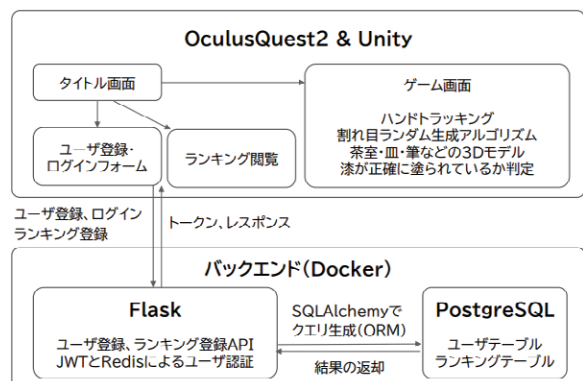


図 3：REST API を用いたシステムの構成図

1. はじめに

地元自治体ではCO₂排出量抑制と土壌 pH 改善を目指して、廃材処理や下水処理で発生した廃材や汚泥をバイオ炭や下水汚泥炭に加工し、活用しています。バイオ炭や下水汚泥炭は土壌に撒くことで土中に炭素を保持し、CO₂抑制や土壌改善の効果があると言われています。農水省では作物による単位面積当たりの施用量目安を示していますが、pH 値の変化を確認しながらバイオ炭や下水汚泥炭を使うことを推奨しており、それが利用促進のネックとなっているようです。そこで、これらの問題を解決するために、私達はバイオ炭治郎を開発しました。

2. バイオ炭治郎の概要

バイオ炭治郎は土壌 pH 管理を補助するアプリケーションです。測定したい土壌にセンサーを設置し、土壌の pH 値を計測します。土壌の pH 値の他にも、時刻・水分量・電気伝導度・温度なども計測します。計測されたデータは Wi-Fi を通してデータベースに送信されます(図 1)。

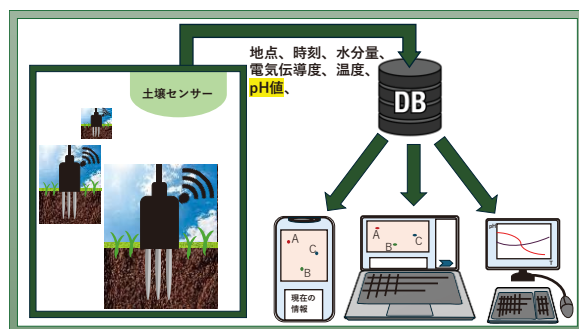


図 1. バイオ炭治郎のシステム構成図

ユーザは自身のスマホまたはパソコンを使用してバイオ炭治郎を利用します。(図 2)まず、地図から pH の値を確認したい場所のマーカを選択することで、その場所の pH 値を確認することができます。また、各マーカに目安の pH 値を設定することができ、設定した pH 値より大きく変動すると通知でユーザに知らせ

ます。特にバイオ炭や下水汚泥炭を追加する必要のあるときは、追肥を促す通知を出します。

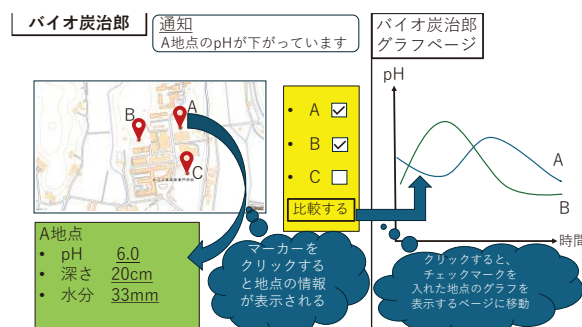


図 2. バイオ炭治郎の実行画面例

3. バイオ炭治郎の機能

機能①：土壌の pH を計測し、データを蓄積

この機能は、マイコンとセンサーで常時土壌 pH を計測し、結果の pH 値の他に、日時や位置情報、温度、水分量などをデータベースに送信します。

機能②：実測した pH 分布をマッピング

センサーの場所が地図上にマーカで表示されており、選択することでその場所の pH 値の時間変化をグラフで確認することができます。加えて、複数のセンサーの計測値を比較することが簡単にできます。

機能③：pH の変化をモニタリングし、対応が必要な場合にユーザに通知

各マーカに設定した pH 値より大きく変動した場合にスマホアプリに通知が届きます。この機能によって、pH を常時確認しなければいけない問題を解決することができます。

4. まとめ

バイオ炭治郎を利用することで、農家の人達は土壌の pH を調整しやすくなるだけでなく、土中の CO₂抑制することが可能となり、積極的にバイオ炭や下水汚泥炭を施用することで環境問題への解決にも繋がります。

22

育てるエコタウン

誰でも楽しく環境活動

熊本
(八代)

岩谷 奏慈郎 (2年) 廣野 達徳 (2年)
緒方 亮太 (5年) 高見 勇斗 (1年)
高山 透也 (5年) 小島 俊輔 (教員)

1. はじめに

現在、ゴミのポイ捨てによる海洋ゴミの増加・汚染などの環境破壊や地球温暖化という問題に直面し、環境に配慮した暮らしや持続可能な社会に興味を持つ人が増えている。しかし、具体的にどのような活動を行えばいいかわからない人や続かずやめてしまう人も多くいる。

そこで、「簡単にできる」、「継続できる」、「楽しくできる」をコンセプトとして環境活動をサポートし、持続可能な社会をめざす「育てるエコタウン」を開発した。

2. システム機能

2.1 エコタウン開発

「育てるエコタウン」は環境に良い活動をして街を発展させていく街づくりゲームである。最初は荒廃した街からスタートし、ユーザーはゲーム中の街を開発することで街の環境を改善し、環境に優しい「エコタウン」を作り上げていく。

後述のミッションを達成することでポイントが貯められ、そのポイントを使うことで自分だけのエコタウンを発展させることができる。

2.2 ミッション

複数の環境活動に貢献ができるミッションが毎日更新され、ユーザーはこれらのミッションを達成することでポイントを手に入れる。ミッションには食品ロス、ゴミ問題、節電など様々なジャンルがある。毎日継続してミッションに取り組むことで貰えるポイントが増えていき、継続して活動するモチベーションを高める。

2.3 スマートゴミ箱

ゴミが捨てられたことを検知するデバイスを用い、ゴミのポイ捨てを抑制し、環境悪化の防止に貢献できる。このデバイスは小型で簡単にゴミ箱に取り付け可能である。Bluetooth でスマートフォンと連携することでゴミをゴミ箱に捨てるとポイントを手に入れる。

2.4 Tips

ゲームのロード中に環境問題に関する豆知識を掲載することによって無駄なスキマ時間を有効に活用して環境問題についての知識を自然と身につけることができる。

2.5 UI

図1の操作画面のUIにあるように下のバーを操作して画面を切り替えることができる。主に3つの画面から構成されており、それぞれエコタウンの街を閲覧・編集（開発）する画面、ミッションの内容を確認し達成を報告する画面、そしてショップでエコタウンの発展に必要な建物などを購入する画面である。これらの画面を切り替えながらゲームを進めていく。

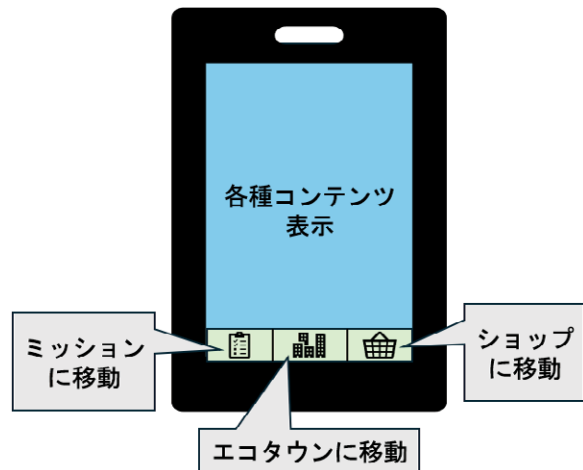


図1 基本的なUIの例

3. システム構成

本システムは図2のようにUnityを用いて画面を表示する。デバイスは主にスマートフォンを想定している。またサーバーにはFlaskとPostgreSQLを用いユーザーの情報を管理している。

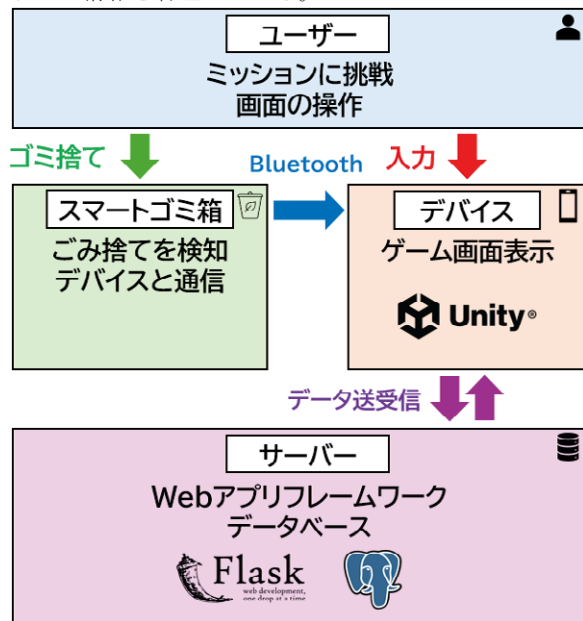


図2 システムの構成

4. まとめ

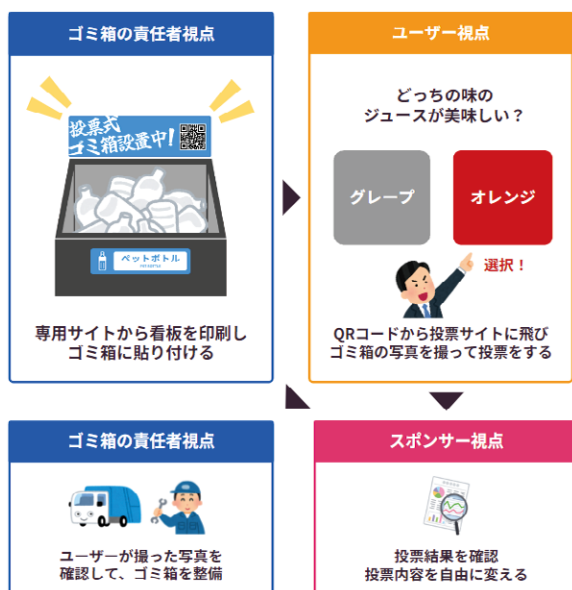
「育てるエコタウン」は環境活動にゲーム性を取り入れ、手軽に楽しく継続して環境問題の解決をサポートするアプリである。環境活動の敷居を下げ、取り組む人を増やし、持続可能な社会の形成を目指す。

1. はじめに

ポイ捨て減少に効果抜群であることを証明した「投票式灰皿」が注目を浴びています。投票式灰皿とは、「人生に大事ななのは愛or金?」といった誰しもが悩む究極の二択の設問を記載し、吸い殻をどちらかに入れたくなる楽しさを生み出した画期的なアイデアです。しかし、「設置・運用コストが高い」「吸い殻でしか投票できない」等の理由から全く普及していません。そんな投票式灰皿をアップグレードした、設置コストを抑えての持続的な運用が可能な「デジタル投票式ゴミ箱 POOI (ポーイ)」を提案します。このシステムにより、投票式ゴミ箱を全国に大量設置することが可能になります。

2. システム概要

本システムは、「ゴミ箱の責任者」「ユーザー」「スポンサー」の三者それぞれにメリットがある関係になることが魅力です。ゴミ箱の責任者が専用サイトからQRコード付き看板を制作し、既存のゴミ箱に取り付けます。そして、ゴミを捨てに来たユーザーはQRコードからwebアプリに飛び、ゴミ箱内を撮影し投票します。その撮影データから、責任者はゴミ箱の状態をオンラインで管理でき、スポンサーはデータ収集と広告効果を得ることが出来ます。



2.1 投票機能 (ユーザー視点)

看板のQRコードをスマホで読み取ることでwebアプリが開き、自分の好きな陣営に「投票玉」をスワイプして投票できます。また、ゴミ箱内を撮影することによってより多くのポイントを投票することができ、自分の好きな陣営のポイントをより増やすことができます。



2.2 ゴミ箱管理機能 (ゴミ箱の責任者視点)

ゴミ箱の管理はたくさんの手間を必要とします。そこで、ゴミ箱の管理を補助する機能を開発しました。それは、ゴミ箱の状態を簡単に確認することができる機能です。ユーザーから「異常なし」「満杯」「破損」の3つの状態が募られ、管理者はゴミ箱の状態を知ることができます。また、ユーザーが投票の際撮ったゴミ箱内の写真の履歴を見ることができるので、現地に行かなくてもゴミ箱内の状態を確認することができます。

2.3 データ管理機能 (スポンサー視点)

企業がデータ収集と広告を目的としてスポンサーになることができます。スポンサーは投票内容を詳細に設定することができ、全国のゴミ箱のデータを管理することができるので、マーケティングに活用することが可能になります。

3. おわりに

POOIは、ユーザーが楽しくゴミを捨てることでポイ捨てを減らし、ゴミ箱の管理も楽になる革新的なシステムです。

自由部門本選参加作品

■自由なテーマで独創的な作品

発表 番号	タイトル	学校名	指導教員	作成学生
1	！ -the world of billiards-	弓削商船	長尾 和彦	石橋 治樹 (3年)、五所 杏太 (3年)、小田 紘平 (2年)、 木村 斗哉 (2年)、濱田 捷聖 (1年)
2	Don't Asleep	小山	大内 翔平	大竹 響己 (4年)、宮原 友哉 (4年)、藤澤 颯介 (3年)、 池田 陽菜太 (2年)、河口 夏音 (2年)
3	Secure Ryman -情報セキュリティ対策推進サーブス-	木更津	米村 恵一	八木 航樹 (専攻科1年)、富澤 太一 (専攻科1年)、 鈴木 聡一郎 (専攻科1年)
4	つくも遊歩 -つくも神と巡るこころの故郷-	東京	山下 晃弘	柴野 真惇 (2年)、石井 岳光 (2年)、田中 璃久 (2年)、 篠倉 詠音 (2年)、西村 朱葉 (2年)
5	uni	香川 (詫間)	金澤 啓三	山田 美羽 (5年)、天竺 寛貴 (5年)、横井 優樹 (5年)、 小原 崇靖 (5年)、菅 勇磨 (5年)
6	AIデンティファイ -安全で楽しいSNSライフをあなたに-	八戸	釜谷 博行	白石 光 (3年)、館向 大輔 (3年)、三浦 士 (1年)、 熊野 りお (3年)、杉本 慎之介 (3年)
7	PeerReach -仲間との協力でトップへ到達-	沖縄	金城 篤史	椎名 咲太郎 (5年)、松田 恋椰 (5年)、前川 優太 (5年)
8	Tourwith -パンダには旅をさせよ-	佐世保	手島 裕詞	吉田 健志 (4年)、前田 一步 (4年)、美甘 悠吾 (4年)、 山川 瑛児 (4年)、松本 大心 (4年)
9	ふたりんごと -どうやってこの感じ伝えようかな-	豊田	都築 啓太	野崎 春太郎 (5年)、志比田 悠翔 (3年)、深谷 春駈 (2年)、 小松 杏菜 (2年)、加藤 遥也 (2年)
10	覚え歌つくくん	仙台 (広瀬)	穂坂 紀子	八重樫 守 (3年)、荒 俣 誠 (2年)、佐藤 佑剛 (2年)、 中島 悠登 (2年)、八木沼 佑斗 (2年)
11	こねくと (仮) -災害用スマホネットワーク-	群馬	渡邊 俊哉	松本 祐真 (4年)、黛 琵琶 (4年)、藤田 恭輔 (4年)
12	SPORTSDAY -球技大会管理プラットフォーム-	富山 (本郷)	石田 文彦	山本 哲也 (5年)、中村 祐輔 (5年)、池岸 太勇 (2年)
13	なまっ talk -テキストでだって方言を-	富山 (射水)	山口 晃史	高山 七帆 (3年)、田中 千尋 (4年)、Ariq Athallah Budi Ramdhany (3年)、 源明 玲煌 (1年)、Naksub Sorachach (2年)
14	エモアンブ	福井	高久 有一	片岡 志菜 (4年)、岡本 篤希 (4年)、高橋 楓哉 (4年)、 大瀬 怜幸音 (4年)、渡邊 展匠 (4年)
15	シオサビン -塩害を検知、予報し 被害を未然に防ぐ塩害対策アプリ-	沖縄	金城 篤史	川満 秀太郎 (3年)、屋宜 元晃紗 (3年)、吉田 信仁 (4年)、 喜友名 朝舜 (1年)、宮里 壮汰 (4年)
16	フィッシュフレンズ -魚をより近い存在に-	香川 (高松)	村上 幸一	三宅 立晃 (3年)、門田 統海 (3年)、 金地 琳太郎 (専攻科2年)
17	Regina -自動学習精算機-	広島商船	岩切 裕哉	谷本 佑樹 (5年)、原 裕次郎 (5年)、祐本 菜々美 (5年)、 室岡 陸 (3年)、祐本 浩輝 (3年)
18	どみとる -学生寮共用設備利用状態管理システム-	石川	越野 亮	福井 悠生 (4年)、村上 垂士沙 (4年)、置田 裕也 (4年)、 毛利 俊太 (4年)、毛利 謙太 (4年)
19	Nahlun -ボクセルで見える！河川水位と地域の安全-	群馬	木村 清和	荒井 日菜子 (5年)、菊池 静琉 (5年)、後藤 順太 (5年)
20	この虫な〜んだ？ -ハンドサインで虫がわかる！！-	徳山	力 規晃	田中 響大 (3年)、池永 珠希 (3年)、平塚 智紀 (2年)、 伊藤 結彩 (2年)
21	30 Days Harvest	キングモンクット 工科大	Sirion Vittayakorn	Pakin Kaophuthai, Peravit Kritchakaj
22	AfterDay Horizon	キングモンクット 工科大	Supannada Chotipant	Parin Dechakorn, Pairat Chuenchom
23	Neko Torae	シンガポール ポリテクニク	WONG Kwee Yin	SENG KAY KANG JERICK, LEROY HONG JAE-YANG
24	QuizRush	新モンゴル 高専	Shur-Erdene Buyannemekh	Bilguuntushig Amarsaikhan, Tuguldur Batsaikhan, Choigun Kherlen, Bayarsaikhan Baatarkhuu, Buyandelger Myagmarjav
25	Sign-lent	タイ高専	Pirapat Tangsuknirundorn	Pradubfah Thanataweerat, Worawalan Woradee

自由部門 プレゼンテーション審査 タイムテーブル

審査日時 10月19日(土) 10:10 - 17:14

会場 小ホール

発表持ち時間 発表時間 8分 質疑応答 4分(海外チーム 6分) 交代 1分

発表順	発表予定時間	タイトル	学校名
1	10:10 ~ 10:22	! - the world of billiards -	弓削商船
2	10:23 ~ 10:35	Don't Asleep	小山
3	10:36 ~ 10:48	Secure Ryman - 情報セキュリティ対策推進サービス -	木更津
4	10:49 ~ 11:01	つくも遊歩 - つくも神と巡るこころの故郷 -	東京
11:01 ~ 11:09 休憩 8分			
発表順	発表予定時間	タイトル	学校名
5	11:09 ~ 11:21	uni	香川(詫間)
6	11:22 ~ 11:34	AI デンティファイ - 安全で楽しい SNS ライフをあなたに -	八戸
7	11:35 ~ 11:47	PeerReach - 仲間との協力でトップへ到達 -	沖縄
8	11:48 ~ 12:00	Tourwith - パンダには旅をさせよ -	佐世保
12:00 ~ 13:00 休憩 60分			
発表順	発表予定時間	タイトル	学校名
9	13:00 ~ 13:12	ふたりんごと - どうやってこの感じ伝えようかな -	豊田
10	13:13 ~ 13:25	覚え歌つくるくん	仙台(広瀬)
11	13:26 ~ 13:38	こねくと(仮) - 災害用スマホネットワーク -	群馬
12	13:39 ~ 13:51	SPORTSDAY - 球技大会管理プラットフォーム -	富山(本郷)
13:51 ~ 14:00 休憩 9分			
発表順	発表予定時間	タイトル	学校名
13	14:00 ~ 14:12	なまっ talk - テキストでだって方言を -	富山(射水)
14	14:13 ~ 14:25	エモアンプ	福井
15	14:26 ~ 14:38	シオサビン - 塩害を検知、予報し 被害を未然に防ぐ塩害対策アプリ -	沖縄
16	14:39 ~ 14:51	フィッシュフレンズ - 魚をより近い存在に -	香川(高松)
14:51 ~ 15:00 休憩 9分			
発表順	発表予定時間	タイトル	学校名
17	15:00 ~ 15:12	Regina - 自動学習精算機 -	広島商船
18	15:13 ~ 15:25	どみとる - 学生寮共用設備利用状態管理システム -	石川
19	15:26 ~ 15:38	Nahlun - ボクセルで見える! 河川水位と地域の安全 -	群馬
20	15:39 ~ 15:51	この虫な〜んだ? - ハンドサインで虫がわかる!! -	徳山
15:51 ~ 16:00 休憩 9分			
発表順	発表予定時間	タイトル	学校名
21	16:00 ~ 16:14	30 Days Harvest	キングモンクト工科大学
22	16:15 ~ 16:29	AfterDay Horizon	キングモンクト工科大学
23	16:30 ~ 16:44	Neko Torae	シガポールポリテクニク
24	16:45 ~ 16:59	QuizRush	新モンゴル高専
25	17:00 ~ 17:14	Sign-lent	タイ高専

プレゼンテーション審査終了

自由部門デモンストレーション審査&マニュアル審査 タイムテーブル

審査日時 10月20日(日) 9:00～12:15

会 場 中ホールホワイエ

審査時間 7分(説明2分, 質疑応答5分, 移動を含む)

審査時間	自由部門			
	E 班	F 班	G 班	H 班
9:00～9:07	1	7	13	19
9:07～9:14	2	8	14	20
9:14～9:21	3	9	15	21
9:21～9:28	4	10	16	22
9:28～9:35	5	11	17	23
9:35～9:42	6	12	18	24
9:42～9:49	7	13	19	25
9:49～9:56	8	14	20	1
9:56～10:03	9	15	21	2
10:03～10:13	休憩 10分			
10:13～10:20	10	16	22	3
10:20～10:27	11	17	23	4
10:27～10:34	12	18	24	5
10:34～10:41	13	19	25	6
10:41～10:48	14	20	1	7
10:48～10:55	15	21	2	8
10:55～11:02	16	22	3	9
11:02～11:09	17	23	4	10
11:09～11:19	休憩 10分			
11:19～11:26	18	24	5	11
11:26～11:33	19	25	6	12
11:33～11:40	20	1	7	13
11:40～11:47	21	2	8	14
11:47～11:54	22	3	9	15
11:54～12:01	23	4	10	16
12:01～12:08	24	5	11	17
12:08～12:15	25	6	12	18

注意事項

- ① E 班, F 班, G 班はデモンストレーション審査
H 班はマニュアル審査を示す
- ② 1～25 はプレゼンテーション審査の発表順の作品を表す
- ③ 色が付いているセルは, 海外チームを表す

1. はじめに

ビリヤードは相手の打つ手を予想しながらプレイする戦略的な奥深さと幅広い年齢層で楽しむことができる多様性を持った競技です。しかし、近年ではビリヤード場の老朽化やプレイ可能な施設が周辺にないため気軽に始めるのが困難などの原因から若者を中心に競技人口は年々減少しています。

そこで、私たちは「ビリヤードを始めたい人」や「ビリヤード場が周辺にない人」、「ビリヤードを観戦したい人」などを対象とした自宅で気軽に実物を用いてゲームを楽しめるシステム「!」を提案します。

2. 概要

2.1 ボール検知

連続的に Azure Kinect から送信されるビリヤード台の画像データからボールの探索範囲を指定し、グレースケール化後に Hough Circles 関数を用いてボールの輪郭を検出します。検出されたボールのカラーヒストグラムと事前に作成した各ボールの平均カラーヒストグラムとのマッチ度を比較し、ボールの番号を判別します。なお、同じ番号のボールを認識した場合、2つのマッチ度の比較結果が高い方を優先することで番号の重複を防いでいます。射影変換でボールの座標を正規化し、ボールの停止が確認された場合に相手ヘターンが切り替わり、座標と番号のデータをサーバに送信します。[1]

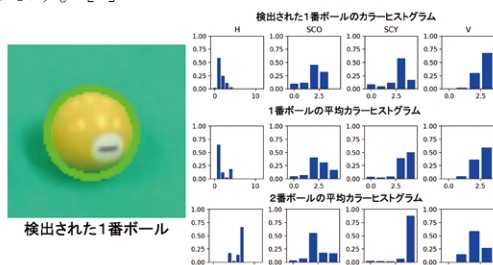


図1 ヒストグラムとのマッチ度

2.2 ボール位置表示

サーバから座標と番号のデータを画像作成部に送信し、ボールの 3D モデルを座標と番号データに従って配置します。出力された画像データを相手のビリヤード台に投影し、プレイヤーはボールの 3D モデルが停止した位置に実際のボールを設置します。

2.3 システム構成

図2に「!」のシステム構成を示します。プレイヤーが実際のボールを指示された位置に設置し、ボールを打つと Azure Kinect で連続的にビリヤード台のボールの画像データを取得します。画像データを画像認識部に記録し、ボールが停止すると座標と番号のデータが出力されます。試合データとしてサーバに保存された座標と番号のデータは相手プレイヤーの画像作成部に送信され、ビリヤード台にボールの 3D モデルを

配置した画像データを相手のプロジェクトからビリヤード台に投影します。

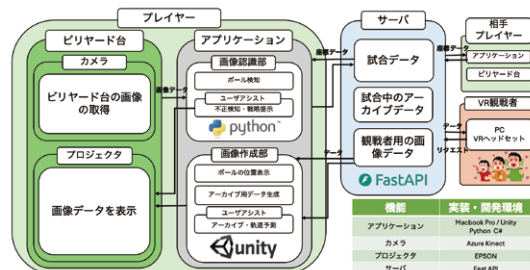


図2 システム構成図

3. 機能

3.1 ユーザアシスト機能

① 軌道予測

Azure Kinect で取得したキュー(棒)の角度と位置の画像データとボールの座標と番号のデータから、打った場合に予測されるボールの軌道を表示することでプレイヤーは事前に打つ方向を確認できます。

② 戦略提示

ルールや戦略を学習した PC が、取得したボールの座標と番号のデータからプレイヤーの勝ち筋を予測し、適切な目標を提示します。

③ 不正検知

指定された位置にプレイヤーがボールを置かなかった場合やボールを指定した位置からずらすなどの不正を検知し、プレイヤーへ指示します。

3.2 観戦機能

ユーザは VR ゴーグルやスマートフォン、PC を用いて Unity 上のビリヤード台で任意の視点からボールの座標変化を確認できます。観戦方法には観戦者向けのリアルタイム視聴とプレイヤー向けのアーカイブ視聴の2つがあります。

4. 今後の課題

「!」は初心者や経験者を問わずビリヤードを気軽に楽しめる環境を提供することを目的としています。開発段階で「ビリヤード台の横に配置したプロジェクトの位置からボールを打ちづらい」などの課題が見つかりました。この問題に関してはプロジェクトを天井に固定することで対応可能と考えています。また、本システムはナインボールのみの実装となっていますが、今後は他ゲームへの対応も検討しています。

5. 参考文献

- [1] 相澤朋希, 三輪貴信, 澤田秀之, 橋本周司. ビリヤードの配置図作成のためのボールの検出および番号識別. 情報処理学会インタラクシオン 2018. <https://www.interaction-ipsj.org/proceedings/2018/data/pdf/1B30.pdf> (参照: 2024-9-30).

1. はじめに

現在、居眠り運転により引き起こされる事故が問題となっています。また、既存の居眠り運転防止デバイスでは、映像による眠気の検知のためバイクなどヘルメットが必要な場合は使用できない、周囲の車が居眠り運転している車を検知できないなど様々な問題が残されています。そこで私たちは、以上の問題を解決するため、「Don' t Asleep」を開発しました。

2. 概要

「Don' t Asleep」は手首に装着するウェアラブルデバイスと専用アプリケーションで構成させる居眠り運転警告システムです。

使用者の脈拍の変化により眠気を感知した際、手首の振動モーターとスピーカーが動作することで使用者の居眠り運転を防止します。また、眠気レベルが高い場合はマップ上に使用者の位置情報と眠気レベルを表示し、音声で知らせることで、周囲の人に居眠り運転の車を通知し危険を防止します。

図1にデバイス利用フローを示します。

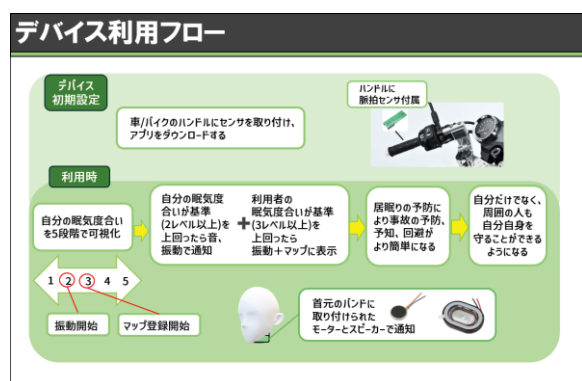


図1 デバイス利用フロー

3.1 眠気検知機能

脈拍センサにより脈拍を計測し、その変動を検知することで、眠気レベルの算出を行います。

3.2 振動・音声通知機能

強い眠気を感知すると、振動モーターとスピーカーにより使用者に警告を行います。

3.3 位置通知機能

算出した眠気レベルが一定を超えた場合、使用しているスマホのGPSの位置と使用者の眠気レベルをマップに表示し、居眠り運転の車の存在を音声と共に通知します。

また、それによる位置情報をデータベースに蓄積し、居眠り運転が起きやすいを統計から導き出し表示します。

4. システム構成

装着デバイスでは、反射型センサでの心拍の測定値をもとに、Raspberry Piで眠気レベルを求め、Bluetooth通信によりアプリケーションへ眠気レベルを送信します。また、強い眠気を感知した場合には、装着デバイスで振動、警告音により運転者に通知します。アプリケーションは位置情報と眠気レベルのデータをサーバーと通信し、周囲のユーザーと車の場所と眠気レベルを共有します。

図2にシステム構成を示します。

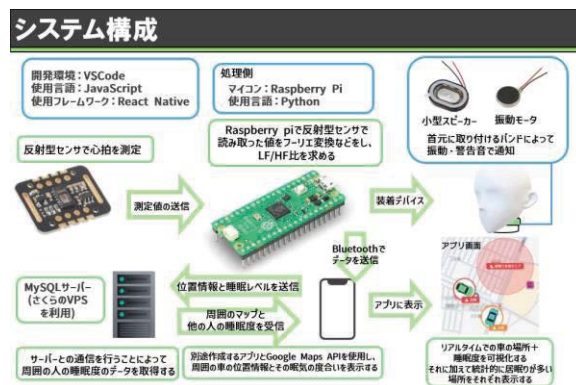


図2 システム構成

5. まとめ

「Don' t Asleep」は振動と音声により居眠りの防止を行い、位置情報により周囲に危険を通知する居眠り運転警告システムです。このデバイスで多くのドライバーの方々の力になれば幸いです。

1. はじめに

近年、中小企業がサイバー攻撃の標的となる「サプライチェーン攻撃」が増加しています。中小企業のセキュリティ対策は、自社だけでなく取引先の大企業のセキュリティリスクも低減させる重要な課題です。しかし、多くの中小企業は学習の難しさ、コスト面の制約、適切な対策方法の不明確さなどの課題を抱えています。

2. 概要

Secure Ryman は実際に診断を行い、現場の声を反映させて作った情報セキュリティ対策推進サービスです。中小企業全体のセキュリティレベル向上を目指し、現場の方々がセキュリティの知識を手軽に身につけられるようサポートします。同時に、経営者のセキュリティ対策への意識向上も促進します。

3. システムの特徴

- ・セキュリティ診断ツール



図1 セキュリティ診断

個人のセキュリティ能力をさまざまな項目を問うアンケートによって可視化します。

- ・カスタマイズされた教育コンテンツ

診断結果に基づいて、優先的に学ぶべき教育コンテンツを提供し、受講後にテストを実施します。



図2 教育コンテンツ

- ・コミュニティプラットフォーム



図3 コミュニティプラットフォーム

他社のセキュリティ能力や対策予算、セキュリティ商品の導入事例を共有し、自社の対策の参考にできます。

4. システム構成

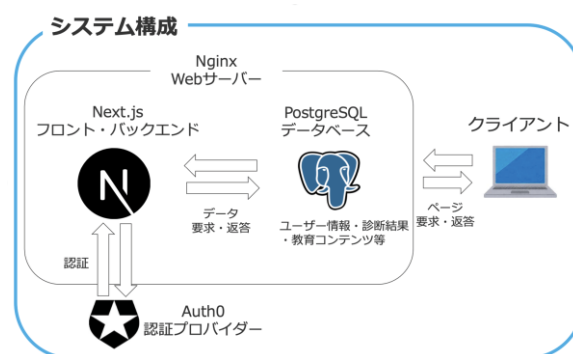


図4 システム構成図

5. 最後に

Secure Ryman は、現場の声を反映させた中小企業のためのセキュリティ対策推進アプリです。日本の中小企業のセキュリティ問題解決に貢献するため、私たちは開発を進めてまいります。

1. はじめに

日本に古くから根付く伝統文化の一つ、「神道」。

その信仰の土台として日本国内に 10 万社以上あるといわれる「神社」は、地方・都市部問わず、豊かな自然やご利益、御朱印、お守りなどの高いコレクション性により多くの参拝者に愛されてきました。

しかし、そのような神社がいま、多くの課題を抱えていることはご存知でしょうか。その課題の最たる例として、「関心の低下」や「参拝者数の減少」が挙げられます。

そこで私たちは、神社の抱える課題をユーザーの視点から、ソーシャルゲームを用いて解決するアプリケーションである「つくも遊歩」を開発しました。

2. 作品概要

「つくも遊歩」は、神社について知り、神社参拝を楽しくする機能を搭載した、神社巡りをサポートする多機能ソーシャルゲームアプリです。

「つくも遊歩」には、主にマップ機能、チェックイン機能、キャラ・ストーリー機能の 3 つのシステムを導入しています。

2.1 マップ機能

標準では、現在地と周囲の神社を表示します。

また、名称や特徴などの情報から検索機能を用いることで、神社を調べることができます。

マップに表示された神社のアイコンをタップすると、神社名や住所、祭神、御朱印の有無などの情報を閲覧できます。

開発段階では全国の官幣、国幣の神社を中心に約 3 万社のデータベースを独自に作成し、マップ機能に導入しています。



図1 マップ機能(開発中)

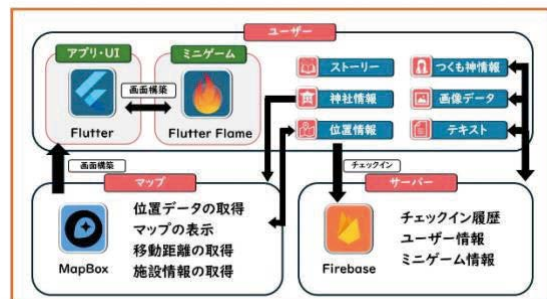


図2 システム構成図

2.2 チェックイン機能

参拝した神社を記録し、過去に訪れた神社やそれぞれの参拝回数を確認することができる機能です。神社に近づくとアプリからチェックインを行うことができ、チェックインした神社の詳細な情報を確認することができます。

チェックイン画面に「絵馬掛け所」を実装しました。神社に訪れた際に願い事を書くことができ、同じ神社で他の参拝者が書いた絵馬を閲覧することもできます。

2.3 キャラ・ストーリー

多くの神社で祭神として祀られている神様を「つくも」というキャラクターとして実装しています。ユーザーは好きな「つくも」をパートナーにして神社を巡ることで、「つくも」との親密度を高め、キャラクター毎のストーリーを読むことができます。

また、神社や神道の歴史を学ぶことができるストーリーも実装しています。

3. まとめ

私たちは、日本に昔から受け継がれてきた神社や神道といった文化には、今の時代でも人々の心の拠り所として高い価値があると考えています。

「つくも遊歩」は、身近にある神社への参拝を、より多くの人が、より快適に楽しめるようになるツールを目指して開発を行いました。

子どもから大人まで様々な人が日常の中で神社に触れることで、神社について関心を持ち、知識や文化を継承していく。このアプリを通じて、そうした未来の形を作ることができると信じています。

1. はじめに

国際化が進む現代、異なる言語を話す人々との交流がますます増えています。しかし、言語が異なると、相手の体験や感情を理解し、共感することが難しく覚えることもあるのではないのでしょうか。

このような課題を解決するために、私たちは言葉に頼らず、風の触覚を用いて誰もが一緒に楽しむことができる universal なシステム「uni」を提案します。ヘレン・ケラーが水を触覚で理解したように、「uni」では触れることで楽しめる新たなシステムを目指します。

2. システム概要

「uni」では、ユニバーサルデザインを採用し、映像・風の感触・音で、言葉による説明を必要としないユーザー同士の共感につながるコンテンツを提供します。

また、風を感じるという体験を通じて、新たな発見や驚きを生み出します。

3. システム構成

本システムの構成を図1に示します。「uni」では、プロジェクトで布に映像を投影し、複数のファンを用いて布を動かし、風を表現します。また布の背面に配置した複数の距離センサで布との距離を計測し、布を入力デバイスとしても活用します。他にも、3D サウンドの出力や Azure Kinect でユーザーの認識を行います。

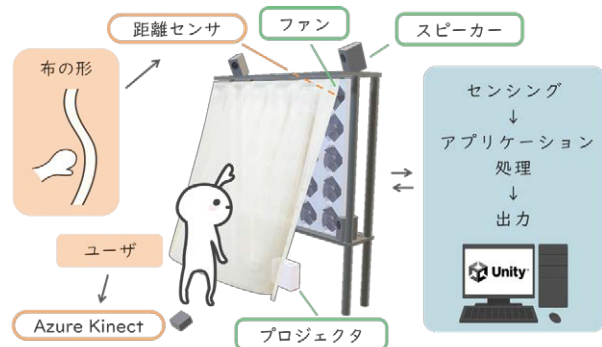


図1 システム構成

4. ユーザのタッチの検出

「uni」では、ユーザーの布へのタッチを検出するために、ユーザーの認識や布の形状のセンシングを行います。図2に各センシングの様子を示します。Azure Kinectから得たユーザーの手の位置の情報と、二次元格子状に配置した複数の距離センサから得た布の形状の変化量を組み合わせることでタッチを検出します。

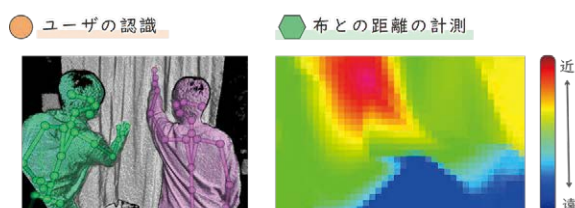


図2 センシングの様子

5. 「uni」の体験

体験の流れの例を図3に示します。「uni」のシステムの前に立つと各ユーザーの前に蝶が現れ、ユーザーがタッチするとその場所に花が咲きます。この動作を繰り返すうちに蝶は飛んでいき、遊びながら一期一会の花畑を創りあげていくことができます。

その他にもさまざまなコンテンツがあり、これらの体験を通じて、新たな驚きの発見や試行錯誤などの共通体験を生み出し、共感や相互理解を促進します。

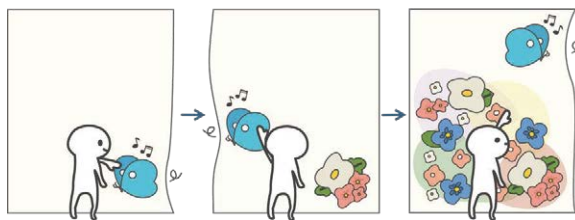


図3 「uni」の体験の流れ

6. おわりに

「uni」では、同じ体験を共有しながら、誰とでも一緒に風で遊ぶことができます。皆さんも「uni」で、これまでにない「風」での交流を体験してみませんか？

1. はじめに

近年、若者を中心に SNS が普及しています。スマホやパソコンがあれば誰でも簡単に全世界に情報を発信できるのは良いことですが、その反面、危険も多く潜んでいます。その中の 1 つが個人情報の漏洩です。たった 1 枚の写真から住所が知られ犯罪に巻き込まれてしまうケースもあります。

そこで私たちのプライバシーを守るために、自然な形で写真を補正し、個人情報の流出を防ぐアプリ「AI デンティファイ」を提案します。

2. システム概要

「AI デンティファイ」はユーザーが撮影した写真から個人情報の特定に繋がる情報を自動的に検出し、画像生成 AI がその部分を違和感なく安全で自然な形に修正してくれます。また、目的に応じて、写真の加工方法も変更することができます。

3. AI デンティファイの機能

「AI デンティファイ」は画像をアップロードし、オブジェクトやテキストを inpaint (画像へ自然な形で補正) 処理するか、モザイク処理するかを選択することができます。図 1 に操作画面を示します。図 2 に処理例を示します。



図 1: 操作画面

ユーザーが指定する消去レベルに応じて、文字や看板などを自動的に消して、図 2 のように撮影場所や個人情報特定されないようにしてくれます。



図 2: 処理例

4. システム構成

ユーザーからアップデートされた画像のデータの処理の流れを図 3 に示します。

物体検出

GroundDINO と YOLO を使用して写真内のオブジェクトを検出します。YOLO は、GroundDINO の入出力関係によって自動でトレーニングされ、GroundDINO 単体で用いるよりも高速、かつ柔軟な物体検出が可能です。

消去対象の決定

消去レベルと画像のシュチュエーションに応じて適切な消去オブジェクトの名前を決定します。

危険度の判定

画像情報をベクトル化し、学習画像のベクトルとのコサイン類似度によって画像の危険度を算出します。

消去プログラムの選択・実行

この関数が画像内の物体を最適な手法で消去します。工夫した点として、消去対象物の面積と種類によって使用するプログラムを変えており、例えば、数字などの場合は OpenCV の inpaint 機能、人などの場合は Stablediffusion を用いるなど、画像処理の質と高速化の両立を目指しました。

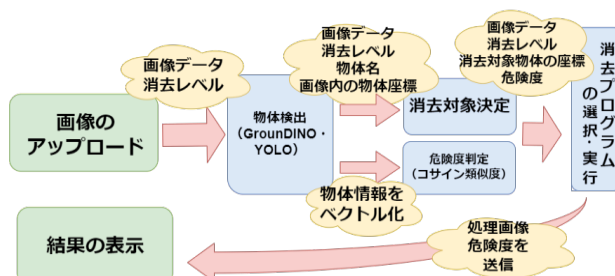


図 3: データ処理の流れ

5. おわりに

日常生活に欠かせないものとなっている SNS は、これからの時代もっと浸透していきます。SNS は便利なものですが使い方を間違えると危険を招くこともあります。ぜひ今日から「AI デンティファイ」を用いた安全で楽しい SNS ライフを始めてみませんか？

1. はじめに

誘惑が多い現代、

- ・みんながどのくらい勉強しているかわからない
- ・ルーティーンを組もうとしても挫折してしまう
- ・ついついスマホに手が伸びてしまう

といった様々な悩みがある。

そこで、共有、競い合いによってタスクを遂行する学生向けアプリケーション「PeerReach」を提案する。

2. 概要

仲間同士でタスクの内容を共有することで、ピアプレッシャーを生み出し、タスクに対する意欲を高める協力型タスク管理アプリである。

「あいつがやるなら俺もやろう」

「自分の達成度も見られている」

という感覚からタスクを遂行する原動力を生み出す。



2.1 ピアプレッシャーとは

ピアプレッシャーとは、同じ立場の仲間からの監視などによって生じる心理的圧迫感を指す。

PeerReach では、ピアプレッシャーを通し、仲間同士がタスクを遂行することにより、仲間同士の競争心を高め、タスク遂行数アップにつなげる。

2.2 Todo 機能

5分・10分・20分に色分けされたボタンのうち、任意のボタンを押し、タスク名とセット数を入力することでToDoリストに追加できる。達成したタスクはコミュニティ内のメンバーに共有される。

2.3 コミュニティー機能

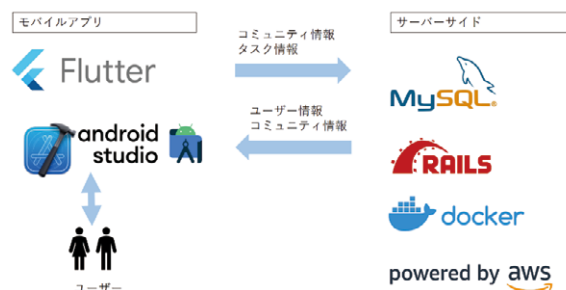
互いに競い合う仲間のグループを作る。誰とどのようなコミュニティを作るか、どのようなコミュニティに参加するか自由、自分の目的に合ったコミュニティを探ることができる。

2.4 ランキング機能

ランキングはリアルタイムに更新され、メンバーのタスク内容と時間を表示することができる。自分やライバルの情報が詳細に公表・共有されるため、競争心をあおり、ピアプレッシャーを生み出す。このアプリのメインコンセプトとなる機能である。

3. システム概要

マルチプラットフォームに対応するためにモバイルアプリはFlutterで、APIを提供するバックエンドはRuby on Railsで開発する。



4. 終わりに

ライバルの勉強時間が気になる。みんながどのくらい頑張っているのか知りたい。そんな負けず嫌いなあなたに是非！

1. はじめに

「旅行に行きたいけど、お金も時間も余裕がない...」
と思ったことはありませんか？変わらない日常の連続
で、退屈な日々だと感じる人もいるでしょう。そこで
私たちは、全国の人々と協力することで、日常の何気
ない時間でも、スマホで気軽に旅気分を味わえるアプ
リ「Tourwith」を提案します。

2. Tourwith とは

「Tourwith」は、オリジナルのパンダアバターを全
国のユーザーがリレーのように繋ぐことで、目的地を
目指しながら日本中を旅することができるスマホアプ
リです。道中で撮影されるあなただけの旅写真や全国
のユーザーとのふれあいにより、唯一無二の旅を体験
できます。

3. 機能紹介

3.1 みまもり機能

自分のパンダアバターの現在地を GPS により地図上
でいつでも確認できる機能です。パンダアバターを運
んでいるユーザーの位置と連動しています。

プライバシーの観点から、位置情報の公開は市区町
村までに制限されます。

3.2 リクエスト機能

自分のパンダアバターを運んでいる人に撮ってほし
い写真や動画をメッセージでリクエストできます。

リクエストの報酬としてアプリ内のコインを与える
ことも可能です。コインはショップ機能で使します。

3.3 ミッション機能

用意されたミッションの達成を目的に、「Tourwith」
を楽しむこともできます。パンダアバターを運べ側、
預ける側の両方のミッションが用意されています。
ミッションをクリアすると、コインを獲得します。

3.4 ショップ機能

コインはパンダアバターの新しいコスチュームや、
「旅を好きな位置から始められる」等の特殊アイテム
との交換に使われます。ショップを活用することで、
オリジナリティ溢れるパンダを作るなど、「Tourwith」
をより楽しむことができます。

4. 特長

我々が作成した Cloud Functions 上の独自のアルゴ
リズムが、ユーザーの直近の行動履歴をもとに、パン
ダアバターを渡す相手を適切に選び、自動で次のユー
ザーに渡します。

これにより、パンダアバターが、多くの人の手を渡
りつつ、目的地に向かいやすくなります。

5. システム構成

Flutter を用いて開発することで、Android と iOS で
使用できるアプリになりました。また、サーバーとし
て Firebase を採用したため、メンテナンスが不要にな
り、安定して「Tourwith」を提供できます。

図1に「Tourwith」のシステム構成図を示します。

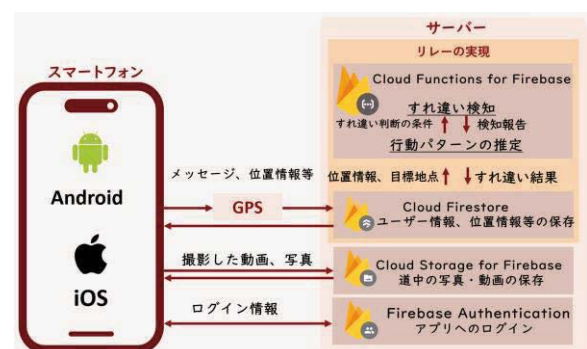


図1 システム構成図

6. おわりに

「Tourwith」で、日常から少し離れた旅気分を体験
してみませんか？

1. はじめに

家族や恋人などの大切な人に自分の気持ちを伝えられていますか？コミュニケーションのなかで家族への感謝の気持ちを伝える機会が少ない、言葉足らずで自分の気持ちが正しく伝えないといった課題がある。本作品は気持ちを「ひとりごと」で終わらせず、相手に伝える「ふたりごと」にして、気持ちを伝える支援をするシステムである。

2. 作品概要



図 1 ふたりんの利用例

図 1 に示すように「ふたりん」は2機1セットで、パートナーなどと1機ずつ持って使用する。機能としては①相手との関係を発展させる上手な伝え方を提案する効果的な伝え方の文章生成、②代わりにメッセージを伝える代理伝達機能の2つがある。

3. ふたりんごとの機能

3.1 効果的な伝え方の文章生成

図 2 に示すように、「ふたりん」は伝えたい内容を理解して、相手との関係を維持発展できる伝え方を提案することができる。

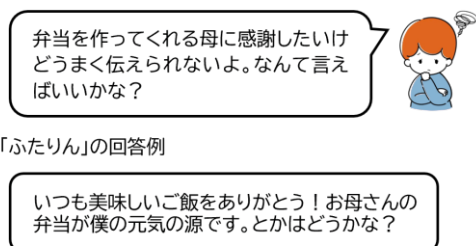


図 2 文章変換の例

3.2 代理伝達機能

直接伝えるにくいことを「ふたりん」が代わりに伝え

ることができる。伝え方は、自分の声を使ったボイスメッセージと、ふたりんがキャラクターの声を用いたメッセージの2つの種類がある。不和・喧嘩をしている時などには、口調や伝え方ひとつで人間関係が壊れてしまったり悪化したりすることもある。「ふたりん」を間に挟むことによって、お互いが冷静に本当の気持ちを伝えることができると考える。

4. システム構成

図 3 にシステム構成を示す。まずハードウェアには、Raspberry Pi Zero を採用している。これにより、Wi-Fi を経由したデータの送受信と、LED やボタン、オーディオ機器などのハードウェアの制御を最小限のスペースで実現することができた。筐体は3D プリンタで作成し、ぬいぐるみ内臓でもアクセサリとしても使用できる。次に、ソフトウェア処理としてユーザの音声データはMicrosoft Azure 上に構築したAPIによって処理される。文字起こし、文章生成、音声合成、データベースへのデータの蓄積を一括して行わせることによって、ハードウェアへの負荷を抑えている。

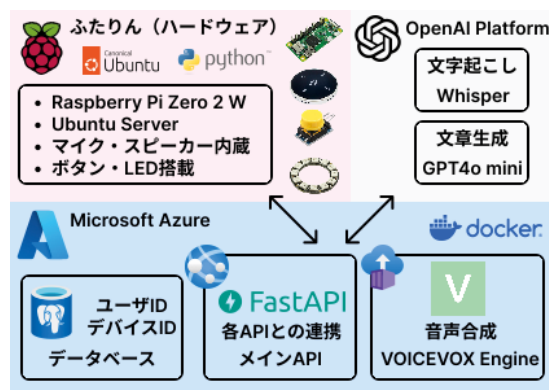


図 3 システム構成図

5. おわりに

本作品の実証実験を行う中で、「ふたりん」の応答が、気持ちをやわらげたり、一旦立ち止まったりすることができる効果を確認した。本作品でより多くの人が勇気を持ち、感謝の気持ちや自分の思っていることを伝えあえる社会につながると期待される。

1. はじめに

学習において暗記は基礎知識の習得や学びを深めるために必要です。しかしその重要性は認識していても、暗記できていないと感じる人は多いです。

そこで、効果的な暗記法の提案を考えました。様々な暗記方法がある中、私たちは「覚え歌」に着目しました。これは覚えたい語句を歌詞とした歌唱曲で「替え歌」、「学習ソング」などとも呼ばれます。メロディーに乗せて繰り返し歌うことで楽しみつつ反復学習することになり、効果的に暗記が出来ます。また、視覚と聴覚など複数の感覚器を使うことも長期記憶に結びつきやすくなり、覚えたものが知識として定着します。

しかしながら、既存の覚え歌は歌詞もメロディーも固定されているものが多いです。「自分の覚えたい語句を好みに合うなじみのメロディーにのせた覚え歌にして歌ってくれるアプリがほしい」そんな夢を実現するのが、私たちが提案する「覚え歌つくるくん」です。

2. システムの説明

2.1 システム概要と構成

本システムではユーザーが歌詞として入力する語句を2.2節で概説する独自のアルゴリズムを用いて、メロディーとなる童謡曲のフレーズに当てはめ、覚え歌の楽譜を作成します。楽譜を作成したのち、VOICEVOX API で合成音声キャラクターの歌声を生成、これと別に作成した伴奏を合わせて覚え歌を再生します。生成した覚え歌や楽譜、入力した歌詞はユーザーのデバイスに保存されます。

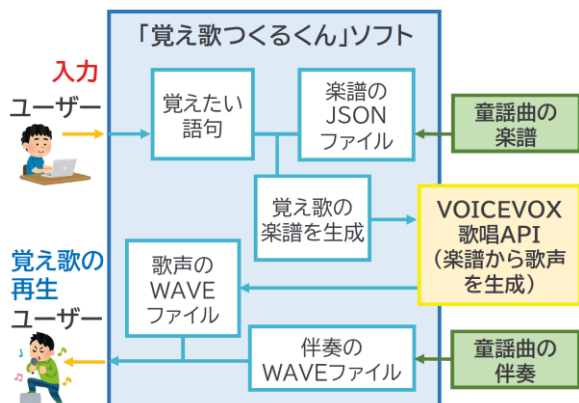


図1 システム構成

2.2 楽譜の自動作成

楽譜を作成するためにフレーズに歌詞を当てはめる際、原曲の歌詞1文字に覚えたい語句の1文字を割り当てるのが基本ですが、曲の区切り（フレーズ）で語句が丁度よく入るようにするために、語句の読みをモーラで取り扱い、モーラ数と音符の数を比較しながら原曲のリズムが崩れないように優先順位をつけて促音や長音を削除する、1音符の長さを変えるといった処理を独自のアルゴリズムを作成して行います。

「幸せなら手を叩こう」の曲の場合

元の歌詞の1フレーズ「しあわせならてをたたこう」

→ここに「札幌、青森、盛岡」という語句を合わせたい

原曲の音符の数 ≠ 語句の文字数（モーラ数）

し あわせ なら て を た た っ っ
さ ぽ ろ あ お も り も り お か } 「っ」を削除して
合わせる

歌詞を「さぽろあおもりもりおか」にすることで解決

図2 促音を削除する例

3. 使い方

3.1 語句の入力

はじめに、歌詞となる「覚えたい語句」を入力します。語句とその読みを入力します。再生画面に表示したい付加情報も入力します。付加情報は覚え歌を作成した後に修正可能です。

3.2 替え歌の作成

リストから覚え歌に使うメロディーを選択します。リストには童謡曲が表示されます。覚えたい語句と曲の相性をシステムが判断し合わせやすい順に並べるため、ユーザーは悩まずに曲を選ぶことができます。同じ歌詞の覚え歌を複数の曲で作成することも可能です。

3.3 覚え歌の再生

覚え歌の生成が完了すると選んだキャラクターの音声で覚え歌が再生できるようになります。画面には歌詞が表示され、歌詞を見ながら覚え歌を聞く、一緒に歌う、また、カラオケ風に伴奏に合わせて歌うことが出来ます。

4. まとめ

覚え歌つくるくんは覚えたい語句を歌詞とした覚え歌を自動生成し、効果的な暗記学習を補助するシステムです。モーラを使ってフレーズに語句を当てはめる処理を行うことで、好きな歌詞で、好きなメロディーを選べるようにしました。色々な方法で覚え歌を何度も歌うことで、楽しく効果的に暗記学習を進めることができます。

1. はじめに

2024年1月1日、石川県能登半島を震源とするマグニチュード7.6の能登半島地震が発生した。この地震により、多くの家屋やインフラ施設が被害を受け、甚大な被害をもたらした。地震発生時、石川県に住んでいる友人に連絡を取ろうとしても通信インフラが被災しているため連絡が取れず、安否確認ができなかった。被災者自身が自力で通信可能な地域まで移動することは大変危険である。そこで、被災者が安全な場所にとどまったままデータ自体に通信可能な地域まで移動してもらい、家族や友人と連絡を取る手法を考えた。スマホからスマホにデータが渡り、安否情報を送受信するシステム「こねくと（仮）」を提案する。

2. 概要

「こねくと（仮）」では、災害時に通信インフラが使用不可能となった状態でも、家族や友人と安否情報を共有し被災状況を把握することができるシステムである。スマホ同士でBluetooth機能を用いたアドホックネットワークを構築し、圏外の端末から圏内の端末まで安否情報を伝達し、安否情報を共有することが目的である。家族や友人の安否情報を確認したい場合は、本アプリや専用サイトを用いることで確認ができる。

3. システムの特徴

3.1. 既存技術の組み合わせ

Bluetooth やアドホックネットワークなどの既存技術を組み合わせることで、広範囲で効率的なネットワークを構築する。

3.2. 軽量データ

データ転送速度と情報量を最適化した軽量データのフォーマットを採用することで、通信負荷やバッテリー消費量を抑える。

3.3. 複数サーバー連携

クラウド上に設置されたサーバーと任意で設置可能なサーバーを連携することで、広域な情報収集と災害

時の通信混雑対策が可能である。

3.4. 過去の位置情報の共有

通信時に過去 24 時間の位置情報を共有することにより、被災者の移動履歴を把握し、安否確認や支援活動をより効率的に行うことが可能である。

4. 動作イメージ

4.1. 通常時

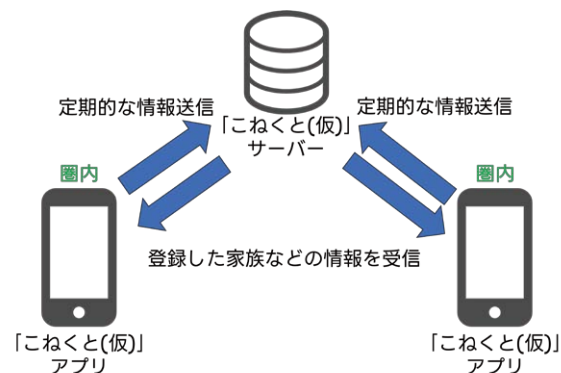


図1: 通常時の動作イメージ

4.2. 災害時

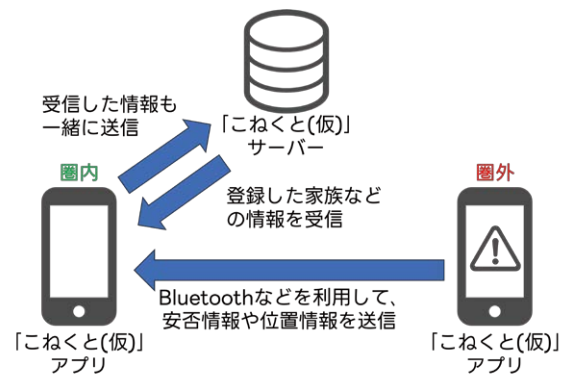


図2: 災害時の動作イメージ

5. 終わりに

近年、南海トラフ巨大地震や首都直下地震などの大規模な災害への警戒が高まっている。通信が社会のインフラを担っている中、大規模な災害発生時に通信を支える技術があったら、とても心強いものになるはずである。本システム「こねくと（仮）」が、災害時に被災者の不安を軽減させることができるものになると期待している。

1. はじめに

「SPORTSDAY」は、球技大会の運営をスムーズにするための Web アプリケーションです。従来、結果集計や試合の進行状況を把握するためにアナログな方法に頼っていたため、集計に時間がかかり、リアルタイムでの情報共有が困難でした。このアプリは、リアルタイムで結果登録や試合の進行状況の把握ができます。

2. システム概要

「SPORTSDAY」は、球技大会に参加するユーザと運営者を対象とし、運営をスムーズにしユーザが簡単に情報を得られることを目的とします。

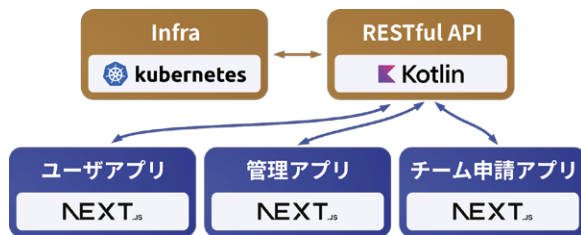


図1 システム構成概略図

3. 機能

3.1 ユーザアプリ

参加している競技や試合予定などのユーザに関連のある情報を簡単に得ることができます。従来は、PDF 形式の試合予定表から自分が参加する試合を探していましたが、簡単に見つけることができませんでした。



図2 ユーザアプリ「SPORTSDAY Panel」

3.2 管理アプリ

リーグ、トーナメントの試合作成を行うことができ、試合結果の入力を行うと、リーグやトーナメントの結

果を自動で算出することができます。また、天候に応じて競技の開催方法が異なる場合でも、柔軟に表示する情報を切り替えることができます。



図3 管理アプリ「SPORTSDAY Admin」

3.3 チーム申請アプリ

チーム申請アプリでは、チームメンバの登録を簡単に行えます。代表者がチームメンバの登録を行った後に、各ユーザ自身が所属するチームを確認することで、入力ミスを最小限に防ぎます。

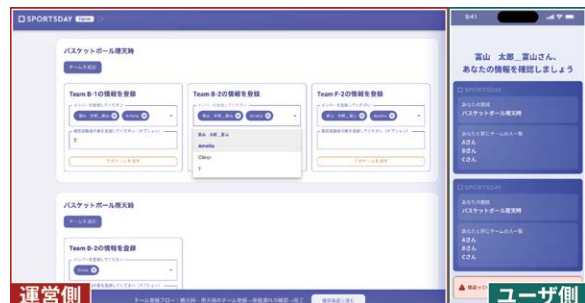


図4 チーム申請アプリ「SPORTSDAY Form」

4 まとめ

「SPORTSDAY」はこれらの機能により、球技大会で抱えていた課題を解決し、運営をスムーズに行うことができます。

2023 年度からの運用を通じて、学生や運営からのフィードバックを反映しながら改良を重ねており、情報の取得のしやすさや入力ミスの削減を図ることで、より効率的で快適な大会運営を実現しています。

1.1 はじめに

チャットやメールを使い、テキストで文章を書くことが当たり前の時代になりました。

それに伴って会話だけでなく、テキストでも方言を利用したい場面があります。しかし、一般的なキーボードアプリでは方言が正しく変換されず、予測変換に出ることもありません。

あなたのキーボードに入力を方言へ変換する補助機能があれば、もっと気楽に方言を使用するようになるのではないのでしょうか。

1.2 方言の重要性

方言は長年その地で生活してきた人々により培われた価値観や地域環境を表現するユニークな言語です。

現在日本中で方言が使われていますが、世代を追うごとに少しずつ使う方言が薄れてきています。

方言の使用を活性化するために、テキストチャットなどでも方言を簡単に入力できるキーボード補助アプリを提案します。

2. 機能と使い方



図1 常時フローティングのイメージ



図2 誤変換防止モードの使用例

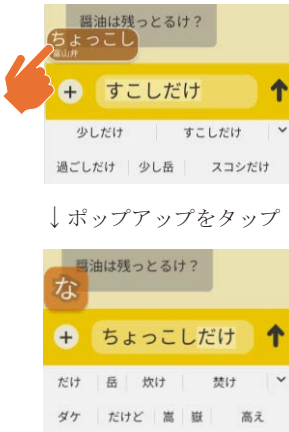


図3 共通語→方言モードの使用例



図4 地域設定画面

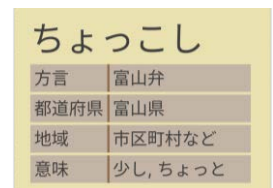


図5 方言の詳細表示

2.1 誤変換防止モード

入力された方言を検知し、ポップアップで方言としての正しい変換を提案する機能です。なまっ talk はキーボード補助アプリなので、使い慣れた配列や愛着ある背景設定はそのままに、共通語と方言の両方をストレスなく使い続けられるよう設計しています。(図2)

2.2 共通語→方言モード

共通語を入力したときに、方言での言い換えを提案する機能です。(図3)

2.3 方言の地域選択

提案する方言の地域を複数設定することも可能です。方言バイリンガルの方、自分の興味や出張先などでの他の地域の方言を学びたい(知りたい)方にも便利です。(図4)

3. データベースについて

現時点では、富山弁を中心に書籍を用いてデータベースを作っています。今後は、日本全国に対応した新しく、より多いデータベース作成のために、クラウドソーシングを活用した情報収集を予定しています。

1. はじめに

近年、Vtuber 市場は、2023 年度に 800 億円程度の規模となる見込みである、2020 年比で約 5 倍になるとみられています。また、国内におけるメタバース市場も急速な拡大が予測されており、2027 年には 2 兆円にも上る規模になると予測されています。そのため、これらの市場の拡大によってアバターの需要が高まっていると考えられます。

現状、Live2D アバターの表現方法には 2 種類あり、内部の人間の表情をトラッキングしたものを Live2D モデルに反映させる方法と、あらかじめ用意した表情や動きを内部の人間が手動で選択する方法があります。これらの方法ではゲーム配信などの両手を離せない状況で、「涙を流す」のような表情に手動で切り替えることは困難です。そこで、私たちは機械学習による豊かな感情表現を可能にする新しいトラッキングソフトウェア、「エモアンプ」を提案します。

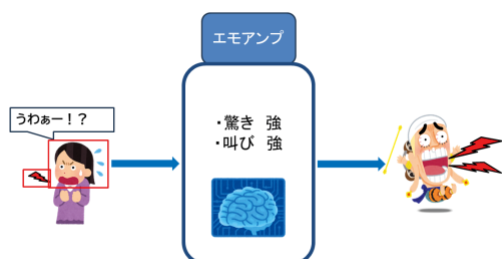


図 1 エモアンプのイメージ

2. 概要

エモアンプは従来のトラッキングに加えて、感情推定等によって独自の表現を可能にするソフトウェアです。ウェブカメラから顔表情、マイクから音声を受け取り、顔表情、声色から感情を推定して、自動かつ連続的に表情を変化させていきます。また、音声から母音を推定し、母音に対応した口の形にしています。

2. 感情推定について

エモアンプでは、ウェブカメラから顔表情、マイクから音声を受け取り、顔表情、声色から感情を推定します。その結果を利用して、パーティクルを追加したり、怒っている表情に変化させたりします。これによって、トラッキングのみでは困難な、アニメ風の表現を自動的に行えるようにします。

3. システム構成

エモアンプのシステム構成を図 2 に示します。PC のマイクおよびウェブカメラで映像と音声を受け取り、Python 上で表情のトラッキングや各種推定を行います。その後、その結果を Unity へ出力し、Unity 上で Live2D モデルの操作を行います。

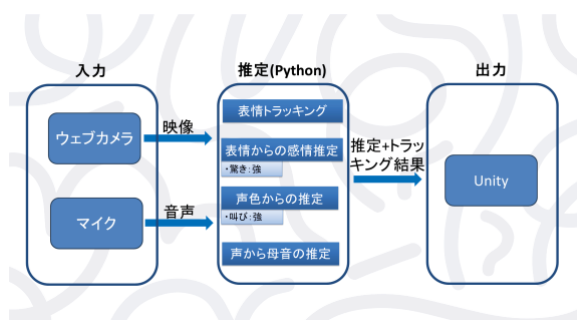


図 2 システム構成図

4. おわりに、まとめ

この「エモアンプ」を使うことによって従来のトラッキングに比べてより豊かな感情表現を行うことができ、さらに、キャラクターの操作に割くリソースを減らすことができるため、より充実した配信を実現することができます。

1. はじめに

沖縄県では、建物や車などを劣化、腐食させる「塩害」が発生します。塩害は、海水が潮風などによって巻き上げられ、建物や車などに飛散することで引き起こされます。さらに、塩害には「目に見えない」、「予報がない」、「情報が無い」といった厄介な点があります。そこで私たちは、塩害の情報をユーザーに分かりやすく伝えるために「シオサビン」を考えました。

2. システム概要

シオサビンは、屋外に専用の塩害センサーを設置し、センサーから得られた情報をアプリケーションに送信します。そうすることで、詳細、広範囲かつ未来の情報を得ることができ、誰でも手軽に塩害への予防と対策ができるようになるシステムです。

2.1 シオサビンアプリ

シオサビンアプリには、主に3つの機能があります。

1つ目は、塩害通知機能です。センサーの情報をもとに塩害の程度を表示し、センサーの運用のために交換が必要な廃液の量などを表示します。

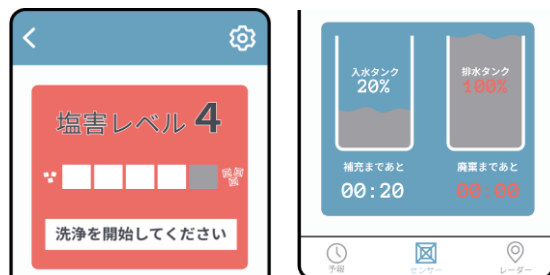


図1 塩害通知画面

2つ目は、塩害レーダー機能です。この機能は、各地点のセンサーの情報をもとに、マップに場所ごとの塩害レベルを表示します。また、過去1日分の被害状況も表示することができ、ユーザーは広範囲における過去の塩害の状況を容易に理解することができます。

3つ目は塩害予報機能です。この機能は、センサーが収集した情報をもとに未来の塩害被害を予測して表示します。



図2 塩害レーダー画面と塩害予報画面

2.2 塩害センサー

塩害センサーは、電解質（イオン）の性質を利用します。水溶液中のイオンが多いほど、導電率（電流の流れやすさ）は大きくなります。潮風によって付着した塩分を洗い流した水溶液中の伝導率を測定することで塩害かどうかを判定することができます。

3. システム構成

塩害センサーからの情報は、サーバに送られ、気象庁のAPIとpython製の予報AIを利用して、アプリ側で処理しやすい情報に変換されます。

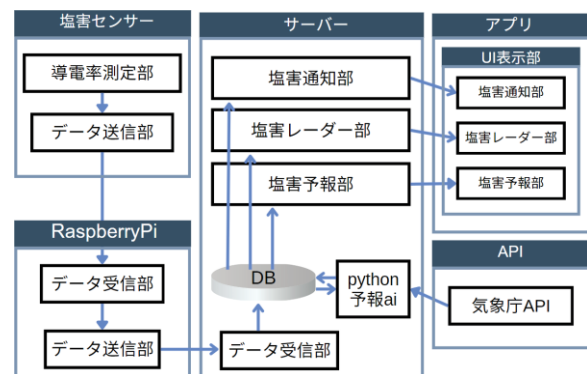


図3 システム構成図

4. さいごに

塩害は、車や自宅、自転車など大切なものへ容赦なく被害をもたらします。「シオサビン」では、そういった被害を少しでも減らすことができれば幸いです。

1. はじめに

皆さんは、何かお魚を飼っていますか？水の中でヒレを揺らし、優雅に泳ぐ観賞魚の姿は、見ている人々の心を癒してくれます。しかし、観賞魚は犬や猫などのような触れ合い方ができません。ともに日常を過ごす家族として今より触れ合えてもっと愛着が湧いたら、日々の生活もより楽しくなると私たちは考えこの「フィッシュフレンズ」というシステムを考案しました。

2. フィッシュフレンズについて

フィッシュフレンズは、観賞魚を飼育している水槽内をモニタリングし、お魚との疑似的な交流ができるアプリケーションです。水槽内のお魚の様子から活発度や性格の分類を行うことで、飼っているお魚の個性を表現します。

3. システム概要

システム概要の図を図1に示します。

本システムは RaspberryPi5、USB カメラ、Unity で主に構成されています。フィッシュフレンズは、お魚の挙動を測定し、どれくらい活発に動くか、どんな性格かを決める「活発度・性格診断」と、3D 空間で飼っているお魚と会話ができる「会話モード」の2つの機能を提供します。「活発度・性格診断」には YOLOv5-Small を用います。30 秒に 1 回、水中画像を取得し、Google drive にアップロードします。その画像を元に、YOLO でお魚の移動距離や滞在場所を判別し、活発度や性格を診断することができます。

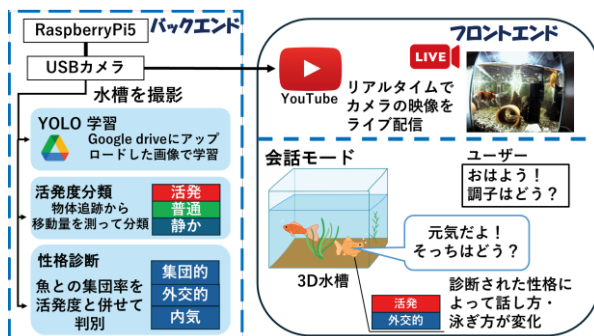


図1 システム概要

4. 機能概要

4.1 活発度・性格診断

一匹一匹に個性があると、それぞれに愛着もわきやすいのではないのでしょうか。この機能では、一定時間お魚の動きを YOLO と Deep SORT を用いて追跡し移動量を計算します。この移動量に応じて活発度は、主に「活発」・「普通」・「静か」の3段階で決定します。この段階に加えて、「異常に活発」・「異常に静か」の2段階があり、これはお魚に異常があることを検知した段階で、この段階に決定された場合はユーザーにその旨のメッセージがアプリ内で表示されます。また、YOLO で、複数の魚が一緒にいる（複数のバウンディングボックスが一定距離に存在する）と判断した際、接近から離れるまでの時間からどれくらい他のお魚と一緒にいるかを測定し、活発度と共に性格診断に使われます。これら2つの要素が「会話モード」でのお魚のふるまいを変化させます。

4.2 会話モード

会話モードでは、Unity で作成した 3D 空間で水槽内部のお魚たちの様子を、事前に 3D モデル化した水槽で見ることができ、活発度と性格から泳ぎ方も変化します。水槽内のお魚たちと会話することもでき、話したいお魚に触れるとチャットが開き、会話文を入力すると、ChatGPT の API を通してお魚からの返答が返ってきます。会話の内容も活発度と性格によって変化し、それぞれの個性が異なるお魚との会話を楽しむことができます。

5. 終わりに

前述したように、お魚は犬や猫のような触れ合い方がなかなか難しい生き物です。

「フィッシュフレンズ」は、お魚を飼っている人、また、これからお魚を飼う人たちに新たなお魚との暮らし方を提供します。

1. はじめに

皆さんはセルフレジを利用しているとき「何度やっても読み取ってくれない」、「バーコードがなかなか見つからない」など思ったことはありませんか？この自動学習精算機「Regina」は、バーコードをスキャンしなくても商品を認識してくれるため、そんな不満を解消してくれます。「Regina」が普及し、店員、顧客の負担軽減につながることを目的としています。

2. 概要

「Regina」はパッケージを自動で学習しバーコードを紐付け、商品をどの角度からでも読み取ることができるシステムです。例として、とあるコンビニは現在全国に20,000店舗以上あります。各店舗に1台「Regina」があると仮定した場合、各店舗で同一商品を一度読み込むだけで一気に20,000以上のデータが手に入ります。一瞬で記録・反映が可能になります。

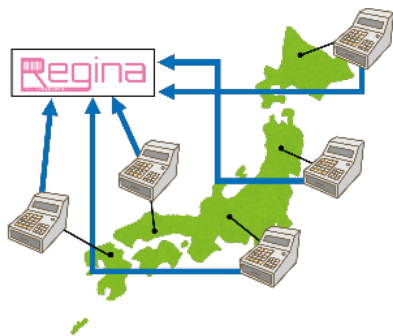


図1 イメージ図

3. 機能

- **パッケージから商品の読み取り**：バーコードを読み取る際にパッケージを学習します。再度スキャンする際には、パッケージから商品を推定でき、バーコードを探さなくてもよくなります。さらに、読み取ったパッケージ写真を使い、再学習させることで読み取り精度を高めることができます。
- **バーコード読み取り**：従来のレジと同様にバ

ーコードを読み取り、商品のデータベースから価格などの情報を引き出すことができます。

- **値引きシール対応**：企業ごとに値引きシールは統一されているため、あらかじめ値引きシールの学習データを用意することで対応可能です。
- **バーコードなし商品対応**：野菜などのバーコードが無い商品については、あらかじめ学習データを用意することで対応させる。(※野菜については規格品であればある程度の形や大きさなどが統一されているため、ほぼ認識可能。どうしても対応できない商品についてはレジのタッチパネル操作での対応になります)

4. システム構成

Reginaのシステム構成を図2に示します。レジ側で撮影画像をコンピュータに送り、4つのパターンで認識できるようにします。この時、値引きシールのみ予め用意してある学習データを使い対応します。その後サーバ側で識別AIが学習データをもとに候補や信憑性を検証し、バーコードやパッケージの商品名・価格の紐付けを行い、レジ側でレシートや合計金額の表示を行います。

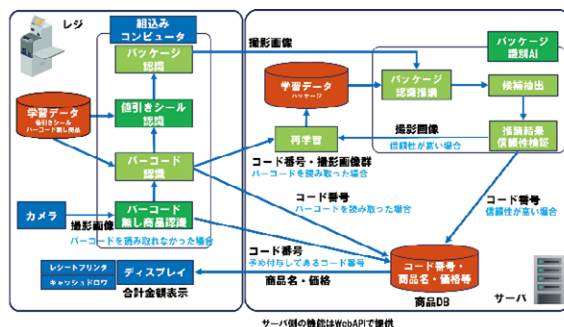


図2 システム構成図

5. おわりに

この自動学習精算機「Regina」を使えば、作業負担軽減やコスト削減が期待できます。

1. はじめに

学生寮では、大浴場や洗濯機などの共用設備が日常生活において必要不可欠である。しかし、これらの設備は時間帯によって混雑したり、利用するまでの待ち時間が長かったりと、不便さを抱えており、多くの寮生が不満を感じている。この問題を解決するために、学生寮の共用設備の利用状況を可視化し、寮生の不満を解消するシステム、「どみとる」を開発した。

2. システム構成・概要

今回開発したシステムでは、寮内の共用設備である大浴場、シャワー室、洗濯機、乾燥機に各種 IoT デバイスを設置し、利用状況が判別できるデータを取得する。取得したデータをもとに Web アプリで各設備の利用状況を表示することで、ユーザはそれぞれの設備の利用状況を確認するために足を運ばずとも、各々のデバイスから共用設備の利用状況を確認できるようになる。

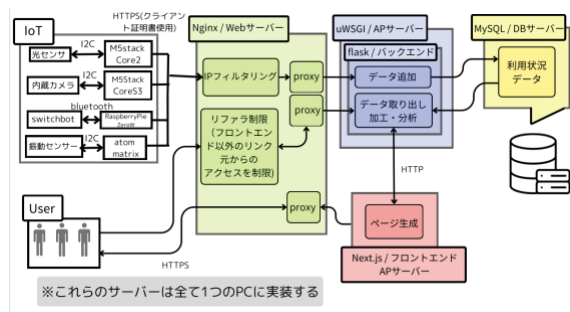


図1 システム構成図

2.1 大浴場

M5Stack Core3 を用いて大浴場の玄関を撮影し、サーバへ画像を送信する。サーバではYOLOv8を用いて画像中の靴が何足あるかを認識することで利用人数を判別する。

2.2 シャワー室

M5Stack Core2 と光センサーユニットを用いて照明の点灯・消灯を認識することで利用状況を判別し、結果をサーバへ送信する。

2.3 洗濯機・乾燥機

洗濯機・乾燥機それぞれに取り付けられたSwitchBotプラグミニで測定した消費電力をBluetooth経由でRaspberry Pi Zero Wに送信する。送られてきた消費電力によって使用状況を判別し、結果をサーバへ送信する。この際、Raspberry Pi Zero Wは複数台のSwitchBotプラグミニからデータを受け取り、まとめて結果をサーバへ送信する。

2.4 リアルタイムでの利用状況の可視化

WebアプリはフロントエンドをNext.js、バックエンドをFlaskで実現している。フロントエンド側の処理でバックエンドから情報を受け取る頻度を一定間隔に設定することで、常に最新のデータをユーザは確認することができる。



図2 アプリトップ画面

2.5 利用データの分析

その日の今までの利用データおよびバックエンドで指定した期間（直近1ヶ月など）で集計した利用データをグラフ化したもの、現在利用されている設備が利用され始めた時間をアプリで表示する。設備が混雑している時間を可視化することで混雑解消と待ち時間の短縮につながる。

3. おわりに

「どみとる」を活用することで、寮の共用設備の利用効率が大幅に向上する。このツールは学生寮だけではなく、ホテルや公衆浴場など多くの人が設備を共有する施設でも利用可能である。

1. はじめに

地球温暖化の進行に伴い、異常気象による河川の氾濫リスクが年々高まっています。増水や氾濫の危険が迫った際、河川の状態を確認しようと現場に向かう人が事故に遭うケースも少なくありません。このような状況の中で、河川の状態を迅速かつわかりやすく把握できるシステムの必要性がますます高まっています。私たちはこの課題に対処するため、河川の水位情報を視覚的に表現し、誰でも簡単に状況の把握ができるシステムの開発に取り組みました。

2. Nahlun が提案するもの

Nahlun は、河川状況を 3D で可視化する手法を提供します。現在、国や多くの自治体が河川に水位計を設置し、計測したデータを Web サイト上に公開しています。しかし、これらの Web サイトで提供される情報だけでは、ひと目で現場の状況を把握するのは難しいという問題があります。これを解決するため、Nahlun はボクセルデータで表現された地形上に水面を可視化し、直感的に現場の状況を理解できる方法を提案します。

3. システムの概要

河川全体をグラフデータベース上で管理し、リアルタイムで水位情報を反映しています。水位計のデータを元に水深を予測し、予測した水位が変化した箇所をクライアントへ更新を通知しています (図 -2)。

地形の 3D データはオープンデータを基に生成しており、より重要度の高い地形には任意のデータソースから生成したボクセルモデルを差し込めるようにしています。これによって、地形全体の整合性を保ちながら詳細な地形を表現したい部分にだけ投資をすることができます。

コンテスト中は VPS での運用を想定しているため、サーバーの性能に制約があります。すべてのデータをストレージに持つておくことはコスト的に不可能です。そのため、DEM(数値標高モデル)や地形分類

情報は国土地理院や産総研などの外部がホストする情報から適宜取得し、必要に応じて地形を表現する 3D モデルや水面形状などを計算して使用頻度の高いもののみキャッシュするようにしています。

3. 今後の展望

今回は短い開発期間の中で、最も重要な 3D での可視化部分を優先的に実装しました。しかし、サービスとしてより良いものにするためには、さらなる機能追加が求められます。例えば、現状では管理者ツールは TUI で提供されていますが、PC に不慣れなユーザーでも操作できるよう、管理用の GUI の実装が必要です。また、地物の検索機能などの地図情報サービスとしての使いやすさも重要な要素です。河川の水位情報に応じて外部の API を呼び出すなど、フック機構を導入すれば、より多様なニーズにこたえられるシステムが実現すると考えています。

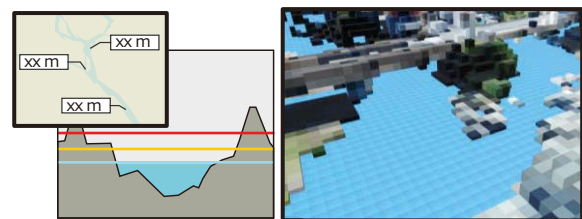


図 - 1 従来の可視化方法(左)と
3D での可視化方法(右)

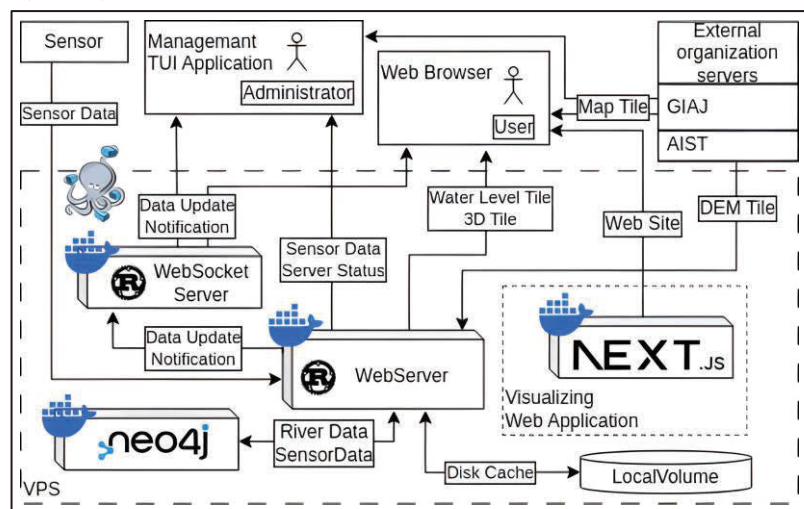


図 - 2 システムの構成

1. はじめに

多くの人が公園や山で偶然虫を見つけたとき、この虫の種類はなんだろうと思い、調べようとしたことがあるのではないのでしょうか。スマートフォンを使って調べようすると、スマートフォンを取り出し、Web 検索を行う必要があり、目的の情報を得るまでにかなりの時間と手間がかかります。

そこで私たちはスマホを取り出さずにハンドサインによって素早く虫を調べることができる「この虫な〜んだ？」を提案します。

2. システム概要

私たちのシステムは、ハンドサインにより虫を指し示すと虫の種類やその虫の情報を音声で教えてくれるシステムです。図1に概要を示します。

使用方法については以下の通りです。

1. 最初にウェアラブルカメラを頭に装着しアプリを起動したらスマホにウェアラブルカメラを USB 接続する。
2. その状態で虫を囲むようにハンドサインを行う。
3. ウェアラブルカメラで得た画像により虫の認識を行い、音声でその虫の種類等を教えてくれる。また



図1. システム概要

端末の画面には認識した虫の画像と音声と同じ虫の情報が表示される。

3. 機能の詳細

私たちのシステムの主な機能はハンドサイン認識による画像の切り出し処理と画像による識別処理の2つの処理で実現しています。

3.1 ハンドサインの認識

ハンドサインの認識とは、USB カメラから取得した映像からハンドサインを認識し、ハンドサインで指定した範囲をトリミングし画像として取得する機能のことです。MediaPipe と TensorFlow Lite を用いてハンドサインを認識します。ハンドサインは人差し指のみ立てたものとしています。また、ハンドサイン中の人差し指の指先の座標の最大値と最小値を用いて画像のトリミングを行います。

3.2 虫の画像認識

ハンドサイン認識の際に取得した画像から、画像分類モデルを用いて虫の種類を判別する機能です。

TensorFlow Lite を用いて、ハンドサイン認識の際にトリミングされた画像に対し推論を行い、昆虫を種類ごとに分類します。画像認識の結果をもとに、音声での説明、アプリ内で昆虫の文での説明を出力します。

この際に用いるモデルは、CNN の MobileNetV2 をもとに学習を行ったモデルから、さらに転移学習を行って作成しました。

4. おわりに

「この虫な〜んだ？」によって、虫のことに少しでも詳しくなってもらえたら嬉しいです。さらに、少しでも多くの人が虫のことに興味を持ってもらえたら幸いです。

Abstract

Farming plays a crucial role in society by supplying food, generating employment, and driving economic growth. In countries like Thailand and Japan, agriculture is a major contributor to GDP and a significant source of jobs. However, the sector faces challenges such as climate change, soil degradation, and limited available land.

To highlight these challenges, we propose a game called "30 Days Harvest." In this game, players assume the role of a farmer tasked with managing and developing their farm to maximize profits from crop sales. Along the way, they must navigate obstacles like seasonal changes, temperature fluctuations, and space constraints.

Keywords: Farming Game, Global warming game, Strategy game

1. Introduction

30 Days Harvest is a single-player PC game designed to raise awareness about the environmental impacts on agriculture. Players step into the shoes of a farmer who must develop and manage their farm while navigating a range of environmental challenges, discovering strategies to overcome these obstacles and sustain their farm's success.

2. System overview

30 Days Harvest aims to raise awareness about the environmental impacts of global warming on farming. To achieve this, we have incorporated several environmental challenges into the game, including:

- **Environmental challenges:** To maximize crop production within the limited farming space, players must optimize environmental factors such as temperature, humidity, and air pollution levels to meet the specific needs of their crops. To do this, players must strategically plant specific types and quantities of trees within the same farming area. We believe that these strategies will help players gain a deeper understanding of the impact of global warming on agriculture.
- **Water Pollution:** Water is a critical factor in agriculture, and its quality greatly affects crop production. To highlight the importance of water quality, players are tasked with regularly collecting trash from the river to ensure a clean water supply for their farm.
- **Animal Invasion:** To add excitement to the gameplay, an animal invasion event is included. Players must protect their crops from insects and wild animals that threaten to eat or destroy them.

An example of a game screenshot is provided in Figure 1.



Figure 1: Example of 30 Days Harvest screenshot.

3. Experiments

We evaluated *30 Days Harvest* using nine male participants (average age 21.44 ± 0.53 years) based on three objectives: 1) raising awareness of environmental impacts on agriculture, 2) providing enjoyment, and 3) encouraging analytical thinking. Participants completed six days of gameplay and answered evaluation questions as follows:

- **Raising awareness of environmental impact:** After the gameplay, participants were asked to identify all environmental factors that influence crop production and explain how to manipulate these factors.
- **Providing enjoyment:** Participants were asked to rate their enjoyment on a scale from 1 to 5, where 1 means "dislike it" and 5 means "love it," in comparison to the baseline game, *HayDay*.
- **Encouraging analytical thinking:** We propose that achieving the highest production during gameplay requires analytical thinking. Therefore, the number of crop production of all participants is recorded.

4. Experimental results

The experimental results reveal that

- **Raising awareness of environmental impact:** 55.6% of participants identified two factors—humidity and temperature—while the rest correctly identified all three, including air pollution. Additionally, 88.9% knew how to manipulate these factors for optimal crop production.
- **Providing enjoyment:** While *30 Days Harvest* scored 3.44 ± 1.01 out of 5 in satisfaction, *Hay Day* scored 3.67 ± 1.12 . We suggest that the small margin between these scores indicates that our game is comparable to a well-known market leader.
- **Encouraging analytical thinking:** With a strategic plan, about 33.33% of participants were able to achieve maximum crop production.

5. Conclusion

30 Days Harvest is a farming game aimed at raising awareness of environmental impacts on agriculture. While it successfully raises awareness and provides enjoyment, it falls short in promoting analytical thinking, likely due to insufficient feedback on factors affecting crop production. We plan to address these issues to fully achieve all three objectives.

Abstract

AfterDay Horizon is a collaborative VR and web-based multiplayer game that aims to raise awareness of global warming and demonstrate the potential consequences if we fail to address this issue soon. Our game presents a world where civilization has collapsed due to global warming. People have taken refuge in bunkers to protect themselves from external dangers such as extreme temperatures, radiation, food and water shortages, and more.

Keywords: Multiplayer Game, Virtual Reality, Survivor Game, Collaborative Game, Global Warming, Unreal Engine

1. Introduction

Our world is on the brink of a critical turning point. The Intergovernmental Panel on Climate Change (IPCC) warns that any further delay in global action to slow climate change and adapt to its impacts "will miss a brief and rapidly closing window of opportunity to secure a livable and sustainable future for all." By 2100, about 50-75% of the global population might face "life-threatening climatic conditions" caused by extreme heat and humidity, leading millions to acute food insecurity and reduced water security.

Everyone in the world contributes to global warming, whether a little or a lot. Thus, it is also our responsibility to mitigate this problem for future generations. Based on our expertise, we propose "AfterDay Horizon," a collaborative VR and web-based multiplayer game designed to raise awareness about global warming and highlight the potential consequences of inaction.

2. System overview

AfterDay Horizon is a two-player game that emphasizes collaboration between a VR player and a web-based player. A compatible VR headset, such as the Meta Quest 2 (also known as Oculus Quest 2) or a more advanced model, is required.

In *AfterDay Horizon*, players take on the following roles:

- **Caretaker:** A player who engages with the game using a VR headset.
- **Leader:** A player who resides in the bunker, providing the caretaker with necessary information and guidance through a web-based manual.

Together, they must ensure the survival and well-being of the bunker's inhabitants by completing four critical tasks within a limited time frame:

- Growing crops to increase the food supply.
- Repairing the oxygen system.
- Fixing the plumbing system.
- Restoring the electrical system.

The *AfterDay Horizon* screenshot is shown in Figure 1. The success of each mission directly affects the well-being of the people in the bunker. For example, failing to replenish the food supply on time could result in a 50% decrease in the bunker's population.



Figure 1: Example of *AfterDay Horizon* game-play

3. Experiments

Several experiments were conducted with 8 participants, consisting of 3 males and 5 females, with an average age of 20.75 ± 0.433 years. They were asked to complete the gameplay at least once to evaluate our game based on the following three objectives:

- 3.1. **Promotes collaboration:** Players must collaborate to complete the game, so we measured the completion time to assess this.
- 3.2. **Enhances strategic thinking:** The game requires strategic planning to overcome challenges within a limited time. We used the completion rate to evaluate this aspect as well.
- 3.3. **Offers enjoyment:** Participants rated their enjoyment on a scale of 1 to 5, with 1 being "hate it" and 5 being "love it."

4. Experimental results

The experiment with 4 pairs of participants reveal that:

- 4.1. **Promotes collaboration and enhances strategic thinking:** All participants successfully completed the missions, with an average completion time of 27.10 ± 4.36 minutes. While every pair finished the missions through effective collaboration, the average completion time exceeded our expectations, surpassing 20 minutes. We suggest that the complexity of the game necessitates a more strategic approach.
- 4.2. **Offers enjoyment:** Our game achieved a satisfaction score of 3.75 out of 5, indicating that participants generally leaned toward positive feedback.

5. Conclusion

Based on the experimental results, we suggest that *AfterDay Horizon* demonstrates satisfactory performance. While the game effectively fosters collaboration among players, it slightly underperforms in enhancing strategic thinking. To address this, we plan to introduce additional hints for efficient planning. Despite the positive feedback from participants, as reflected in the satisfaction scores, we aim to further improve these scores and benchmark them against a strong baseline in the near future.

1. Introduction

Do you like cats? *Nekotorae* is a VR experience where you play as a cat caretaker. However, the cats are very mischievous and are always trying to escape! Try your best to keep as many cats in the room as possible in this hectic experience without letting any of them escape! Have fun!



2. Features

2.1 Gameplay

In the beginning, there was only one cat. As time goes by, more and more customers will come by and drop off cats. The cats will try to escape by any means: running out of the door, crawling through the vents or even jumping out of the window! (don't worry, they are perfectly safe) Your job is to not disappoint the customers by losing any of the cats. Use anything at your disposal to do the job.



2.2 Cats!

There are so many kinds of cats! Different cats have different abilities and will require different way of keeping them in!

There are:

Normal cat: a normal cat that walks around

Flying cat: this cat can fly! It makes it harder to catch!

Spaceman cat: this cat ignores gravity and bounces everywhere—on the ceiling, on the floor, everywhere!

And many more other cats to discover!



2.3 Upgrades

To handle more cats, you can unlock upgrades to help you.

There is laser pointer to some cats, a sticky ball to stick cats together and many more!



Come find us at our booth to have a good time!

1. はじめに

外国から日本の高等専門学校に通う学生や、日本で学び、働くことを希望する外国人にとって、日本語の語彙や漢字の習得は困難である。このような学習者は通常、フラッシュカードや新しい単語帳を利用して単語を書いて覚える方法を取るが、このアプローチは必ずしも効果的とは言えない。カスタマイズ可能なフラッシュカードや専用の日本語学習アプリケーションも存在するが、これらは競争要素やコミュニティへの貢献が不足している。これらの既存の学習方法を統合し、より興味深く、よりカスタマイズ可能な学習体験を提供するために、私たちは「QuizRush」という学習ツールを開発した。

2. 概要

「QuizRush」は主に2つのモードで構成されている。一つはシングルプレイヤーモードで、これは教師や他者、生徒自身がアップロードした学習データを用いて練習するためのものである。もう一つは、インターネットを介して他のプレイヤーと対戦するためのマルチプレイヤーモードである。ゲームの基本的な仕組みはシンプルで、プレイヤーに対して単語や漢字に関する質問が提示され、その回答に基づいてポイントを獲得する。獲得したポイントは、アイテムやスキルの取得、アバターの強化に使用される。プレイヤーはこのアバターを用いて対戦相手と戦う。ラウンドの時間が終了するか、対戦相手が敗北すると、残りの質問が表示され、図1に示すような学習のループが形成される。ゲームはデフォルトで3ラウンド後に終了する。

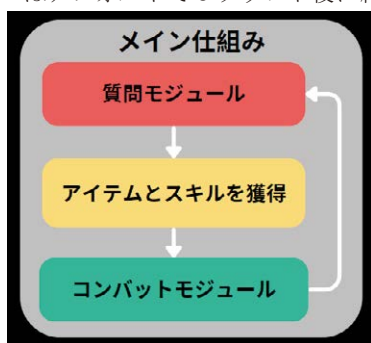


図1. メイン仕組み

3. システム構成

本システムは、二つの主要なモジュールから構成される。これらのモジュールを使用して、さまざまなゲームプレイモードを提供する。このような複数のゲー

ムプレイモードは、異なる学習視点を提供し、効果的な学習支援を実現する。

3.1 質問インプット

質問インプットモジュールは、データベースにデータをアップロードする役割を果たす。さらに、レベルで使用する質問を決定し、質問の形式（選択肢形式や文の完成形式など）を設定する。このプロセスにより、学習内容のカスタマイズが可能となり、個別の学習ニーズに応じた調整を行うことができる。

3.2 ゲームシステム

ゲームシステムモジュールは、完全な戦闘型 RPG やターン制のゲームメカニクスを推奨する。プレイヤーは、アバターやマップの選択、ラウンド数のカスタマイズなど、さまざまなゲームプレイパラメータを調整することができる。このモジュールにより、プレイヤーは自分の好みや学習目標に合わせてゲーム体験を調整し、より柔軟な学習支援を受けることができる。

4. 特徴

本ゲームの設計は、レゴブロックのように、ユーザーが質問データとゲームプレイ要素を選択してレベルを構築する形式である。ユーザーは、自分の学習ニーズに応じて、個別の「ブロック」を組み合わせることで、カスタマイズされたレベルを作成することができる。このアプローチにより、学習内容やゲームプレイの多様性が確保され、より適応的な学習体験が提供される。

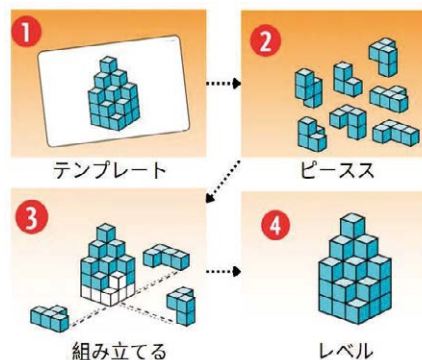


図2. レベル構築構図

5. まとめ

本ゲームの開発により、オープンソースで無料かつ楽しみながら日本語を学ぶことが可能になると信じている。さらに、日本語の学習にとどまらず、自然科学や技術に関するレッスンなど、複数の知識分野に対応するセクションを作成することを目指している。

Abstract

Sign language plays an essential role in communication for millions of Deaf individuals worldwide. However, the availability of qualified sign language interpreters is limited, and interpreter services can be expensive and inaccessible in many situations.

This paper focuses on the WLASL dataset, which is a collection of word-level American Sign Language (ASL) videos. By utilizing deep learning techniques, particularly CNNs and LSTMs, machine learning systems can process sign language videos and translate gestures into spoken or written language in real time. Despite significant progress, there are challenges related to accuracy, context understanding, and handling variations in sign language across regions.

Methodology

Training the Model: The combined CNN-LSTM model is trained on the WLASL dataset, which is split into training and validation sets. The model learns to recognize gestures from the videos and map them to corresponding word-level labels. Data preprocessing includes extracting frames from video clips, normalizing input sizes, and labeling gestures to ensure consistency across the training data.

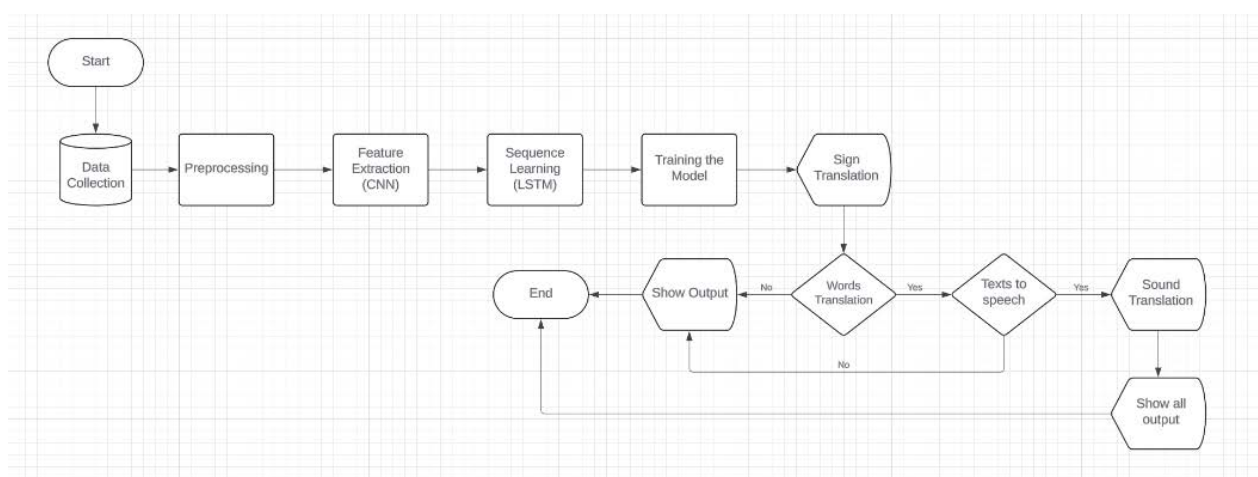
Results and Discussion

The machine learning model trained on the WLASL dataset has demonstrated promising results, achieving a high in recognizing word-level signs. It shows us the output of translation of the sign-language into English and the user can be able to translate the phrase into other language and read it out loud. This is a significant improvement over previous approaches that relied solely on traditional image processing techniques. However, there are areas where further improvements are

1. Real-Time Processing: Although the system can operate in real time, occasional delays or inaccuracies in gesture recognition occur due to factors like lighting conditions or the speed of gestures.

2. Expanding Beyond Word-Level Translation: The current model is limited to word-level translation. To achieve sentence-level understanding, future models need to incorporate contextual cues, including body language and expressions that convey emotions or intentions.

Figure 1: Sign Language Translation Flowchart



応募全作品

■課題部門「ICT を活用した環境問題の解決」

登録順	タイトル	学校名	指導教員	参加学生
1	SDCs ー100年続くまちづくりー	舞鶴	井上 泰仁	山本 瑞樹 (3年)、平田 爽馬 (4年)、河本 泰尚 (3年)、松崎 伶音 (3年)、高岡 優羽 (3年)
2	Noverflow ー「ボイ捨て」を取り除くー	舞鶴	井上 泰仁	伊勢 巧 (4年)、池田 翔太郎 (4年)、小寺 恭平 (3年)、森本 柊哉 (3年)、半林 駿典 (3年)
3	エコライフスターター	函館	小山 慎哉	岸 真央 (2年)、豊蔵 巧大 (2年)、伊藤 涼太 (2年)、古澤 琉威 (1年)、吉田 海翔 (専攻科1年)
4	Excess Informer ー買い過ぎ防止システムー	小山	大内 翔平	菅原 晃一郎 (3年)、横井 蛭汰 (5年)、藤沼 心吾 (2年)、黒田 若葉 (2年)、小堀 咲陽 (2年)
5	ラビビ ーRubbish navigationー	香川 (詫間)	宮崎 貴大	三好 湊稀 (1年)、藤原 瑞月 (1年)、来田 久一郎 (1年)、高畑 葉響 (1年)、村口 友星 (1年)
6	オトセバ ー音によるゴミ分別システムー	大島商船	重本 昌也	神崎 友希 (5年)、西川 陸 (5年)、小松 咲絵 (4年)、岡本 羽琉夢 (4年)、倉増 風紗 (3年)
7	やしえーど ー効果の可視化で緑のカーテンの普及をー	沖縄	金城 篤史	城間 華 (4年)、兼久 紗嬉 (4年)、鳥袋 伊玄 (4年)、新垣 善也 (4年)、仲村 優佳 (4年)
8	プラたべる	大島商船	重本 昌也	縄田 嵐 (2年)、倉本 悠雅 (2年)、車 優月 (2年)、横道 玲旺 (2年)、宮崎 蘭 (2年)
9	だすばす ー農家と家庭をつなげる堆肥共有プラットフォームー	国際	伊藤 周	江口 太一 (2年)、角 隼 (2年)、五百蔵 蒼天 (1年)、藤原 聖也 (1年)、中澤 琉月 (2年)
10	アースリン ー地球にやさしくなるアプリー	津山	川波 弘道	小津野 智葉 (5年)、山下 珂徳 (3年)、小林 将太 (2年)、河本 明未佳 (1年)
11	Triplean ーインバウンド対応・清掃支援システムー	鳥羽商船	江崎 修央	白川 琥大 (4年)、松葉 勇希 (4年)、北仲 悠人 (3年)、山下 温斗 (3年)、鮎川 颯 (1年)
12	SEA-VIS ー水中ドローンとAIによる調査システムー	沖縄	中平 勝也	山川 響樹 (4年)、西田 皓太 (4年)、當間 一代 (4年)、仲宗根 由弥 (4年)
13	りょーメシ ー持続可能な寮食支援システムー	弓削商船	長尾 和彦	井上 裕太 (5年)、黒河 天晴 (5年)、岡田 右京 (3年)、久保 竜壮 (2年)、福田 煌輝 (1年)
14	LOOK CO2 ー家庭内のCO2排出量を見える化しようー	香川 (高松)	重田 和弘	泉 凜太郎 (5年)、出上 裕大 (5年)、香川 爽真 (5年)、林野 圭将 (5年)、宮下 大空 (5年)
15	クラスdeECOウォッチ ー環境教育支援システムー	宇部	久保田 良輔	中野 晃聖 (4年)、村谷 凌 (5年)、木村 友亮 (2年)
16	DeerHere	香川 (詫間)	宮崎 貴大	岩本 理玖 (3年)、香川 琴音 (3年)、牧野 友輝 (2年)、松本 悠臣 (2年)、平尾 風咲 (2年)
17	グリーンデリバリー ー地域のフードロスに挑む食品フリマー	米子	松本 正己	石原 琉翔 (5年)、野田 柚月 (5年)、松本 颯人 (5年)、松山 凜 (3年)、森澤 結 (3年)
18	地賛地頌 ー詠もうぜ地元!君の言葉でー	仙台 (広瀬)	安藤 敏彦	司東 晃大 (2年)、村木 祐太 (3年)、千葉 翔太 (3年)、伊藤 颯汰 (3年)
19	TerraMeal	鳥羽商船	中井 一文	前田 遥 (4年)、田中 映音 (4年)、岡 駿希 (4年)、糸川 颯人 (1年)、西根 明日翔 (1年)
20	金継ぎVR	熊本 (八代)	小島 俊輔	武藤 淳之助 (4年)、山下 遼太 (5年)、盛高 雄寛 (2年)、脇田 龍之介 (4年)、吉里 彩心 (1年)
21	RescueLink ー自然災害支援のための連絡・情報共有アプリー	福井	小松 貴大	齋藤 巧輝 (4年)、古川 大希 (4年)、宮本 圭 (4年)、山内 壮良 (4年)
22	Wish Connect	津山	川波 弘道	石原 希光 (3年)、田中 詩麻 (5年)、横浦 宏一 (5年)、山本 樹 (2年)、仲田 紗莉那 (1年)
23	育てるエコタウン ー誰でも楽しく環境活動ー	熊本 (八代)	小島 俊輔	岩谷 奏慈郎 (2年)、廣野 達徳 (2年)、緒方 亮太 (5年)、高見 勇斗 (1年)、高山 透也 (5年)
24	まもるん ー役場と町民の災害時連携アプリケーションー	神山	春田 麻里	薦田 葵 (2年)、井上 明 (2年)、梅原 琉 (1年)、下島 夏 (1年)
25	CO2-ntrol ー努力を可視化そして「リアル」へー	呉	藤井 敏則	島田 倅希 (3年)、佐々木 恒宣 (3年)、白井 陽士 (1年)、福島 東吾 (1年)、益子 賢一 (1年)
26	Mealloop	高知	立川 崇之	石内 遥 (3年)、中平 慧 (4年)、前田 恭祐 (3年)、河内 瑛莉子 (3年)
27	EV ー車を用いた環境プログラミング教育ー	福井	小松 貴大	東 凜太郎 (4年)、小川 拓也 (4年)、相馬 健太 (4年)、中西 奏一郎 (4年)、南部 雅斗 (4年)
28	アルCO ー地球を守る、優しいアプリー	奈良	上野 秀剛	崎本 俊治 (4年)、日下 優人 (4年)、池本 拓冬 (4年)、森岡 大和 (3年)、増井 崇 (3年)
29	Bio Scouter	小山	平田 克己	平田 啓介 (3年)、舟迫 颯 (3年)、大嶋 陽菜 (3年)、北條 真弓 (3年)、土田 暁大 (1年)

■課題部門「ICTを活用した環境問題の解決」

登録順	タイトル	学校名	指導教員	参加学生
30	アツマル ーダーウィンになろうー	松江	杉山 耕一郎	澤和 楓音 (2年)、田中 煌珠 (2年)、池田 小夏 (1年)、 長谷川 大晴 (1年)、鎌田 真太郎 (1年)
31	バイオ炭治郎 ー炭素貯蓄編ー	松江	村橋 究理基	山崎 巧実 (2年)、岩成 広樹 (2年)、辻 蓮治 (1年)
32	節電だョ!全員集合 ー節電の実践と情報提供で意識を高めるシステムー	阿南	吉田 晋	木下 陽平 (4年)、松本 琉希 (4年)、廣島 承悟 (4年)、 北山 秀磨 (4年)、天羽 優太 (4年)
33	POOI ーデジタル投票式ゴミ箱ー	阿南	太田 健吾	泉 創太 (3年)、井口 蒼生 (3年)、久保田 昇輔 (3年)、 山上 遥己 (3年)、阿瀬川 祥永 (5年)
34	Foodvisor ーレシートから賞味期限管理ー	群馬	菊地 洋右	生方 来夢 (3年)、飯島 慶士 (4年)、岩下 悠斗 (1年)
35	Clean Clicker ークリック連打で町をキレイにー	福島	植 英規	酒井 海里 (4年)、齊藤 侑貴 (4年)、平子 湧貴 (4年)、 関根 佳孝 (4年)
36	PhotoRou ー環境と人に優しい習慣作りアプリー	神山	春田 麻里	菊井 福音 (2年)、武田 璃香 (2年)、遠近 壮馬 (2年)、 松田 紗亜弥 (2年)、森川 結太 (1年)
37	緑のみちびきシステム ー猟師、林業従事者の新規参入作戦ー	長野	伊藤 祥一	秋山 侑 (3年)、黒崎 陽暉 (3年)、橋爪 黎 (3年)、 小林 歩生 (3年)
38	横取りVスタディ ーVR横取りシステムで地域学習ー	八戸	若狭 尊裕	佐藤 聡太 (3年)、竹内 陸央 (1年)、古瀬 龍司 (1年)、 古田 晃一郎 (5年)、中村 亮太 (5年)
39	デジタル百葉箱 ーリアルタイムな気象データで快適にエコを実現するー	富山 (本郷)	石田 文彦	西尾 優志 (3年)、杉山 正悟 (3年)、宮嶋 陸人 (3年)
40	GarbaGPS	神戸市立	高田 峻介	金子 尚暁 (3年)、荒瀬 遼統 (3年)、林 克己 (3年)、 福本 陽希 (2年)、榮 遼陽 (2年)
41	EcoMeow	タイ高専	Krittanik Srithanasarn	Nuntanut Poonpayap、Thad Choyrum
42	Share CARE	タイ高専	Teerapong Orachon	Natthanon Weerapong、Punnawith Sutsukon
43	Smart faucet	モンゴル コーセン	Khurelbat Khishigt	Ganjuurdalai Enkh-Uchral、Ariunbold Zorigtsaikhan

■自由部門「自由なテーマで独創的な作品」

登録順	タイトル	学校名	指導教員	参加学生
1	uni	香川 (詫間)	金澤 啓三	山田 美羽 (5年)、天竺 寛貴 (5年)、横井 優樹 (5年)、小原 崇靖 (5年)、菅 勇磨 (5年)
2	KNOW農	舞鶴	片山 英昭	南 歩汰 (3年)、坂田 雄哉 (3年)、辻本 健斗 (2年)、平野 茂吉 (3年)、長澤 結司 (3年)
3	ぶらなびpig ー貯金×ぬりえによる目標達成ツールー	舞鶴	森 健太郎	片岡 竜希 (3年)、六田 薫 (3年)、山田 和佳 (3年)、塚本 樹 (2年)、若山 拳斗 (3年)
4	Luciole ー暮らしの集合知ー	有明	松野 良信	足立 竣弥 (5年)、井上 詩音 (5年)、緒方 陽都 (5年)、松本 光聖 (5年)
5	SNAPEDAL	石川	三吉 建尊	中西 一稀 (3年)、坂口 恵美 (3年)、石川 響己 (3年)、石地 樹 (3年)、岩本 裕之介 (3年)
6	Smart Shelf ー未来に近づく物品整理ー	大島商船	北風 裕教	嶋 諒大 (専攻科1年)、岩政 綾馬 (専攻科1年)、成田 健志郎 (5年)、神所 恭佑 (5年)、蛭子 莉愛 (4年)
7	Splush ーShibuki + Hamonー	香川 (詫間)	金澤 啓三	金森 隆太郎 (3年)、豊島 圭吾 (3年)、馬場 裕一 (3年)、濱田 健司 (3年)、HARNKUMNEDPONG PATTARACHANON (3年)
8	まふいん ーあなたのおじさん育てませんかー	有明	松野 良信	片野 凱介 (4年)、古賀 雅寛 (2年)、猿渡 優衣 (5年)、古川 蒼太郎 (5年)、西山 相奈 (2年)
9	SIGNALOID ーあなたのプレゼンをスマートにー	呉	藤井 敏則	櫻井 健太 (4年)、佐々木 一護 (4年)、宮川 聖章 (5年)、土井 蒼太 (5年)、川田 悠平 (4年)
10	Surfinity! ーサーフィン大会の総合支援システムー	鳥羽商船	江崎 修央	北仲 一登 (4年)、鈴木 ライヤ (4年)、長瀬 真生 (4年)、中森 立樹 (3年)、塚本 真己也 (3年)
11	ブライズシューティング	明石	土田 隼之	井上 元葵 (3年)、尾上 智哉 (3年)、山中 一真 (3年)
12	星空生活	小山	平田 克己	平野 七々紗 (2年)、朝稲 太一 (2年)、荒井 陽向 (2年)、杉山 彰洋 (2年)、渡邊 優杜 (2年)
13	きゃわきゃわきゃめら	福井	高久 有一	駒井 百花 (4年)、齊藤 朱 (4年)、杉本 心音 (4年)、米沢 日那 (4年)
14	エモアンプ	福井	高久 有一	片岡 志菜 (4年)、岡本 篤倉 (4年)、高橋 楓哉 (4年)、大瀬 怜牟音 (4年)、渡邊 展匠 (4年)
15	この虫な〜んだ? ーハンドサインで虫がわかる!ー	徳山	力 規見	田中 響大 (3年)、池永 珠希 (3年)、平塚 智紀 (2年)、伊藤 結彩 (2年)
16	Pixflower ーメッセージに彩りと遊び心をー	松江	渡部 徹	北島 壮一郎 (3年)、池田 翔哉 (3年)、濱田 晏慈 (3年)、洗川 見蔵 (1年)、成相 拓真 (1年)
17	アクトール ARで気軽に演技上達ー	熊本 (八代)	小島 俊輔	吉田 日向 (2年)、嶋中 海人 (5年)、稲富 将大 (4年)、土井 恋海 (2年)、廣瀬 優 (2年)
18	マブノウン ーランダム目的地マップー	香川 (高松)	川上 裕介	大住 陽雲 (5年)、鈴木 智大 (5年)、小橋 勇太 (3年)、村上 功一 (3年)
19	AIデンティファイ ー安全で楽しいSNSライフをあなたにー	八戸	釜谷 博行	白石 光 (3年)、館向 大輔 (3年)、三浦 士 (1年)、熊野 りお (3年)、杉本 慎之介 (3年)
20	HouCycle ーVRand new cycling lifeー	茨城	滝沢 陽三	安齋 朴音 (4年)、大槻 育生 (4年)、神永 耀宏 (4年)
21	STaiLE ーあなただけのファッションストーリーー	長岡	竹部 啓輔	大関 啓太 (5年)、岩村 天清 (5年)、布川 康太 (5年)、八幡 歩軌 (4年)
22	! ーthe world of billiardsー	弓削商船	長尾 和彦	石橋 治樹 (3年)、五所 杏太 (3年)、小田 絃平 (2年)、木村 斗哉 (2年)、濱田 捷聖 (1年)
23	ふおとらてじー ーあ!いい写真撮れた!ー	米子	松本 正己	下山 虹 (5年)、谷森 充啓 (5年)、小椋 雄太 (3年)、田上 大翔 (3年)、宮代 保榮 (3年)
24	につきー ー毎日続けられる日記アプリー	仙台 (広瀬)	力武 克彰	渡邊 天 (4年)、山本 瑞貴 (2年)、増子 巧真 (1年)、菅原 伊織 (1年)
25	理由を投稿するSNS「教えてri-U」 ー思考力、表現力を育てるー	久留米	田中 諒	山福 乃愛 (2年)、山福 桜綺 (専攻科1年)、栗原 陸 (専攻科1年)、浦塚 武士 (専攻科1年)、松岡 海誓 (2年)
26	PeerReach ー仲間との協力でトップへ到達ー	沖縄	金城 篤史	椎名 咲太郎 (5年)、松田 恋椰 (5年)、前川 優太 (5年)
27	Nahlun ーボクセルで見える!河川水位と地域の安全ー	群馬	木村 清和	荒井 日菜子 (5年)、菊池 静琉 (5年)、後藤 順太 (5年)
28	こねくと (仮) ー災害用スマホネットワークー	群馬	渡邊 俊哉	松本 祐真 (4年)、熊 琵琶 (4年)、藤田 恭輔 (4年)
29	SHOESAY	鳥羽商船	中井 一文	小久保 七緒 (4年)、高橋 大和 (4年)、三野 琳久 (3年)、小林 叶姫 (3年)、池田 瑛人 (1年)
30	Internet Survival ー楽しく学ぶ情報モラルー	高知	立川 崇之	中野 陽葵 (4年)、宮崎 勇悟 (4年)、山本 拓実 (4年)、土居 和楓 (4年)
31	みちすじ ーみんなで描く点字ブロックの道ー	都城	丸田 要	下村 翼 (5年)、園師 颯汰 (4年)、須志田 昊 (4年)、松岡 萌衣 (3年)、小林 由希奈 (2年)
32	Don't Asleep	小山	大内 翔平	大竹 響己 (4年)、宮原 友哉 (4年)、藤澤 颯介 (3年)、池田 陽菜太 (2年)、河口 夏音 (2年)
33	ふたりんごと ーどうやってこの感じ伝えようかなー	豊田	都築 啓太	野崎 春太郎 (5年)、志比田 悠翔 (3年)、深谷 春駈 (2年)、小松 杏菜 (2年)、加藤 遥也 (2年)

■自由部門「自由なテーマで独創的な作品」

登録順	タイトル	学校名	指導教員	参加学生
34	idefine －アイデアを定義する－	高知	千秋 元	岡田 真周 (5年)、松平 真之介 (5年)、伊藤 達哉 (5年)、三谷 重友翔 (5年)
35	フィッシュフレンズ －魚をより近い存在に－	香川 (高松)	村上 幸一	三宅 立晃 (3年)、門田 続海 (3年)、金地 琳太郎 (専攻科2年)
36	次世代防災・災害アプリ“Crisis”	旭川	森川 一	辻本 純平 (2年)、越後 奨 (2年)、川守田 優斗 (2年)、田中 伸 (2年)、川上 晃弥 (2年)
37	Talk +More －心地よいコミュニケーションの場－	新居浜	占部 弘治	白石 稜真 (2年)、林 拓士 (2年)、兵頭 勇哉 (2年)
38	つくも遊歩 －つくも神と巡るこころの故郷－	東京	山下 晃弘	柴野 真惇 (2年)、石井 岳光 (2年)、田中 璃久 (2年)、篠倉 詠音 (2年)、西村 朱葉 (2年)
39	RoomAId －掃除×ゲームで生活を改善！－	国際	藤澤 武	泉屋 匠吾 (4年)、山崎 史依 (5年)、バシユロ 琉海 (5年)
40	Secure Ryman －情報セキュリティ対策推進サービス－	木更津	米村 恵一	八木 航樹 (専攻科1年)、富澤 太一 (専攻科1年)、鈴木 聡一郎 (専攻科1年)
41	なまつ talk －テキストでだって方言を－	富山 (射水)	山口 晃史	高山 七帆 (3年)、田中 千尋 (4年)、Ariq Athallah Budi Ramdhany (3年)、源明 玲煌 (1年)、Naksub Sorachach (2年)
42	どみとる －学生寮共用設備利用状態管理システム－	石川	越野 亮	福井 悠生 (4年)、村上 亜土沙 (4年)、置田 裕也 (4年)、毛利 俊太 (4年)、毛利 謙太 (4年)
43	ニムモ	沼津	鈴木 康人	荻野 壮 (2年)、北村 侑也 (2年)、遠藤 大祐 (5年)、横山 由宇 (4年)、三村 治弘 (1年)
44	SSNS －aquarium of youth－	阿南	岡本 浩行	正木 彩月 (3年)、赤尾 奈葉美 (3年)、奈良 結依 (3年)、原田 柚月 (2年)、廣瀬 七海 (2年)
45	CHaser Arena	東京	松林 勝志	中川 治香 (4年)、澁谷 知希 (2年)、今井 公河 (1年)、福本 圭悟 (1年)、島田 航平 (1年)
46	Mathtoro －数学の質問・解決を効率よく－	久留米	田中 諒	田村 司字 (4年)、井上 棕太 (4年)、日永田 幸多 (1年)
47	覚え歌つくくん	仙台 (広瀬)	穂坂 紀子	八重樫 守 (3年)、荒俣 誠 (2年)、佐藤 佑侗 (2年)、中島 悠登 (2年)、八木沼 佑斗 (2年)
48	ARPON －スマホで簡単AR－	佐世保	嶋田 英樹	田畑 賢悠 (3年)、廣重 伊織 (3年)、中島 寛隆 (3年)、廣瀬 温音 (3年)
49	SPORTSDAY －球技大会管理プラットフォーム－	富山 (本郷)	石田 文彦	山本 哲也 (5年)、中村 祐輔 (5年)、池畠 太勇 (2年)
50	Tourwith －パンダには旅をさせよ－	佐世保	手島 裕詞	吉田 健志 (4年)、前田 一歩 (4年)、美甘 悠吾 (4年)、山川 瑛児 (4年)、松本 大心 (4年)
51	DiaLog －挫折しない対話型日記－	神山	春田 麻里	金谷 高興 (2年)、メリット キア (2年)、福田 晴 (2年)、山口 空 (2年)、宮野 柊太 (2年)
52	Regina －自動学習精算機－	広島商船	岩切 裕哉	谷本 佑樹 (5年)、原 裕次郎 (5年)、祐本 葉々美 (5年)、室岡 陸 (3年)、祐本 浩輝 (3年)
53	SimPhy	神山	春田 麻里	丹那 伊織 (2年)、高橋 空知 (1年)、鈴木 楓可 (1年)
54	ProfessionalLink －料飲業界の職業紹介をDX化！－	神戸市立	高田 峻介	高橋 颯 (4年)、鷹尾 心優 (4年)、長村 圭祐 (4年)、岡 俊博 (4年)、田村 優衣 (4年)
55	MiReHSeL －複合現実を用いたハードウェアセキュリティ学習－	長野	伊藤 祥一	赤澤 楓 (3年)、小島 拓也 (4年)、佐藤 悠太 (専攻科1年)
56	cleanAR －部屋の整理整頓促進システム－	熊本 (熊本)	藤井 慶	砂川 みなみ (5年)、船津 ひなた (5年)、水野 李咲 (5年)、吉田 翔太郎 (5年)
57	シオサビン －塩害を検知、予報し 被害を未然に防ぐ塩害対策アプリ－	沖縄	金城 篤史	川満 秀太郎 (3年)、屋宜 元晃紗 (3年)、吉田 信仁 (4年)、喜友名 朝舜 (1年)、宮里 壮汰 (4年)
58	30 Days Harvest	キングモンクット 工科大	Sirion Vittayakorn	Pakin Kaophuthai, Peravit Kritchakaj
59	QuizRush	新モンゴル 高専	Shur-Erdene Buyannemekh	Bilguuntushig Amarsaikhan, Tuguldur Batsaikhan, Choigun Kherlen, Bayarsaikhan Baatarkhuu, Buyandelger Myagmarjav
60	AfterDay Horizon	キングモンクット 工科大	Supannada Chotipant	Parin Dechakorn, Pairat Chuenchom
61	Neko Torae	シンガポール ポリテクニク	WONG Kwee Yin	SENG KAY KANG JERICK, LEROY HONG JAE-YANG
62	Legacy of Genghis Khan －The Mongol Empire－	新モンゴル 高専	Batchuluun Batnasan	Munkh-Orgil Myagmardorj, Mungunsarnai Kherlen, Purevdorj Enkhjur, Ankhlan Chingis-Od, Sukhbold Altansukh
63	Sign-lent	タイ高専	Pirapat Tangsuknirundorn	Pradubfah Thanataweerat, Worawalan Woradee
64	Dancing shrimps	タイ高専	Tanapon Keatsamarn	Nutchakramol Piamsaard, Punyavee Nanthanawat
65	Smart faucet	モンゴル ゴーセン	Khurelbat Khishigt	Ganjuurdalai Enkh-Uchral, Ariunbold Zorigtsaikhan

■競技部門「シン・よみがえれ世界遺産」

登録順	チーム名	学校名	指導教員	参加学生
1	random.randint	舞鶴	森 健太郎	矢野 凌太郎 (3年)、兼房 和輝 (3年)、藤井 風貴 (4年)
2	巨肩の上に立つ	八戸	細川 靖	長谷川 隼也 (3年)、佐藤 僚亮 (3年)、仲山 陽大 (3年)
3	天体観測	大島商船	北風 裕教	岩政 佑樹 (2年)、村川 諒太 (2年)、樋口 聖人 (2年)
4	U18	津山	川波 弘道	撰 咲汰 (3年)、大西 美碧 (3年)、杉本 昊大 (2年)
5	デュートリチウム	秋田	竹下 大樹	佐々木 隆稀 (2年)、佐藤 虹太 (2年)
6	情研.yata	奈良	内田 眞司	瀧口 大地 (2年)、吉岡 竣平 (2年)、新主 悠人 (2年)
7	Unfixed	長岡	竹部 啓輔	千葉 幹太 (1年)、横山 佳大 (1年)、津輕 健太 (1年)
8	ゲーミング文化財	東京	松崎 頼人	三浦 理稀 (5年)、前沢 完齊 (3年)、豆田 匠音 (2年)
9	3.66i	石川	越野 亮	松元 颯矢 (4年)、中川 耀登 (4年)、酒井 琉太 (3年)
10	Open The Door	一関	村上 力	後藤 柊矢 (5年)、岩淵 拓 (5年)、及川 博幹 (4年)
11	疾風迅雷	鳥羽商船	中古賀 理	横山 朋輝 (3年)、内田 英都 (3年)、橋爪 天聖 (3年)
12	いなかんぱ	福島	小泉 康一	阿部 港 (5年)、常盤 海玖 (5年)、柳沼 孝太 (5年)
13	ゴリゴリズム2.0.0	熊本(熊本)	藤井 慶	永松 日月 (3年)、牛嶋 叶志朗 (4年)、富永 颯真 (4年)
14	CeroRe:Master	群馬	渡邊 俊哉	山縣 蒼人 (3年)、豊巻 煌 (4年)
15	例によって締切寸前で殺気立つ高専生	香川(詫間)	宮崎 貴大	大西 俊輔 (4年)、三宅 巧 (1年)、野田 八雲 (1年)
16	while True: print ("HelloProcon")	熊本(八代)	小島 俊輔	石堂 優人 (2年)、嶋中 海人 (5年)、高松 龍希 (2年)
17	三者三様	函館	倉山 めぐみ	水島 悠翔 (4年)、上野 晃 (4年)、中野 名惺 (4年)
18	かみやまひらがなこうせん >>「くらえッ！タマネギ！」	神山	春田 麻里	伊藤 楽大 (2年)、相木 絆煌 (2年)、佐藤 誠 (1年)
19	ボボボード・ボードド	仙台(広瀬)	藤原 和彦	工藤 聖晴 (3年)、岸 和希 (3年)、岡田 達樹 (3年)
20	DRL ーだって乱数でlearningデータ集めたから	大阪公大	窪田 哲也	田村 唯 (5年)、竹内 珠佑 (3年)、辰己 由空 (2年)
21	鹿の手も借りたい	弓削商船	長尾 和彦	岡田 慈英 (5年)、CHALEUNKIT THANASINE (5年)、松田 柚鈴 (3年)
22	高専祭には目もくれず、彼らはプロコンに来た	釧路	天元 宏	小関 将人 (3年)、櫻井 文太 (3年)、松本 浩貴 (2年)
23	チーム名が決まらなかったのでお好きなようにお呼びください	都城	丸田 要	園田 馳弥 (3年)、加治屋 翔大 (3年)、東園 獅之丸 (2年)
24	クッキーカッターズ	香川(高松)	柿元 健	大鹿 友混 (4年)、大石 悠馬 (4年)、伊戸 寛哲 (3年)
25	夜行バスには乗らない	仙台(名取)	佐藤 隆	鈴木 佑 (4年)、佐藤 至 (5年)、黒澤 徹真 (2年)
26	npm install -D 型抜き	近畿大	坂東 将光	楓 悠斗 (5年)、西半 涼賀 (5年)、山本 隆翔 (4年)
27	ラウンドロビン	沼津	鈴木 康人	藤田 康佑 (2年)、山本 環太 (4年)
28	編入数学徹底研究徹底研究部	沖縄	金城 篤史	平良 昂也 (専攻科2年)、田中 柊 (5年)、知念 涼太郎 (5年)
29	神託機械? ～逆襲のif文～	大分	徳尾 健司	引田 鴻志 (3年)、染矢 雅尊 (2年)、村井 春陽 (2年)
30	限界集落	豊田	三浦 哲平	宇井 颯汰 (5年)、鈴木 健太 (5年)、吉野 弘晟 (1年)
31	ヤメテアゲテネ	和歌山	森 徹	岡崎 惇平 (4年)、浅井 正希 (4年)、坂口 勢吹 (4年)
32	創立50周年記念	徳山	力 規晃	棟近 祐希 (5年)、児玉 隆之介 (5年)、門原 裕大 (3年)
33	忍耐力を付けたかった	新居浜	占部 弘治	中家 海翔 (3年)、中平 智也 (3年)、西元 銀弥 (3年)
34	-4℃高専	高知	立川 崇之	福岡 伊織 (4年)、宗石 響 (2年)
35	Static Programming	鈴鹿	青山 俊弘	白髭 樹 (4年)、中村 春琉 (3年)、市川 敬士 (3年)

■競技部門「シン・よみがえれ世界遺産」

登録順	チーム名	学校名	指導教員	参加学生
36	人力全探索	小山	小林 康浩	森 悠貴 (2年)、鈴木 真 (2年)、藤澤 怜央 (2年)
37	回鍋肉	松江	橋本 剛	三島 知樹 (専攻科2年)、田中 庵 (3年)、長谷川 友音 (4年)
38	深夜テンション	岐阜	菊 雅美	渡邊 雅晃 (4年)、福富 陽生 (4年)、北川 大雅 (2年)
39	if (優勝！) 番狂わせ else また来年！	鹿児島	武田 和大	柿木 瑞貴 (3年)、花田 純正 (2年)、亀沢 優月 (2年)
40	Cたすたす	サレジオ	宇都木 修一	井上 将秀 (4年)、境 結菜 (3年)、小林 千尋 (4年)
41	聖徳.mch	苫小牧	山本 椋太	藤田 響 (4年)、治田 風冬 (3年)、野澤 和真 (2年)
42	プロコンに傷心	久留米	古賀 裕章	平井 零大 (3年)、古川 樹 (3年)、日永田 幸多 (1年)
43	410 Gone	宇部	田辺 誠	山本 健一朗 (3年)、池本 悠生 (3年)、井藤 啓晴 (1年)
44	あげししゃも	旭川	森川 一	長谷川 碧 (3年)、長谷川 輝 (3年)、晒谷 悠斗 (3年)
45	1年生教育研修ツアー	富山 (射水)	山口 晃史	土橋 晴人 (4年)、岡本 倫規 (1年)、廣田 響記 (1年)
46	少納言大豆	阿南	平山 基	米積 悠翔 (2年)、Naphat Soonthornkorani (2年)、寺尾 惟吹 (2年)
47	演算クラブ	福井	小松 貴大	向井 海瑠 (4年)、保川 晃 (4年)、ナツアグドルジ ブヤンヒシグ (4年)
48	無限即席冷麺	米子	徳光 政弘	田中 拓弥 (3年)、池口 寛大 (3年)
49	文化財修復安全確保支援士	都立 (品川)	佐藤 喬	石井 一肇 (5年)、平良 文哉 (3年)、田中 智之 (3年)
50	ホタルイカレジェンズ	富山 (本郷)	石田 文彦	安念 翼 (4年)、西村 旭生 (4年)、早水 拓海 (4年)
51	永久盤上領域 アガロン・ク・レ	呉	藤井 敏則	江島 朋来 (3年)、吉岡 雅希侶 (3年)、城 ジュニアスプラタマ (5年)
52	雄峰寮焼きなまし部	木更津	米村 恵一	内藤 正浩 (4年)、越智 優真 (3年)、星 凌誠 (3年)
53	国の宝が文化祭を捨ててでもやりたいこと	長野	藤田 悠	野村 岳歩 (3年)、小林 俊太 (3年)、赤澤 楓 (3年)
54	最悪鹿せんべいだけ食べて帰る	佐世保	嶋田 英樹	豊田 虎 (4年)、矢吹 隼人 (4年)、今村 泰哉 (4年)
55	バナナより犬派	広島商船	岩切 裕哉	東方田 匠真 (2年)、河原田 良明 (2年)、正宗 一康 (3年)
56	よねびーなっつ	有明	森山 英明	内田 陽太 (4年)、栗山 大樹 (4年)、米村 慶太 (4年)
57	"; DROP TABLE teams; --"	都立 (荒川)	鈴木 弘	高野 陽大 (5年)、豊田 アディール (5年)、三澤 進生 (5年)
58	寿司が好きなコンピュータ、何か知ってる？「データ巻き」ってね	神戸市立	朝倉 義裕	杉本 遼羽源 (3年)、明 幸音 (2年)、北尾 匠望 (2年)
59	VTC	香港VTC	Choy Shu Sang	Chau Ka Tai, Lam Long Hei Jason, NG,Tsz Wun
60	PAM	ハノイ工業大	Ma Thi Chau	Nguyen Huy Hieu, Doan Duc Kien
61	XStar	新モンゴル高専	Soronzonbold Nachin	Tselmeg Tamir, Dulguun Zolzaya, Kyungchan Kim
62	Ludwig van Beethoven	タイ高専	Saunghninpwint Oo	Phubadee Baimiden, Pranitan Chantawong
63	Moko	モンゴルコーセン	Jambaldorj Oyundari	Tumenjargal Bumbayar, Batbayar Tsogtbayar
64	Coremind	モンゴル科技大	Khuder Altangerel	Gundalai Luvsanchultem, Tsatsral Sainbolor

■競技部門 司会・進行担当紹介

高田 伸一

石川県在住。プロのナレーター／朗読手／学校演劇指導員。第13回石川大会以来、現在に至るまで競技部門の司会・進行を担当。「はあじめっ！」「ここまでっ！」のかけ声など、いまや高専プロコン競技部門の名物／風物詩になっている。

■競技部門 BGM 担当紹介

鈴木 亨

岡山県在住。現在は保育士として勤務する。高専プロコン第21回高知大会（2010年）から競技部門の試合進行中に流されるBGMの作曲を担当している。

競技部門のルール

今年の競技部門では、はじめに複数のピースから構成されるボード（修復前の文化財）と、複数の抜き型（修復道具）が用意されます。抜き型をボードに適用すると抜き型と一致したピースが抜け、抜けたピースを上下左右いずれかの方向から詰めて補います。うまく抜き型を適用して、ボードを最終盤面（修復後の文化財）にします。できるだけ少ない手数かつ早く修復前の文化財を修復後の文化財に「修復」したチームが勝利する競技です。

ボードと抜き型

ボードの構成要素をピースと呼び、ピースを矩形に並べてボードを構成します。ボードサイズ（ピースの数）は縦、横、それぞれ最大で 256 個、最小で 32 個、縦と横の構成数は同じとは限りません。ボードサイズは問題によって異なります。

ピースには 0 から 3 までの整数値が割り振られます。0 から 3 までの各値はいずれも全体のピースの 10% 以上に割り振られます。

抜き型の構成要素をセルと呼び、セルを矩形に並べて抜き型を構成します。抜き型のサイズ（セル数）は縦、横、それぞれ最大で 256 個、最小で 1 個、縦と横の構成数は同じとは限りません。抜き型のサイズは抜き型によって異なります。

セルには 0 または 1 の値が割り振られます。すべてのセルの値が 0 の抜き型はありませんが、0 と 1 の割合に制約はありません。

型抜き

ボードのある領域に対して抜き型を適用することを「型抜き」といい、1 回の型抜きを「手数」としてカウントします。同じ抜き型を複数回型抜きすることができます。型抜きではボードに対して以下の操作を順に行います。

- ① 抜き型の 1 の値のセルに対応するボードのピースをボードから抜く
- ② ボードに残ったピースを指定方向に寄せる
- ③ 空いたスペースに①で抜いたピースを元の並びで戻す

型抜きはボードの任意の位置に対して適用可能で、抜き型の一部がボードに重なっていればボードからはみ出して型抜きすることも可能です。

はみ出した部分の抜き型は型抜きになにも作用しません。

型抜き位置は抜き型の左上セルを適用するボードの座標で表します。ボードから上や左にはみ出すためボード外の位置を指定しなければならないときは、マイナスの座標で表します。

移動方向は上下左右のいずれかを指定できます。

順位決定方法

一致度、手数、回答時間により、以下の優先順位で勝敗を決定します。

- (1) 最終状態との不一致ピースが少ないチームが勝利します。
- (2) 手数が少ないチームが勝利します。
- (3) 最終回答時間が早いチームが勝利します。
- (4) サイコロなどで勝敗を決めるか引き分けとします。

競技部門の組合せ

1 回戦

ブース	第 1 試合	第 2 試合	第 3 試合	第 4 試合	第 5 試合	第 6 試合	第 7 試合
1	宇部	東京	サレジオ	都立(品川)	群馬	阿南	熊本(熊本)
2	舞鶴	奈良	沖縄	苫小牧	釧路	津山	一関
3	長野	都城	仙台(名取)	富山(本郷)	岐阜	弓削商船	鈴鹿
4	熊本(八代)	富山(射水)	小山	神山	高知	都立(荒川)	和歌山
5	広島商船	松江	福島	神戸市立	徳山	有明	久留米
6	新居浜	香川(詫間)	佐世保	大分	大島商船	豊田	沼津
7	米子	長岡	鹿児島	大阪公大	鳥羽商船	石川	八戸
8	福井	香川(高松)	木更津	近畿大	仙台(広瀬)	秋田	旭川
9	函館	呉	新モンゴル高専	モンゴルコーセン	香港VTC	タイ高専	ハノイ工業大

※各試合上位4チームが準決勝へ進出する。5位以下は敗者復活戦へ。

敗者復活戦

ブース	第 1 試合	第 2 試合	第 3 試合	第 4 試合
1	1-1-5	1-2-5	1-3-5	1-4-5
2	1-5-5	1-6-5	1-7-5	1-1-6
3	1-2-6	1-3-6	1-4-6	1-5-6
4	1-6-6	1-7-6	1-1-7	1-2-7
5	1-3-7	1-4-7	1-5-7	1-6-7
6	1-7-7	1-1-8	1-2-8	1-3-8
7	1-4-8	1-5-8	1-6-8	1-7-8
8	1-1-9	1-2-9	1-3-9	1-4-9
9	1-5-9	1-6-9	1-7-9	

※各試合上位2チームが準決勝へ進出する。

準決勝

ブース	第 1 試合	第 2 試合	第 3 試合	第 4 試合
1	1-1-1	1-2-1	1-3-1	1-4-1
2	1-5-1	1-6-1	1-7-1	1-1-2
3	1-2-2	1-3-2	1-4-2	1-5-2
4	1-6-2	1-7-2	1-1-3	1-2-3
5	1-3-3	1-4-3	1-5-3	1-6-3
6	1-7-3	1-1-4	1-2-4	1-3-4
7	1-4-4	1-5-4	1-6-4	1-7-4
8	C-1-1	C-2-1	C-3-1	C-4-1
9	C-4-2	C-3-2	C-2-2	C-1-2

※各試合上位2チームが決勝へ進出する。

決勝戦

ブース	第 1 試合
1	S-1-1
2	S-2-1
3	S-3-1
4	S-4-1
5	S-1-2
6	S-2-2
7	S-3-2
8	S-4-2

※ X-Y-Zは X 回戦-第 Y 試合-第 Z 位を表します。

※ ただし、X 部分のCは敗者復活戦、Sは準決勝を表します。

※ 全国高等専門学校プログラミングコンテストの順位については、海外チームを除きます。

・提出された原稿をそのまま印刷しています。

1

random.randint

舞鶴

矢野 凌太郎 (3 年) 兼房 和輝 (3 年)
藤井 風貴 (4 年) 森 健太郎 (教員)

1. 概要

各列と各行をそれぞれ最終盤面と比較することで算出した各ピースの個数の一致度に重みを置き、ボードの操作がより効果的に働くものを探索することを最終盤面にたどり着くための基本的な方針とする。不一致ピース・手数を少なくするためにはボードを操作することによって生じる無駄を減らすことを考え、そのうえで基本的な方針に沿う操作を実行する。

2. 手数を減らすために

今回の競技テーマの特性上、型抜き後の寄せる方向によって既にそろっているピースがずれてしまい無駄な手数が生まれる可能性が存在する。そのため、あきらかな悪手については図 1 に示すような事前に制作したシミュレータで把握しておき計算から省くこととする。

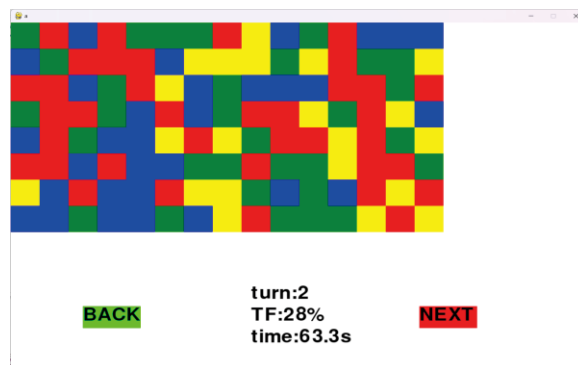


図 1 シミュレータの例

これは計算時間を短縮させ、非効率的な手が選択される可能性をなくすためである。

3. 開発環境

OS:Windows10, 11

言語:C++, Python

エディタ:Visual Studio Code

2

巨肩人の上に立つ

八戸

長谷川 隼也 (3 年) 佐藤 僚亮 (3 年)
仲山 陽大 (3 年) 細川 靖 (教員)

1. 概要

盤面に対しての評価値を持つことが難しく、型抜きの順番によって状態が変化するので、ルールベースの貪欲法を改良したアルゴリズムを採用した。

2. アルゴリズム

2.1 盤面について

盤面や型が最大で 256×256 と非常に大きいため、計算量を抑えて実装する必要がある。使用用途に合わせてデータを削減した。

2.2 主要なアルゴリズム

一行、または一列ずつ貪欲にそろえていくことで、現実的な計算量で揃えることが可能である。揃えるための補助として、盤面において必要な数字を偏らせ、手数を下げる工夫を行った。

2.3 貪欲法の改善

制限時間を有効活用するために、ビームサーチを用いて、多様性の確保や、手数の最小化を試みた。

3. ビジュアライザ

問題の作成や、型抜きの操作を実際に見ることができるビジュアライザを作成した。

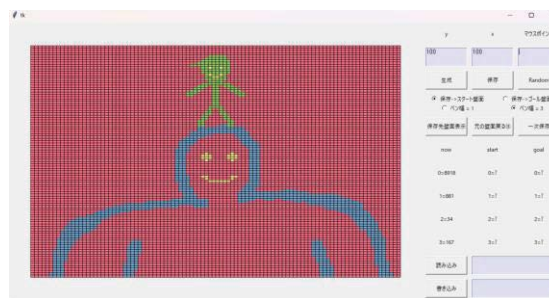


図 2 製作中のビジュアライザ

4. 開発環境

言語 C/C++ ライブラリ OpenMP

3 天体観測

大島商船

岩政 佑樹（2年） 村川 諒太（2年）
樋口 聖人（2年） 北風 裕教（教員）

1. はじめに

本プログラムでは、ボードの大きさが 32×32 から 256×256 まで幅広く設定されており、計算回数が指数的に増加するため、ボードサイズによっては完全な探索が難しい場合がある。そこで、ボードのサイズによって分析アルゴリズムを切り替えることで、問題を解決する。

1. 小規模ボード：最短経路問題を解く
2. 大規模ボード：近似解を用いた探索手法を解く

2. 探索アルゴリズムの選択

2.1 小規模ボードの場合：

小規模であれば、最適解を見つけることが保証されると考え、使用できるメモリの範囲内で A*アルゴリズムを使用する。目的の値と誤差の値が最小になるように評価関数を設定する。

2.1 大規模ボードの場合：

計算資源が限られる状況で現実的な解を得るために、Beam Search を使用して近似解を求める。探索幅と深さの最適な割合を分析し、ボードを分割して探索時間を短縮する。また、2台のパソコンを使用し、縦横の2パターンで探索して最適な結果を算出し、結果を送信する。

2.2 評価関数

現在のボードと目標ボードを比較したときの一致したピースの数を評価値として使用する。

3. 開発環境

言語：C++

動作環境：Windows11

開発環境：Visual Studio 2022

使用フレームワーク：Open Siv 3D

4 U18

津山

撰 咲汰（3年） 大西 美碧（3年）
杉本 昊大（2年） 川波 弘道（教員）

1.はじめに

今回の競技部門は、コンピュータによるパズルゲームである。しかし、今回は複数の定型抜き型と一般抜き型があるため、これらをうまく使うことが手数を減らすうえで重要になる。

2.アルゴリズム

今回はボードのサイズ、抜き型の数・サイズが、試合開始前にわかるため、ボードの盤面に応じた抜き型を使用するアルゴリズムを設計した。同時に非常に多くのピースを揃えることは難しいため、ボードの端からピースを揃えるように型抜きの手を決定する。そろえる必要のあるピースが少なくなると全探索を行い、さらに、手数を減らすことのできる手を決定する。

3.GUI

ボードの状況や型抜き品の状況などを表示することにより、プログラムが正常に動作しているかを確認できるようにして不測の事態に対応できる工夫を行う。



図 1.開発中の GUI

4.開発環境

【OS】 Windows11、Windows10 pro

【Language】 C++

【Frame Work】 Siv3D

【IDE】 Visual Studio

5 デュートリチウム

秋田

佐々木 隆稀 (2年) 佐藤 虹太 (2年)
竹下 大樹 (教員)

1. アルゴリズム

ボードの四端のいずれかから、最終状態に一致するよう、順にピースを移動する。移動するピースは最短の移動量となるよう、適切な種類のピースのうち、移動完了後の位置から最も近い位置にあるピースを選択して移動する。移動の手数を少なくするため、できるだけ移動量が多くなる抜き型を選択して移動させる。

初期配置において一致するピースを移動の対象としないアルゴリズムの実装も検討する。この場合、移動の経路はA*探索を用いる。A*探索に迂回によって移動量が大きくなる場合や、一致位置への移動ができないピースについては、既に一致しているピースも移動可として最短経路で移動させる。

2. GUI

図1は開発中のGUIである。開始状態と最終状態のボ

ード及び一般抜き型の表示や各種パラメータを表示できる。

3. 開発環境

使用言語 : Python, C++

開発環境 : IDLE, Visual Studio

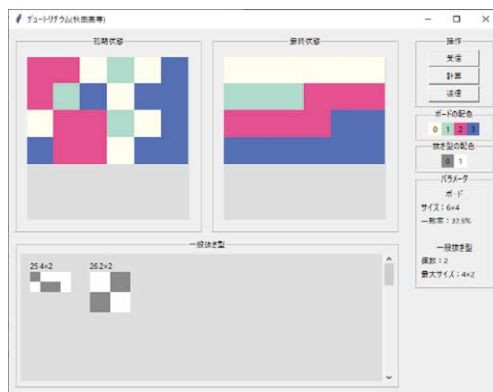


図1 開発中のGUI

6 情研.yata

奈良

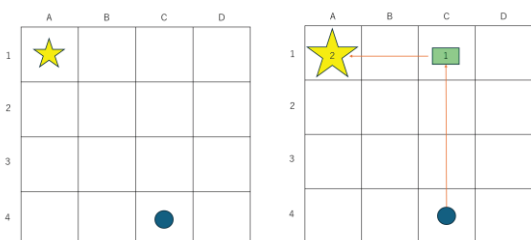
瀧口 大地 (2年) 吉岡 竣平 (2年)
新主 悠人 (2年) 内田 真司 (教員)

1. 概要

今回我々は、初めに定型抜き型を用いて回答を作成し、一般抜き型を用いて回答を省略するという手法を採用することにした。

2. アルゴリズム

対象のマスに数字を目的のマスにもっていくとき、X軸とY軸でそれぞれ動く幅を求めてそれを 2^n のみで表す。例えば、図の丸から星へ数字を移動させるには、4C→1C 1C→1Aの2手が最速である。



3. 不一致ピース・手数を少なくする方法

定型抜き型数個と同じ動きをする一般抜き型を探し、作成した回答の中の定型抜き型数個と一般抜き型を置き換えていくことで手数を少なくする。図で例えると、4Cから1Aまで数字を移動させることができる一般抜き型があれば、最速手数を2手から1手へと縮めることができる。また、同じ動きをする一般抜き型の検索には焼きなまし法を用いる。

4. 開発環境

使用言語 : C++ Python

開発ツール : Visual Studio Code

OS : Windows11

1.概要

Python を用いて、ゴール盤面に持ち込むアルゴリズム実装した。2 つのアルゴリズムを作成しプログラミングしている。

2.解決法

[アルゴリズム 1]

1. ピースは左上から右下に向かって揃えていく。
2. まだ揃えられていない左上の 1 マスを候補地として、候補地をボードの大きさに応じて拡張する。
3. 候補地と同じマスの塊を候補地以外から見つける、又は型抜きをして生成する。
4. 見つけた塊を移動したい方向のマスを型抜きすることによって候補地に移動させる。

この作業を繰り返し、ゴールの盤面にする。

[アルゴリズム 2]

1. 縦と横で要素数が多いほうのベクトルを一塊とする。
2. 最終盤面との要素の種類ごとの過不足数を出し、その過不足数が 0 に近づくように要素数が少ないほうのベクトルで調整する。(また、この時に過不足ができるだけ 0 に近づくように要素数が多いほうのベクトルも使う時がある。)
3. 過不足が全て 0 になったら要素数が多いほうのベクトル内で型抜きすべきピースを見つけ出す。
4. ベクトル間をまたぎながら、手数が少なくなるように、複数のベクトルで同時作業のできる抜き型を駆使して最終盤面に持ち込む。

3.開発環境

VisualStudioCode

1. アルゴリズム

今回の競技内容では、ボードサイズは縦横それぞれ最大 256 ピース、抜き型の数が最大 256 個と入力のサイズが大きく、全ての操作を網羅するのは難しいため、ルールベースのアルゴリズムで問題を解く。

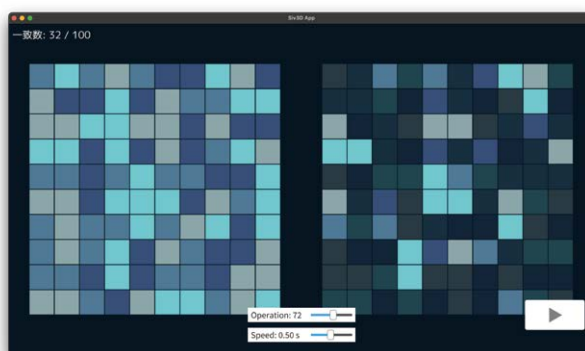
本問題の特徴として、1 回の操作による盤面の変化が非常に大きいこと、型抜きされたピースはボードの端に移動されることの 2 つが挙げられる。

そこで、ボードの端を活用して最終盤面の一部を作り、作成した箇所が後の操作で崩れないように移動させることを繰り返して最終盤面を作る。

また、貪欲法やビームサーチなどのアルゴリズムのほか、ピースが 0~3 の 4 種類しかないなどの問題設定を活用することで、効率的に問題を解く。

2. UI

今回は、Siv3D を用いて下図のような UI を作成した。手数ごとの盤面を左に示し、右の最終盤面と比較する。手数ごとの盤面すべてを保持するのではなく、手数ごとの盤面だけを記録する。この方法により、無駄なメモリ消費と計算を抑えつつ、効率的に盤面の状態を再現することが可能となった。



1. システム概要

本システムは、A*アルゴリズムと chokudai サーチの 2 つの手法を用いる。A アルゴリズムの探索能力と、chokudai サーチの多様性を組み合わせることで、より優れた解を短時間で導き出す。両者の結果を比較し、最適な解を最終回答として選択する。通信部分には Python を使用し、C 言語による高速な演算と Python の柔軟な通信処理を統合する。

2. A*アルゴリズムと chokudai サーチの組合せ

A*アルゴリズムは、探索の効率を重視し、評価関数に基づいて最も有望な経路を優先的に探索する手法である。これに加えて、探索の対象となるノード数を制限することで、処理速度を最大限に引き上げる。ただし、探索が進むにつれて解の多様性が失われ、最適解に到達しにくくなる問題が生じることがあった。この問題に対処するため、chokudai サーチを併用する。

3. A*アルゴリズムの詳細

A*アルゴリズムでは、ヒューリスティックな評価関数として最終盤面と異なるセルの数を使用する。また、オープンリストに追加するノードの数を制限し、評価値が上位のノードのみを探索対象とする。この手法は最適解を保証するものではないが、処理速度を重視したアプローチである。

4. chokudai サーチの詳細

chokudai サーチでも、盤面の評価には A*アルゴリズムと同様に、最終盤面と異なるセルの数を使用する。ただし、探索の深さに最大値を設定し、各ビーム（探索経路）が一定の深さを超えないように制限する。初期盤面から複数のビームを展開し、その時点での最適な盤面を暫定的な解として保持する。さらに、この暫定解を基に再度ビームを展開し、探索結果を効率的に絞り込んでいく。これにより、多様性を確保しつつ、メモリ使用量を抑え、制限時間内に解を得る。

1. はじめに

今回の競技の勝敗は、まず盤面の一致度で評価され、それが同じ場合に手数で評価される。盤面を完全に一致させる手順を求めることは難しくないので、完全一致させる手順の中でどれだけ手数を減らすことができるかが勝負のカギとなる。

そのため、手数は多いが確実に盤面を完成させることができるルールベースによる手法と、より少ない手順を求めることができる可能性のある探索を行う手法を組み合わせ問題で問題を解く。

2. アルゴリズムについて

2.1 ルールベースアルゴリズム

1×1サイズの抜き型を利用して、1行ずつ順番に寄せていくことで盤面を完成させる。

2.2 ビームサーチ

ルールベースの手法では1×1サイズの抜き型以外を使えないので、ビームサーチを用いて探索を行う。盤面を評価する際にはルールベースの手法で盤面を変換する際にかかった手数をを用いて盤面の距離を計算する。

工夫として、そのまま盤面評価を行うには時間がかかりすぎるため盤面のペアからかかる手数を予測するようなモデルを学習することで高速化を図る。

2.3 機械学習

盤面と行動の列から次にとるべき行動を予測するようなモデルを作成して最適な手順を求めるモデルを作成する。

学習の際に最初は手数が少ないものから始め徐々に手数を増やすことで学習を効率的に行う。

11

疾風迅雷

鳥羽商船

横山 朋輝 (3 年) 内田 英都 (3 年)
橋爪 天聖 (3 年) 中古賀 理 (教員)

1. 概要

今回のゲームは、並び替えの有効手の多さと制限時間当たりの計算量から人力の介入は難しいと判断した。そこで、ボードから特定のパターンのピース群を探索し、一度に複数のピースを並び替えるアルゴリズムを考えた。

2. 並び替えの流れ

以下の手順を問題ボードの[0, 0]座標から不一致ピースを対象に行方向から順に行う。

- ① ボードの最終状態から揃える範囲のピース群情報を“テンプレート”として取り出す。
- ② テンプレートに完全一致するピース群をボードから探索する。該当するピース群が無い場合はテンプレートの範囲を縮小して再度探索を行う。
- ③ ②のピース群をピース数や正解座標への並び替えに必要な手数を評価し、評価値が高いものを並び替え、

その手を回答として保存する。評価についてはピース群のピースが多いもの、並び替えの手数が少ないものを高く評価するように設定する。

テンプレートを利用することで、複数ピースの並び替えによる手数短縮と一致度の向上を図る。

3. テンプレートの決定法

テンプレートは以下の条件でボードの最終状態から取り出す。

- ・ボードの不一致ピースの座標のみ使用する。
 - ・範囲は対象の座標から行方向を優先して拡張する。
- 行方向を優先して揃えることで、並び替える際に不一致ピースが生成されることを防ぐ。

4. 開発環境

OS: Windows11 開発言語: Python

エディタ: Visual Studio Code

12

いなかんぼ

福島

阿部 港 (5 年) 常盤 海玖 (5 年)
柳沼 孝太 (5 年) 小泉 康一 (教員)

1. 方針

盤面を落ちものパズルと見立ててそろえやすく整える操作, 中心からボードを最終盤面にする操作の 2 つを実行する。

2. アルゴリズム

2.1 盤面整理

盤面を 2 分割し, 分割した領域内で落ちものパズルを行う (図 1)。例として, 上半分の領域 A でこれを実行すると, 赤線を底面として必要なピースが下, 不要なピースが上に蓄積される。これを右半分の B, 下半分の C, 左半分の D, A と時計回りに実行していくと各領域内で不要なピースが存在しない状態を実現できる。

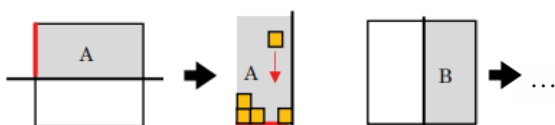


図 1. 盤面整理アルゴリズム

2.2 最終盤面形成

中心から縦横に盤面を 4 分割し, 中心からピースを一致させていく。このとき, 抜き型適用後のピースを挿入する方向は近い外側の面 2 方向に限定される。分割領域同士での干渉がないため, この操作は 4 つの領域で並列処理を行う。

3. 開発環境

【言語】C++

Python

【OS】Windows10/11

【環境】Visual Studio

Visual Studio Code

Siv3D

13 ゴリゴリズム 2.0.0

熊本
(熊本)

永松 日月 (3 年) 牛嶋 叶志朗 (4 年)
富永 颯真 (4 年) 藤井 慶 (教員)

1. GUI について

競技部門は、人力が有効な場合がある。そこで、“Manual”、“Algorithm”、“Auto”の三種のボタンにより、「人力/アルゴリズムで部分的に解く」・「完全オートで解く」モードを実現した。これらにより、試合時間を有効に使うことができるようになる。

人力モードでは、任意の場所にマウス・矢印キーで適用することができ、適用した場合の盤面の変化を表示するなど、有利に進められるような情報を表示する。



2. アルゴリズムについて

独自の評価関数を基本としたビームサーチにより、最適化を目指す。評価には「短期的な損得」と「長期的な損得」を組み込む。前者は主に貪欲法である。後者は、貪欲法だけではたどり着けない、遠回りのようで近道となるような手に着目している。ビーム幅と評価関数を巧みに設定することで、盤面の変化の多様性を確保し、1手ごとの利益を最大化する。また、GPGPUを活用し、盤面サイズが大きくなっても高速に解けるように設計する。

3. 開発環境

OS : Windows11, Windows10, WSL2

言語 : C++, Python

エディタ : Visual Studio, Visual Studio Code, vim

ライブラリ : OpenSiv3D, CUDA

14 CeroRe::Master

群馬

山縣 蒼人 (3 年) 豊巻 煌 (4 年)
渡邊 俊哉 (教員)

1. はじめに

今回の競技部門では、ボードに抜き型を適用し最終状態と一致させることが目標である。この問題の性質上、型抜きをした後のボードを予測することが時間・空間計算量的に難しい。また、ボードの生成方法によって大きくスコアが変わりうる。そこで、複数の解法を使用してボードを一致させる。

2. アルゴリズム

2.1 方針1

ある1つのセルについて同じ行・列の任意の場所に、定型抜き型のタイプⅠを用いて1手、タイプⅡ・Ⅲを用いることで2手かけて移動させることができる。このとき、セルを特定方向の辺にそって揃えていくことで、すでに揃えたセルに抜き型を干渉せず最悪3手で移動できる。

2.2 方針2

まず各行について、同じ行内にあって最終状態に不必要なセルを交換用の列に集める。端の行から順に、最終状態に不必要なセルと他行のセルを交換し、最終状態に必要なセルが揃った状態にする。行の最終状態の連続部分列と、現在の行との最長共通部分列以外を動かすことで行を高速に揃えられ、全体としても少ない手数でボードを一致させられる。

2.3 その他

節2.1、2.2に示した方法を含めた様々な解法を並列に実行し、結果が最も良いものを最終的な出力とする。

3. 開発環境

言語 : C++

ライブラリ : JSON for Modern C++, OpenSiv3D

15

例によって締切寸前で
殺気立つ高専生香川
(託間)大西 俊輔 (4 年) 三宅 巧 (1 年)
野田 八雲 (1 年) 宮崎 貴大 (教員)

1. はじめに

今回の競技部門の問題は、抜き型を適用するたびピースの並びが大きく変わるため、ボード全体を徐々に最終状態に近づける手法は非常に複雑になると考えた。そのため、ボードの端から確実にピースを一致させていく手段をとる。

2. アルゴリズム

最終状態の左上からピースの並びが 1 番多く一致している領域をボードから探索し、最終状態と同じ位置に移動させる。領域を移動させる際使用する抜き型と適用する位置によって他のピースの並びが変わるため、それぞれの場合を計算し最終的な手数が少なくなるように型抜きを行う。

既に一致させた領域を崩さないようにボードの左上か

ら右下にかけて探索と移動の操作を繰り返し、ボードを最終状態にする。

3. GUI

ボードと最終状態、抜き型を表示し、ボードの状況を見て大幅な手数の短縮が狙える場合は手動で型抜きができるようにする。

4. 開発環境

使用言語:C++

エディタ:Visual Studio Code, Visual Studio

ライブラリ:OpenSiv3D

16

while True: print
("HelloProcon")熊本
(ハ代)石堂 優人 (2 年) 嶋中 海人 (5 年)
高松 龍希 (2 年) 小島 俊輔 (教員)

1. システム概要

3 台の PC を用いて複数回探索・型抜きを行い、最も手数が少ないものを回答としてサーバーに提出する。

2. 型抜き

問題フォーマットを模したサンプルデータを自分たちで作成した。このサンプルデータと IDA*探索を主としたアルゴリズムを用いて、抜き型のタイプや適用方向、その後のボードの状態に応じた評価値を算出した。この評価値が最も高くなるような型抜きを繰り返す。

序盤はサイズの大きい抜き型で型抜きを行い、粗く整える。最終盤面に近づくにつれてサイズの小さい抜き型で細かい調整を加える。また、形が不定な一般抜き型での調整が難しい場合、定型抜き型で対応する。

その他、特定のピース配置に対する抜き型の適用順序を事前に計算し、テーブルに保存している。状況に応じてこれ

を参照・適用することで、処理を高速化する。

3. GUI

図 1 に示すように、ボードの状態や試合の進行状況、過去の提出結果等を表示する。

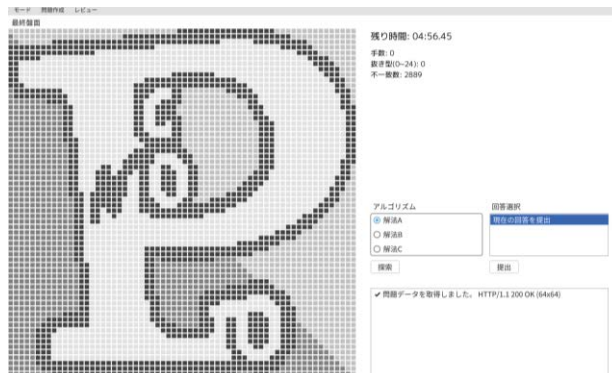


図 1. 開発中の GUI

4. 開発環境

OS: Windows11 IDE: Visual Studio 2022

使用ライブラリ: OpenSiv3D v0.6.15 言語: C++20

1. アルゴリズム

1.1 ボードを最終盤面にする方法

今回のパズルでは一手で多量のピースの位置が変化し、一度最終盤面と一致していたピースがあったとしても、どこかで場所がずれてしまう可能性が高いため、ピースを最終盤面の位置に移動させる方法は難しい。そのため、一度ボードを分割し、分割されたボードの中で必要なピースを集めてから最終盤面にそろえる。まず、ボードを二分割して、分割した片方で0~3の数値の個数を算出して、0~3の数値の個数を最終盤面の0~3の個数と比較し足りないものや余っているものを判断する。抜き型を使用し、足りないものをもう片方からもってきたり、余っているものをもう片方にもっていったりする。一方の個数が合えば逆側もそろえるので片方の動作のみを考えればよい。なお、手を加えた盤面と最終盤面の一致率を評価関数として、評価関数

の値によりの有効性を評価するほか、上記アルゴリズムを切り替える参考とする。

1.2 不一致ピース、手数を少なくする場合

基本的に局所探索法によって次の手を決める。また、盤面が小さく手数が比較的少なく済む可能性のある場合には、山登り法やダイクストラ法の採用も検討する。一手でできるだけ多くのピースを移動させるため、ボードより小さい各抜き型に含まれるものから順に使用する。最も数の多い抜き型で局所探索法を行い、前の状態から、より必要なピースが集まったらその手を最善手として移動させる。最善手が見つからなかった場合は、次に数の多い抜き型で局所探索法を行い、最善手が見つかるまでこの動作を繰り返す。最善手が見つからなかった場合は、その中で評価関数の値が最も大きいものを最善手として選び、ピースを一致させていき、その都度最善手を探していく。

1. 概要

本課題は、一致度を高めることが最重要である。本チームは既一致ピースをずらさないように型抜きを行うことが効率的だと考えた。よって、指定方向を左と上に限定し、左上から順に一致させるアルゴリズムを開発した。

2. アルゴリズム

2.1. 目標座標の設定

ピースを一致させる目的の座標を目標座標と定義し、最終盤面の目標座標のピースを目標ピースと定義する。座標(0,0)から順に目標座標を設定する。

2.2. 型抜き

目標ピースを同行、列、目標座標に近い行の0~N列目の順で探す。目標座標と同列、行になるように型抜きをし、目標ピースを目標座標に移動させる。指定方向を左と上に制限し、既一致ピースを動かさずに一致度を上げる。

2.3. 手数を少なくする工夫

目標ピースの座標と目標座標の距離ベクトル(x, y)を得る。成分それぞれを二進数と考え、n桁目が真のとき、抜き型No(3(n-2)+1)を目標座標に適應する。(n=1が真であれば、No0を適應。)xとyがともに0でない場合は、目標座標のy成分に目標ピースのy成分を代入して計算する。これにより、抜き型No0を複数回用いて一つのピースを一致させることによりを短縮させる。

1	0	3	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0
2	3	0	3	1	3	2	1	3	3	3	2	1	1	1	2	2	2
1	2	2	2	0	3	1	3	1	2	2	2	2	2	2	2	3	3
2	3	3	3	2	0	3	2	3	3	3	2	3	3	3	3	3	3

座標(0,0)から最終盤面に
行ごとに一致度を高める。盤面一致度100%により、一致させていく。
型抜き方向を左、上に制限
し、既一致ピースが移動しないようにする。型抜きを終了。

3. 使用言語・環境など

C++, MacOS, VScode2

19 ボボボード・ボードド

仙台
(広瀬)

工藤 聖晴 (3年) 岸 和希 (3年)
岡田 達樹 (3年) 藤原 和彦 (教員)

1. 概要

私たちはボードの大きさ、型抜きの高さから理論値を探索することは現実的でないと考え、1手ごとの最善手を採用し続けていく手法を使用する。

今回の競技では一番そろったピースが多くなる手を闇雲に採用し続けてもゴールにたどり着かず、局所解に至る可能性が高い。これはボードの端からそろえていくことによって対策し、1手ごとに動作の評価を行うビームサーチを用いる。

2. 通信部

問題の解答とサーバーとの通信はそれぞれ別のプログラムで実装する。通信部では問題と解答を GUI で確認してサーバーへ送信する。

3. 開発環境

言語: Python、C++



図 通信プログラムの GUI

20 DRL 一だって乱数で learning データ集めたから

大阪公大

田村 唯 (5年) 竹内 珠佑 (3年)
辰己 由空 (2年) 窪田 哲也 (教員)

1. 概要

今回のゲームでは考えるボードのサイズが最大で 256x256 というので純粋なアルゴリズムで処理するのは難易度が高いと考えた。そこで、深層強化学習を用いて現在の状態から使用するべき抜き型と座標、方向の三つの要素を推測する。

2. アルゴリズム

SAC(Soft Actor Critic)という深層強化学習アルゴリズムをベースとして開発を進めた。各ネットワークには CNN(畳み込みニューラルネットワーク)を使用することで各ピース間のつながりを考慮しながら、与えられたボードと目標ごとに学習を行う。事前に適当な型と目標で学習を行い、初期の重みを学習に有利なように調整する。

3. 工夫点

3.1 高速化

環境(抜き型使用後のボードや報酬を出力する class)を python でなく C++で実装しそれを python 上で使用することで学習時間の短縮を図る。

3.2 GUI 化

型を数値ごとに色分けし、目視で目標値との差を確認できるよう GUI を作成。

4. 開発環境

Python, pytorch, c++, OpenSiv3D, Windows11

21

鹿の手も借りたい

弓削商船

岡田 慈英 (5 年)

CHALEUNKIT THANASINE (5 年)

松田 柚鈴 (3 年) 長尾 和彦 (教員)

1. はじめに

今回の競技では、不一致ピースが少ないほど順位が上がるため、どんな盤面からでも必ず最終盤面と一致させることができるアルゴリズムを作成する。

2. アルゴリズム

2.1 列の要素数を揃える工程

まず、ボードの各ピースに対し、それぞれ何列目で使われるかを定義する。その後、各ピースを定義された列へ型抜きを使って寄せていき、1 列目から要素数を揃えていく。

2.2 全体の盤面を揃える工程

先ほどの工程で各列の要素数が揃ったため、上から順番に合致したピースを移動させることで、必ず全体の盤面を揃えることができる。

3. GUI

アルゴリズムの考察を助けるため、マップの表示と型抜き

きの実行を行うことができる GUI を用意する。実際の競技では、盤面の大きさが最低でも 32*32 と大きいので、盤面の表示は行わず、アルゴリズムの実行と回答データの送信のみを行う。

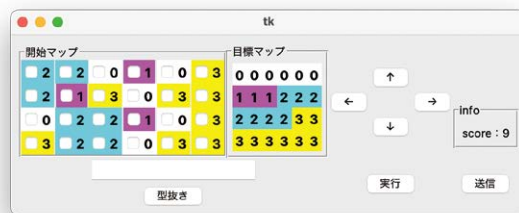


図 1 GUI

4. 開発環境

使用言語 : Python3

IDE : PyCharm, VSCode

22

高専祭には目もくれず、
彼らはプロコンに来た

釧路

小関 将人 (3 年) 櫻井 文太 (3 年)

松本 浩貴 (2 年) 天元 宏 (教員)

1. システム概要

ネットワーク経由で提供された問題を読み取り、システムで取り扱いやすいデータ構造に変換し、その後にアルゴリズムを適用して最適解・準最適解を求める。

2. アルゴリズム

探索範囲が膨大なため、何も考えずにアルゴリズムを適用するとメモリ使用量と計算量が爆発的に増加することが考えられる。そのため、ビームサーチと A* アルゴリズムを併用して問題を解く。

まず、最初にビームサーチでコストの概算を行って A* アルゴリズムの初期探索範囲を制限する。その後、A* アルゴリズムで本格的な探索を行う。

2.1 ビームサーチ

基本的に、ビームサーチでは適切な最適解を求めることはせず、コストの概算を中心として行う。ビーム幅を指定

でき、これにより一定候補だけを残して探索する。ビーム幅は CUI 経由で指定できるようにする。

2.2 A* アルゴリズム

次に、A* アルゴリズムで最適解または準最適解を求める。ビームサーチで得られた情報を元に、効率的な探索を行う。制限時間までに最適解を求められないと推測される場合は、途中で中断し、その解で提出する。

3 その他

計算が終わらず、何も解を提出できないまま制限時間に達してしまう事態を防止するため、一定時間ごとに途中解を提出する。この際、前回提出した解より良い解であるかどうかを比較し、よりよい解でなければ提出しない。

4 開発環境

言語 : Python 3

開発環境 : Visual Studio Code, PyCharm, Windows 11

23

チーム名が決まらなかったの
でお好きなように呼びください

都城

園田 馳弥 (3年) 加治屋 翔大 (3年)
東園 獅之丸 (2年) 丸田 要 (教員)

1. はじめに

本大会における課題は、「短時間でいかにして最終状態までの正確かつ最短手順を導くか」だと考え、効率よく探索ができ、なおかつ高確率でよい手を導くことができるモンテカルロ木探索を採用しました。

2. 戦略

✧ アルゴリズム

モンテカルロ木探索を用いて、あらかじめ、ボードの状態に沿って最適な抜型を選択できるようにシミュレーションを行い、そのシミュレーションに沿って問題に記載されている最終状態に近づけるというアルゴリズムです。

✧ 操作性

主な操作はGUIで行い、抜型の適応回数を指定することや、必要に応じて抜型を手動で適応するこ

とを可能とすることに苦労しました。



※上図は、制作過程のものです。

3. 意気込み

去年は参加こそできたものの、敗者復活戦敗退ということでも悔しい思いをしました。今大会ではその気持ちをばねにして、少しでもいい結果を残せたらと思います。

4. 開発環境

エディタ : Visual Studio Code ・ Visual Studio 2022
開発言語 : Python ・ C#

24

クッキーカッターズ

香川
(高松)大鹿 友滉 (4年) 大石 悠馬 (4年)
伊戸 寛哲 (3年) 柿元 健 (教員)

1. 概要

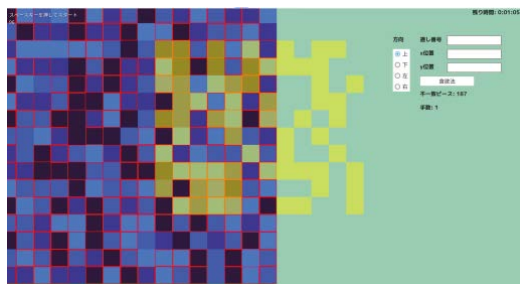
今回の競技は多種多様な型や動かし方などがありそれらをうまく駆使することによって最終盤面にもっていく。全通りを試すとなると莫大な計算量となるため、方向やあてはめ方を工夫していくことが大切である。

2. アルゴリズム

今回のアルゴリズムは開始の盤面において左上のピースを一致させることを考える。この時寄せ方向は左上である。このようにすることで、左上から一致させていった場合、その部分は必ず型をあてはめないと移動しないため、無視して他部分をそろえられる。評価関数としては、最初にそれぞれのマスに一致させたい座標を定め、その座標に近づいた場合、スコアを加算、遠のいた場合スコアを減算する。その中で最もいいものを手として採用する。

3. GUI

手動で操作できるように現在の盤面を色を用い表示した。また残り時間や寄せ方向、型の表示などを視覚的にわかりやすく表示している。



4. 開発環境

OS: windows11
使用言語: c++, python
IDE: visual studio
Frameworks: Siv3d

25 夜行バスには乗らない

仙台
(名取)

鈴木 佑 (4 年) 佐藤 至 (5 年)
黒澤 徹真 (2 年) 佐藤 隆 (教員)

1. はじめに

今回の競技では、最大 256 個×256 個のピースで構成されるボードを、最終状態との不一致ピースが可能なかぎり少ない状態になるまで修復する事が求められている。

2. アルゴリズムについて

2.1 アルゴリズムの流れ

我々のチームは型抜きをおこなった際のピースの動きに着目した。型抜きが適用されると、抜き型の 1 の値のセルに対応するピースの座標は、指定された方向の端っこの座標に変更される。この動きは同方向であれば型抜き位置の座標に依存しない。そこで端のピースの座標を基準として抜き型を探索するアルゴリズムとした。

まず縦横どちらで型抜きをおこなうか方向を指定し、指定された方向の端から順に 1 回の型抜きで最終状態と最大限一致するような型抜きをおこなう。つづく型抜きでは

一致した部分を除いた範囲に対して、同様に端から処理をおこない、最終状態まで修復する。

2.2 指定しなかった方向に対する型抜きについて

前処理として、指定しなかった方向に対する型抜きをおこなう。これは、座標の一致・不一致は問わずに、指定した方向の列または行において、そのなかに並んでいるピースに振られた数値が、すべて最終状態と一致するように実施する。たとえば、型抜きの方向を横方向に指定した場合には、縦方向の型抜きを、各行に存在する各ピースの数値と最終状態のものとが同じになるまで繰り返す。

2.3 抜き型の探索について

基本的には 2.1 で述べた型抜きをおこなえるような抜き型を探索する。望んだ型抜きができない場合は、1 回の型抜きで一致させる端の最大値を下げる事や、動かすピースの座標を変更する事で柔軟に対応する。

26 npm install -D 型抜き

近畿大

楓 悠斗 (5 年) 西半 涼賀 (5 年)
山本 隆翔 (4 年) 坂東 将光 (教員)

1. はじめに

今回の大会では、抜き型を用いて盤面を目的の形にすることが求められているが、我々は、手数よりも正確性を重視したアルゴリズムを採用し、確実に得点を積み重ねることを目指している。特に、シンプルなアルゴリズム設計により、安定したパフォーマンスを発揮するだけでなく、操作性を向上させるための GUI も実装し、人間との柔軟なインタラクションを可能にする。

2. システムの詳細

2.1 アルゴリズム

前述のとおり、手数よりも正確性を重視することを目標としているため、計算量を抑え、1 マスずつ確実にそろえるためのアルゴリズムを基本としている。

2.2 GUI

人間による介入を目的とするため、盤面および抜き型の表示、操作の手動入力、回答の作成を可能とする GUI を、React を用いて作成する。

2.3 その他の処理

問題の取得、回答の提出に必要となるサーバーとの通信は、セキュリティの観点からブラウザで行うことは推奨されない。この問題を解決するため、Electron を用いてバックエンドの処理を実装する。

3. 使用言語および開発環境

使用言語: JavaScript

フレームワーク等: React, Electron

OS: Windows 11

使用ソフトウェア: Visual Studio Code, Node.js

27 ラウンドロビン

沼津

藤田 康佑（2年） 山本 環太（4年）
鈴木 康人（教員）

競技部門

1. 問題へのアプローチ

1.1 解法1

合法手の数が膨大であるため、私達のチームはこのパズルをトーラスパズルの一種だと考え、簡略化した。なお、トーラスパズルと同じ考え方を使得行うより手数が少なくて済む場合があるときは、トーラスパズルの合法手に限るという制限を無視して操作する。

1.2 解法2

当初は、問題から型抜きを、完成から型抜きの逆をそれぞれランダムに実行する。まず、その結果一致した盤面があればそこから完成させる手順がわかるのでそれを使う。ただ、この手法でシミュレーションすると正解到達率 30 パーセントが頭打ちで効率が悪いと考えたため、ランダムに

行動するのは最初の数百手で打ち切りにし、点と点を入れ替えるアルゴリズムでそろえることにした。

解法1,2のうち問題によって適切な方を選択し、解決する。

2. GUI について

今回大会は例年の競技と比べてできる行動の数が数百万以上と非常に多く、また盤面が大きいことや、操作結果が人間にとって想像しにくいことから人間の介入は不可能と考えたため、盤面がどのような状況になっているか分かるようにするもの以外は作成しなかった。

3. 開発環境

Windows11 Pro, Python 3.12.0, VScode

28 編入数学徹底研究 徹底研究部

沖縄

平良 昂也（専攻科2年）
田中 柊（5年） 知念 涼太郎（5年）
金城 篤史（教員）

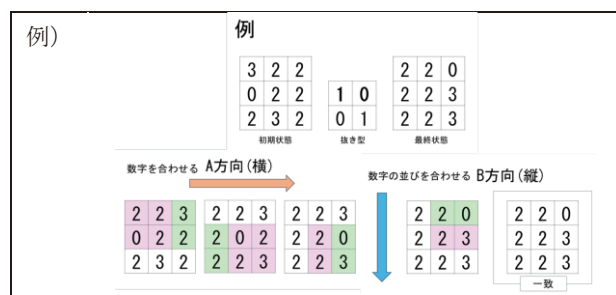
1. はじめに

今回の競技部門は計算量が膨大な点から、手数よりも一致度が重要と考えた。そのため、手数は多くなるが、一致度が高くなるようにするためにピースの移動をA方向とB方向に分けて段階的に行う戦略を採用した。

2. 戦略

2.1 ピースの移動を段階的に行う

ピースの移動を横方向と縦方向に分けて考える。初めにA方向（今回の例の場合は横）の各数字の個数が揃うようにピースを移動させる。次に、B方向（今回の例の場合は縦）の各数字が最終状態の並びになるようにキューの整列を行う。このようにピースの移動方法を制限し、段階的に移動を行うことで、計算量と不一致ピースが減少すると考えた。



2.2 型抜きを使う順番

サイズが大きい型抜きから使用し、現在の移動方向にある数字の数と配置を数字ごとにチェックする。現在の型抜きでボードが最終盤面に近づかない場合に次の小さい型抜きを使用する。

3. 開発環境

動作環境：Windows11、Windows10

使用言語：Rust、Python

29 神託機械？～逆襲の if 文～

大分

引田 鴻志（3 年） 染矢 雅尊（2 年）
村井 春陽（2 年） 徳尾 健司（教員）

1. 概要

本チームでは AI を用いた自動回答を行うアルゴリズムと、全探索を行いある決められた順番に入れ替えを行うアルゴリズムの二つを用いて、適宜使い分ける。2 つの方法を組み合わせることによって、どの場面においても少ない手数で確実に問題を解くことができる。

2. 競技中の流れ

まず、サーバーからデータの受信を行う。次に初期状態、最終状態、一般抜き型の数や形などの情報を元に適切なアルゴリズムを選択する。選択したアルゴリズムに基づいて処理したのち、サーバーに結果を送信する。

3. アルゴリズムの内容

AI を用いたアルゴリズムには、畳み込みニューラルネットワークと強化学習を組み合わせた手法である DQN

(Deep Q-Network) を用いる。これを用いて型抜きの手数が小さくなるような手を導くモデルを事前に構成し、問題を解く。

全探索アルゴリズムは、基本として 1x1 の定型抜き型を用いて座標(0, 0)を作業部とする。各行ごとの各ピースの個数を交換する行を探索しながら揃え、すべての行が最終状態とその行が持つ各ピースの個数を等しくする。等しい状態になったら、各行の並びを順番に整える。このアルゴリズムは定型抜き型を基本的に用いるため、どのような状況であっても確実に修復することができる。

4. 開発環境

OS: Nix OS 23.11, Ubuntu 22.04.3 LTS, Windows 11

言語: C++20, Python3

エディタ: NeoVim, VSCode

30 限界集落

豊田

宇井 颯汰（5 年） 鈴木 健太（5 年）
吉野 弘晟（1 年） 三浦 哲平（教員）

1. システム概要

初期状態から操作を行い最終盤面へとそろえるプログラムと、操作の途中の盤面を確認したり、問題や回答データの送受信をしたりすることができるプログラムを作成する。

2. アルゴリズム

定型抜き型のみを用いて初期盤面から 1 行ずつ完成させ、最終盤面へとそろえていく。

2.1 ある行の各ピースの色がそろえたい位置の真下にすべてある時

抜き型の左下にそろえたいピースと同じ色のピースが来るような寄せ方に注目する。その内、操作後に他の列のピースの色が横にそろえる数を評価値として、最も良い値となったものを操作として選び、1 行をそろえる。

2.2 2.1 でないとき

部分的にそろえたい位置の真下にピースがある時、2x1 や 1x1 の定型抜き型を適切に用いる。そうでない時、同じ色のピースが真下に来るような操作を行い、定型抜き型を用いることでそろえることができる。

3. ビジュアライザ

現在の盤面の状態を表示するビジュアライザを作成する。盤面の表示によってより効率的な解法を模索し、アルゴリズムの強化に利用する。また、問題データの受信は回答の送信などもビジュアライザで行う。可能な限り手動部分を減らしてヒューマンエラーを防ぐ。

4. 開発環境

使用言語: C++, Python3

開発環境: Visual Studio Code, Visual Studio 2022

OS: Windows, Ubuntu (WSL)

ライブラリ: OpenSiv3D

31 ヤメテアゲテネ

和歌山

岡崎 惇平 (4 年) 浅井 正希 (4 年)
坂口 勢吹 (4 年) 森 徹 (教員)

1. アルゴリズムの概要

基本的に縦横のサイズが 1 である定型抜き型 0 番の抜き型のみを使用し、1 行ずつ順にボードを最終盤面に近づけていきます。0 番を用いることによって 1 ピースずつ確実に最終盤面へと近づけることができます。

1 行ずつそろえていく

1 行目	0 1 2 1 2 1 2 3 . . .
2 行目	1 2 0 3 1 3 2 1 . . .
3 行目	0 1 3 2 1 2 1 0 . . .
4 行目	3 2 1 2 0 1 1 1 . . .
5 行目	0 0 1 1 0 2 1 3 . . .

2. エフ

一般抜き型は全く使用せず定型抜き型、さらにいうと縦横サイズが 1 である 0 番しかほとんど使用しません。0 番のみでは手数が多くなってしまいますが、アルゴリズムを簡単にして確実にそろえることを重視するため、一般抜き型は使用しません。また、一般抜き型を考慮しないことによりコードの量は少なくなり、最終回答時間が早くなります。

3. 開発環境

言語 : C++

IDE : Visual Studio Code

32 創立 50 周年記念

徳山

棟近 祐希 (5 年) 児玉 隆之介 (5 年)
門原 裕大 (3 年) 力 規晃 (教員)

1. 探索アルゴリズム

複雑な操作を禁止し、問題を簡単にすることで最小費用流の問題に帰着させ効率的に問題を解く。また、最小費用流に帰着させることが難しい部分はメタヒューリスティックの技法を用いて良い解を求められるようにした。

1.1 最小費用流

使える抜き型を制限し特定の箇所のみ揃えることを考えると、盤面を揃えるのに必要な操作回数の最小値は最小費用流に帰着させることで厳密解が求められる。よって、問題をいくつか分割してそれぞれに対して最小費用流を適用する。

1.2 メタヒューリスティック

焼きなまし法やビームサーチなどのアプローチを適用することで近似解を得られる。これらと最小費用流を組み合わせることで、短時間で良い解を得られるようにした。

1.3 その他の工夫

Rolling Hash や Z-algorithm などのライブラリや並列処理を用いることでプログラムを高速化した。

2. ユーザインタフェース

図 1 に UI を示す。割り振られた整数値でピースの色を塗り分けてボードの状態を表示する。ボードのサイズやピースの色の偏りによって解法を選択する機能も有する。

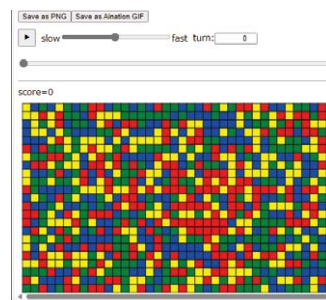


図 1 GUI のスクリーンショット

3. 開発環境

言語 : C++, Python, Rust

ライブラリ : OpenSiv3D, AtCoder Library,
visualizer-template-public

33

新居浜

中家 海翔 (3年) 中平 智也 (3年)
西元 銀弥 (3年) 占部 弘治 (教員)

1.概要と背景

本大会ではボードが与えられ、抜き型を使ってそのボードを復元することが目的である。

また復元のパターンは膨大にあると考えられるため、計算リソースや開発期間の都合上、PyTorch などを使用した AI のみによる解法は困難であると予想した。

そこで探索アルゴリズムを利用することにした。

2.解法

探索アルゴリズムとしては、深さ優先探索や幅優先探索などがあるが、復元のパターンは膨大であると考えられるため、これらの手法では探索に時間が掛かると予想できる。そこで、ヒューリスティック的な解法を併用し、探索の高速化を考えている。

また、ボードの復元状況やその過程などのさまざまな局

面に応じたアルゴリズムを用意することも検討している。
そのアルゴリズムの切り替えは人間が行うことを想定している。

3.今後の展望

計算量が多くなりがちなパズル問題において、アルゴリズムを実装するだけではなく、高速化にも取り組みたい。高速化をするために、ボードの復元率が低い状況では、なるべくサイズが大きい抜き型を使用することで復元率を上げ、復元率が高い状況では小さい抜き型を使用し、復元率を上げている。

4.開発環境

OS : Windows10

言語：C#(.NET8.0)

34

高知

福岡 伊織 (4 年) 宗石 響 (2 年)
立川 崇之 (教員)

1. 概要

今回のゲームにおいて、膨大な計算で導けることから、処理速度が鍵となると考えた。本プログラムにおいては、開発時には Python を使用して、本番環境では Rust を使うことを考えた。

2 本プログラムの特徴

本プログラムでは、Python のみの開発が多い中、フロントエンド、バックエンドを分けて開発を行った。これにより計算リソースの増強を行うことができた。

3. 問題解決

ゴール状態から現状の状態まで探索を行う。計算サイクルを早く回すことにより、膨大な量の計算を可能とした。また、複数案出た際には人が選択を行う

4. 開發環境

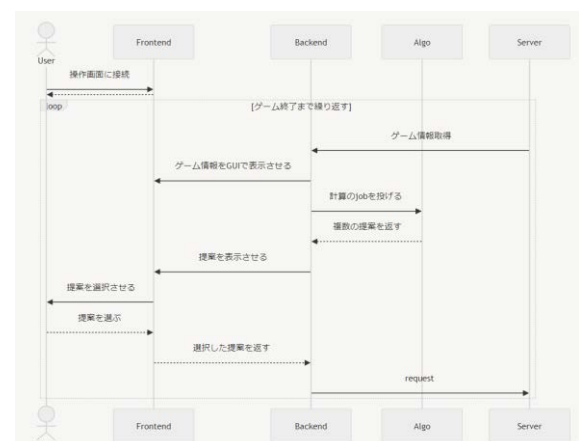
OS: windows 11 pro

Frontend: Typescript x React x TailwindCSS

Backend: Rust

Algo: Python

4. シーケンス図



35 Static Programming

鈴鹿

白髭 樹（4年） 中村 春琉（3年）
市川 敬士（3年） 青山 俊弘（教員）

1. はじめに

今回の競技部門は、ピースを矩形に並べて構成されたボードを抜き型によって修復前のボードから修復後のボードに「修復」するまでにかかった手数、修復できたピース数を競うものである。

2. 戦略について

戦略としてはモンテカルロ法を用いた愚直な焼きなまし法や、強化学習、以下に述べる独自のアルゴリズムといった複数の方法を同時に適用し、より良い結果を得るものを採用するという方針をとる。

2.1 各行の数字の数を揃える

修復前と修復後において 0~3 の数字が各行においてどの列に存在しているかを確認する。確認したものは各行各数字について set で座標を管理する。

管理された座標に基づいて余分に存在する種類の数字

を別の場所に存在する必要な種類の数字と入れ替える。

2.2 各行の数字を正しい順序に揃える。

正解の並び順の間に不要なものが挟まっているかどうかを評価値とし、各行において抜き型を使用する回数を最も抑えられる場所を計算する。

2.3 手数を抑える方法について

通しでプログラムを実行するとして、抜く場所や、使用する抜き型を変更するなど行う。

変更を行った際には修復できたピースの数、手数の最高値が更新する可能性があるため、更新したときには毎回手順を出力するようにする。

3. 開発環境

使用言語：C++

使用ライブラリ：Boost C++・OpenCv3D

使用エディター：VSCode

36 人力全探索

小山

森 悠貴（2年） 鈴木 真（2年）
藤澤 怜央（2年） 小林 康浩（教員）

1. はじめに

本チームは確実に盤面を揃えられるように、主に貪欲法を用いてできるだけ少ない手数で盤面を一致させることを目指す。

2. プログラムの概要

このプログラムでは、1行目の左端から右端、2行目の左端から右端…という順で盤面と最終盤面を比較し、不一致ピースに対して主に 1×1 の抜き型を使用して最終盤面とピースを一致させる。必要なピースは揃えていない部分から探し、また揃える対象と同じ行、列から優先的に探す。型抜きは揃える対象の位置に行い、対象の位置と見つけたピースの位置から動かす方向を定め、型抜きを繰り返す。この操作をすべての不一致ピースについて行うことで盤面を揃える。

3. GUI

作成したプログラムがどのように動いているかの把握を行いやすくするため、数に対応した色を付けて GUI を作成し、盤面の推移を視認できるようにした。

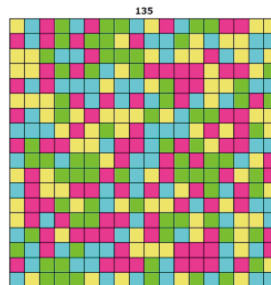


図1 GUI のイメージ

4. 開発環境

使用言語：C++, Python3, JavaScript

開発環境：VSCode, Windows11

37 回鍋肉

松江

三島 知樹（専攻科2年）
田中 庵（3年） 長谷川 友音（4年）
橋本 剛（教員）

1. はじめに

今回の課題は盤面に対し複数回の操作を行うことで完成形を目指すものであるため、複数の状態を持ちながら盤面を徐々に初期状態から完成形に近づけていくことを目指す。

2. アルゴリズム

ビームサーチを用いる。盤面の upper 側から完成形に合わせていくことを考え、評価関数を作成する。また、状態の遷移が膨大になるためそれを削減するための工夫も取り入れる。

3. テスト

本番でどのような問題が与えられるかわからないため、事前に様々なケースを想定した問題を複数作成・検証し、最も性能の良いプログラムを採用する。

4. ビジュアライザ

回答の作成に人間が関与することはないが、問題や生成された回答の把握、開発時のデバッグのためにビジュアライザを作成する。

5. 開発環境

OS : Windows

使用言語 : C++, Python

開発環境 : Visual Studio, Visual Studio Code

38 深夜テンション

岐阜

渡邊 雅晃（4年） 福富 陽生（4年）
北川 大雅（2年） 菊 雅美（教員）

1. はじめに

最初に、行にある数字の個数を最終盤面と一致させ、次に列を合わせるというようにして完成させる。

2. 戦略

大きく分けて二つの段階に分けて処理を行う。この方法で導き出せる手順を時間の限り導く。

2.1 第一段階

盤面の各行に含まれるピースの個数を最終盤面と合わせる操作を行う。現在のボードの一番下の行を最終盤面のボードの各行にある数字のピースの個数と同じになるように抜き型を使う。その後、一番下の行を丸ごと一番上の行に移動させるように抜き型を使う。これを列の数だけ繰り返す。使う抜き型が確定できない場合は、ビームサーチを用いて決定する。

2.2 第二段階

全てのピースを最終盤面と一致させる。現在の盤面の一番左の列を、最終盤面の各列と一致させるように抜き型を使い、一致したら一番右に移動させる。これを列の数だけ繰り返すと全てのマスが最終盤面と一致する。

3. GUI

Siv3D を用いて、プログラムの実行の進捗など情報の表示を画面表示によって実現する。進捗状況に応じて、型抜きの方針を切り替える仕組みをボタンを用いて実現する。

4. 開発環境

OS : Windows11, MacBook Air

言語 : C++

フレームワーク : Siv3D

IDE : Visual Studio

39

if(優勝!!){番狂わせ}
else {また来年!}

鹿児島

柿木 瑞貴 (3年) 花田 純正 (2年)
亀沢 優月 (2年) 武田 和太 (教員)

1. はじめに

今回のシステムでは基本的に A*アルゴリズムを使った探索によって次の手を決定する。この競技では人間の判断でシステムの回答をサポートするのは難しいと判断したため今回のシステムでは基本的に人間は介入せずにサーバーに送信するまでを実装してある。

2. アルゴリズム

今回のシステムで開発初期の段階では AI を使おうと考えていたが、合法手が多く、精度がでないと考えたため、A*アルゴリズムを使った探索を行うことにした。

A*アルゴリズムで探索を行っているときメモリ不足によって処理落ちしてしまうことが多くあったため、ある自分で決めたところまで探索を行った後その時点で一番評価値が高いものを次の手として更新する。

3. 開発環境

OS : Windows11

使用言語 : C++, Python

エディタ : Visual Studio Code

40

C たすたす

サレジオ

井上 将秀 (4年) 境 結葉 (3年)
小林 千尋 (4年) 宇都木 修一 (教員)

1. 探索方法

焼きなまし法などを使用し、探索する前に大雑把に最終盤面により近い盤面とした後に、15 パズルなどを解くときに使われる A*アルゴリズムを使用することによって最終盤面までの最適解の導出を試みる。

最終盤面に至らなかった場合は、特定の抜き型のみの使用を想定した A*アルゴリズムや、かかる手数などを一切無視して抜き型によるある特定の操作を繰り返して無理やり最終盤面にする独自のアルゴリズムを使用してより高速に不一致ピースを少なくする。また、ボードの状況に応じて様々な探索方法の使い分けを行う。

2. GUI

ボードの状況の直感的な把握や、探索方法の指定に図 1 の GUI を使用する。

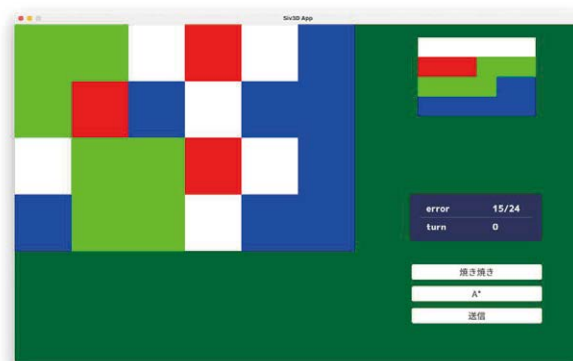


図 1 開発中の GUI 画面

図 1 では、左側の大きいパズルが問題の初期盤面であり、右上のパズルが問題の最終盤面を表している。

3. 開発環境

開発環境: Mac OS, Ubuntu

言語: C++ フレームワーク: OpenSiv3D

41 聖徳 .mch

苦小牧

藤田 響（4年） 治田 風冬（3年）
野澤 和真（2年） 山本 椋太（教員）

1. はじめに

今回の競技部門ではブース内で型抜きが一番効率的な解を求め、点数を競い合う対戦方式の試合であり、パズルのようなゲームであると考えた。このようなゲームは解法が大量に存在しており、型抜きをする上で最適な解法を見つけ出すために、探索アルゴリズムを用いた手法と AI を用いた手法で問題を解き、より最適な解を採択する方針でアルゴリズムを設計した。

2. 解の探索方法

2.1 探索アルゴリズムを用いた手法

幅優先探索、深さ優先探索のアルゴリズムを用いて、どのように型抜きをし、最終状態にするかを計算する。例えば、試合が開始した時、このアルゴリズムから考える型抜きの解を探索し、解の決定に有益な情報を計算する。

また、これらの処理について探索範囲が広く計算時間が大量に必要になってしまう問題への対策として、CPU の性能を活かせるようなマルチプロセッシングによるプログラミングを行っている。

2.2 AI を用いた手法

tensorflow を用いて構築したエージェントを Deep Q-Learning により学習し、解の決定に有益な情報を計算する。

2.6 開発環境

言語: Python

開発環境: wsl2, Docker, Python 3.11.9

42 プロコンに傷心

久留米

平井 零大（3年） 古川 樹（3年）
日永田 幸多（1年） 古賀 裕章（教員）

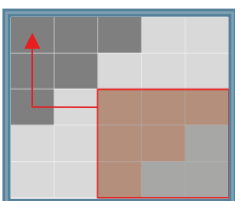
1. 戦略

不一致ピースの数を 0 にすることは容易なため、完全一致させるまでにかかる手数を少なくすることで他のチームより優位に立つことを目指す。

2. 概要

2.1 アルゴリズム

左上のピースから順に一致させていく。まず、それぞれの位置で最終盤面のピースに一致するものを現在の盤面から探索し、発見したピースから、さらに「階段状に連なる最終盤面に一致するピース群」を探索する。



次に、それらを「発見したピース数/一致させるのに要する手数」で評価し、最も評価の高いものを x 軸、y 軸方向に平行移動させて一致さ

図（階段状に連なる最終盤面に一致するピース群の例）

せる。これを繰り返すことで最終盤面に完全一致させる。

2.2 手数を減らす工夫

ある抜き型を適用したい範囲があるとき、その全て、あるいは可能な限り広い範囲を覆うことのできる抜き型を適切に選択し、不要な手数の増加を抑えている。

また、上の行から順に「行内のすべての要素が一致したときその行に抜き型を適用して、その行を盤面の最下端へ移動させる」という操作を行う。これにより前述のアルゴリズムで探索されたピース群の移動先が常に盤面の最上端となるため、十分に盤面のサイズが大きい場合に限り、移動に掛かる手数を減らすことができる。

3. 開発環境

言語: C++, Python

エディタ: Visual Studio Code

OS: Windows 11, Ubuntu 22.04.3

43 410 Gone

宇部

山本 健一朗 (3 年) 池本 悠生 (3 年)
井藤 誉晴 (1 年) 田辺 誠 (教員)

1 方針

左上から 1 ピースずつ修復を行う貪欲アルゴリズムを考えると、定型抜き型を上手く使うことで 1 ピース辺りの操作回数を 3 回以下にできる。私たちはこのアルゴリズムを基に更に操作回数の改善を試みた。

2 ビームサーチ

この問題のような「操作を繰り返して評価を高くする」という形式の問題はビームサーチという手法が有効であることが多い。前章の貪欲法を基にビームサーチを作ると、ビーム幅が 100 程度でも操作回数は貪欲法のおよそ半分になった。残り時間からビーム幅を調整するようにして、制限時間に間に合わせつつ大きな計算量を保てるようにした。

3 データ表現の工夫

ボードのデータ表現で 64 ビット整数型にピースの情報を 32 個ずつ乗せるようにすると、単純な列挙型の二次元配列に比べて省メモリ化できるだけでなく、ピースを 32 個の塊で動かせるようになり処理時間の改善にもなる。

ビームサーチ中に保持する操作列も同様に省メモリ化して、コピーコストを削減した。

4 GUI

どのような問題でも柔軟に対応できるようにするため、人力でも問題に挑めるよう GUI を用意した。図 1 のように画面に問題を表示し、抜き型をドラッグして操作できる。

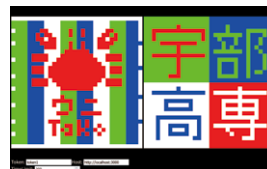


図 1: 開発中の GUI

5 実行環境 / ライブラリ

3 台の PC の内、1 台を回答用、残りの 2 台を探索用として用いる (図 2)。

OS

- Windows 10/11
- WSL2 (ArchLinux)

TypeScript

- Deno 1.45.5
- esbuild v0.20.0

C++23

- GCC 14.2.1
- cpp-http lib v0.16.3

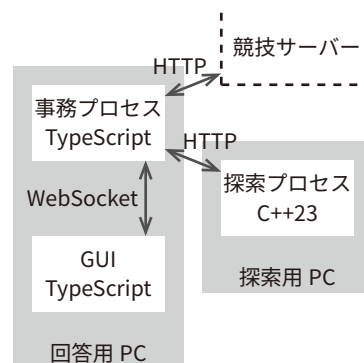


図 2: システム構成図

44 あげししゃも

旭川

長谷川 碧 (3 年) 長谷川 輝 (3 年)
晒谷 悠斗 (3 年) 森川 一 (教員)

1. システム概要

今回の競技では、できるだけ少ない手数と早い実行時間で最終盤面を完成させることが重要である。しかし、盤面の大きさや抜き型によって実行時間が左右されやすい。そこで、手数の短縮を重視したアルゴリズムと、確実に時間内に終わらせることを重視したアルゴリズムの 2 つを同時に実行し、完成したものから大会側のサーバーに送信するという方針を採る。

2. プログラムについて

2.1 プログラム 1

本プログラムは、最少手数での完成を目的としている。ビームサーチを用い、評価値は最終盤面との一致度とした。すべての抜き型を適用し、盤面のすべての座標に、すべての方向にセルを移動させ、評価値が最も高いものを一手と

する。

2.2 プログラム 2

本プログラムは、実行速度と完成度に重きを置いている。プログラム 1 が実行に時間がかかることが予想されるため、それを補うことが目的である。使用する抜き型と抜き方 (誤入力ではない) を限定的にすることで、実現している。そのため、手数が最適になるわけではないが、制限時間内に完成形を確実に送信する。

3. 開発環境

言語 : Python、C

OS : Windows 11

エディター : Visual Studio Code 2022

45 1年生教育研修ツアー

富山
(射水)

土橋 晴人 (4年) 岡本 倫楓 (1年)
廣田 響記 (1年) 山口 晃史 (教員)

1. 概要

この問題は、初期状態のボードと最終状態のボードを比較し、以下に正確に、そして早く最終状態の形に持っていた。特にルール上正確である方が重要なため、絶対に誤配置をしないような方針で作成した。

2. アルゴリズム

まず、初期状態のボードと最終状態のボードを比較し、不一致であるピースを見つけ出す。そして、不一致ピースに一番近いピースを、BFSを用いて探索する。その後、ピースを移動させるが、1回の移動距離をできるだけ増やす、一度に移動できるピースを増やし移動回数を削減するなどして、移動回数を最小限に抑える工夫をする。

また、移動が完了したあとに盤面チェックを行うことで、移動のミスをなくすようにしている。

3. GUI

盤面の操作前・捜査後の情報のみではプログラムが正しく動作しているか分かりにくいいため、盤面の様子や移動回数、探索状況を表示し、正常に動作しているかを監視できるGUIを開発する。

5. 開発環境

OS : Windows11

使用言語 : C++/Python

IDE : Visual Studio

フレームワーク : OpenSiv3D

46 少納言大豆

阿南

米積 悠翔 (2年)
Naphat Soonthornkorani (2年)
寺尾 惟吹 (2年) 平山 基 (教員)

1. はじめに

私たちは三つのゾーンを用いたアルゴリズムを使う。

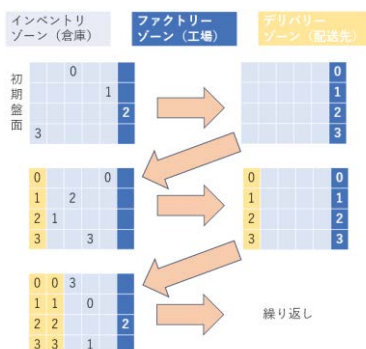
2. アルゴリズムについて

2.1 基本方針

三つのゾーンをピースが流れるように動いていき、一致率100%に少ない計算量で到達できる。

2.2 図を用いた具体例の説明

以下のように三つのゾーンを使い分ける。



2.3 メリット

このアルゴリズムのメリットは盤面のサイズが最大でも高いパフォーマンスを発揮できることである。5分以内にゴール盤面まで並びかけることができる。

2.4 デメリット

このアルゴリズムのデメリットは手数が多くなることである。

2.5 システムの構成

開発にはC++を使用した。通信から回答までの、すべての工程を単一の言語で行うことで管理がしやすくなり、開発効率が上がる、

3. まとめ

このアルゴリズムは探索には時間を使わず、また堅実に、確実に100%へそろえることができる。私たちはこの安定的かつ分かりやすいアルゴリズムを使い、優勝を狙う。

47 演算クラブ

福井

向井 海瑠（4年） 保川 旻（4年）
ナツァグドルジ プヤンヒシグ（4年）
小松 貴大（教員）

1. はじめに

今回の競技内容は、手数が膨大であり最善手を見つけるのが難しいと考えられる。そこで今回はヒューリスティック探索法を用いて問題を解決するプログラムを作成した。

2. 戦略

ボードを最終盤面にするアルゴリズムとして以下のようなものを考えている。

- (1) 1行目をそろえようとしたときに、1行目、2行目、3行目…とそれぞれの行を見て1行目との正答率が何%かを確かめる（「正答率」は位置関係なくそろえたい行にある数字と同じ数字のピースが何%あるかを基準とする）。
- (2) それぞれの行の中で正答率が高いものを選び、評価関数を用いてより正答率が高くなるようにほかの行にあるピースと並び変える（評価関数は行とどれだけ近いかなどを指標とする）。

(3) そして並び替えた行と一行目とを入れ替える（一行目が最も正答率が高いときは入れ替えない）。

(4) 2行目以降も(1)～(3)と同様の操作を行う（揃え終わった行は(2)の際で正答率が高い行として選ばれないようにし、(2)の「正答率が高くなるようにほかの行にあるピースと並び変える」という操作の時も並び終わった行のピースとは並び替えることができないようにする）

3. 開発環境

言語：Python

OS: Windows

統合開発環境：VScode

48 無限即席冷麺

米子

田中 拓弥（3年） 池口 寛大（3年）
徳光 政弘（教員）

1. はじめに

今回の競技では、盤上の情報を効率的に扱うことが求められた。盤の状態を数値として取り扱う必要があり、大量のデータが発生する。そのため、メモリの効率的な使用が重要な課題となった。私たちは、この課題に対応するために「ビットボード」と呼ばれるデータ構造を採用した。

2. 盤の操作

ビットボードを用いることでシフト演算が使用可能になる。シフト演算は、ビット列を左または右にずらす操作であり、ビットボードの操作において重要な役割を果たす。今回の競技では、盤上の情報を効率的に操作するために、シフト演算を活用した。

3. 探索アルゴリズム

ビットボード同様に、乱数を使ってランダムにパラメータを生成し、その中から最適な手を選択するアルゴリズムBeamSearchを採用した。全探索と比較して、計算量を抑えつつ、限られた時間内で結果を得ることが可能である。

2. 開発環境

使用言語：C++/Python

OS：Windows11

IDE：Visual Studio Code

49 文化財修復安全確保支援士

都立
(品川)

石井 一肇 (5 年) 平良 文哉 (3 年)
田中 智之 (3 年) 佐藤 喬 (教員)

1. システム概要

今大会の競技で用いられる盤面は、最大サイズが横 256×縦 256 と大きくなっています。そのため盤面の探索には膨大な候補手数が現れます。したがって候補手数が膨大な問題にも有効なビームサーチを主軸とした方法で、盤面を最終状態に復元する解答を作成します。

さらに解答の復元手順を図として表すビジュアライザを作成します。このビジュアライザを用いて作成した解答が効率的かどうかの評価を行います。また評価結果からビームサーチの評価関数を修正します。

2. 盤面の復元

盤面の復元方法として、手数が多いが計算時間が短い方法と、手数が少ないが計算時間が長い方法の 2 つを用意します。これにより制限時間内に確実に盤面を復元できるようにし、また時間があればより手数の少ない解答を作成する戦法を取ります。以下にそれぞれの方法の概要を示します。

2.1. 左上から 1 セルずつ復元する方法

左上から 1 セルずつ復元する方法では、即座に解答を作成することができます。復元したいセルの位置から最も距離が短いセルを、幅優先探索により発見します。そして定形抜き型を使用することにより、距離の対数に比例した手数で復元位置にセルを移動する解答が作成できます。

2.2. 各行の値の数を揃えてから復元する方法

各行の値の数を揃えてから復元する方法の効果的な点は、盤面の行ごとに独立して探索できる点です。行ごとに独立して探索できるため、問題の縦のセル数 H 、横のセル数 W とした時、評価対象のセル数は HW 個から W 個に減ります。これにより探索の候補手数が減少し、さらに探索時の盤面のコピー時間も短縮されるため、探索をより深く、かつ高速に行うことが可能です。

行ごとに値の個数を揃えた後、ビームサーチを用いて手数が少ない解答を作成します。

3. ビジュアライザ

図 1 に示すようにセルを値ではなく色で表示し、使用者がすぐにセルを判別できるようにしました。また最終状態と一致するセルは専用の色で表示し、揃っていないセルも使用者がすぐに判別できるようにしました。

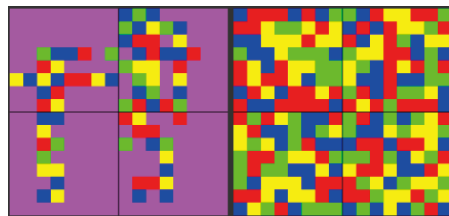


図 1 ビジュアライザ (左が現在盤面、右が最終状態)

50 ホタルイカレジェンズ

富山
(本郷)

安念 翼 (4 年) 西村 旭生 (4 年)
早水 拓海 (4 年) 石田 文彦 (教員)

1. 戦略と工夫点

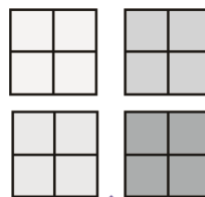
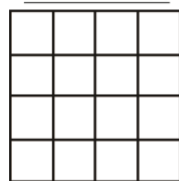
与えられた盤面と最終盤面をおなじ大きさのいくつかの盤面に分割する。そして、その分割した盤面をそれぞれ比較して最終盤面を近似していき、手数を少なくする。

それぞれの盤面を分割して比較することで、一つ一つの盤面の大きさが小さくなり簡単に最終盤面を近似することができる。一度完成した盤面には触れないようにしており、大きい盤面が与えられた時でもより簡単に近似を行うことができる。また、どのような抜き型の縦横比でも、対応し比較することができるようにした。

2. システム

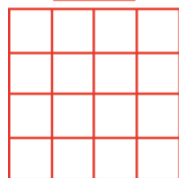
はじめに、スタートとゴールを読み込む。次に、json の型を読み込んで小分けにした盤面同士を数式によって関数のように比較し最終盤面に近似していく。最後に回答を json 形式のデータに記録する。

与えられた盤面



分割盤面同士を比較

最終盤面



3. 開発環境

言語 : python3.12

IDE : Visual Studio Code

OS : Windows 11

51

永久盤上領域
アガロン・ク・レ

呉

江島 朋来 (3 年) 吉岡 雅希 (3 年)
城 ジュニアスプラタマ (5 年)
藤井 敏則 (教員)1. はじめに

今年度の競技部門の内容は条件が特殊な並び替え問題である。指定した領域以外のボードの形も変形させるため、一手に対する盤面の評価の仕方と探索による計算量の削減が、勝負のカギと考えた。

2. 具体的なアルゴリズム

型抜きによるボードの変形の仕方は極めて不規則であり、実際に型抜きをしてしらみつぶしにする以外方法がないのではないかと考える。そうすると計算量は莫大になり実装は現実的ではない。

私たちはそれに対する解決策として、型抜きを組み合わせることによって、そもそもの交換方法を読み替えた。具体的には、ボードの形はそのままに、指定した二つの領域の交換を行うアルゴリズムを実装した。これを用いること

で、局所的な領域の評価のみを行い、探索の計算量を大幅に削減した。しかし純粋な型抜きによる最適解よりも交換手順が数倍多いので、その点が憂慮する部分であるだろう。

3. GUI について

問題の処理を行うプログラムを子プロセスで実行させ、その出力を express によるサーバにて受け取り、CSS 及び JavaScript でスタイリングした HTML (EJS) をレンダリングするという流れで GUI を実装した。

処理中はその進行に合わせて Ajax によるリアルタイムな演出を行い、終了時には処理の詳細結果を確認できる。

4. 開発環境

[使用言語] JavaScript、HTML、CSS

[実行環境] Node.js 20.12.2、npm 10.5.0

52

雄峰寮焼きなまし部

木更津

内藤 正浩 (4 年) 越智 優真 (3 年)
星 凌誠 (3 年) 米村 恵一 (教員)1. 導入

まず、手数を考慮せず、単に最終盤面に一致させることを考える。この場合は、サイズが 1 の定型抜き型を適切に用いて 1 ピースずつ揃えていくことが可能である。

次に、最短手数で一致させることを考える。今回の競技は、盤面のサイズが最大で 256×256 と大きいことに加えて、操作の選択肢も多い。そこで、我々は焼きなまし法を用いた手法によって手数の少ない粗い解を見つけ、その解を初期状態とした 1 ピースずつ揃えるアルゴリズムで最終盤面に一致させるアプローチを採用する。

2. 盤面の評価について

我々の解法の本質的なポイントは、焼きなまし法における適切な目的関数の設定である。

ナイーブな目的関数として、最終盤面と一致しているピースの個数を用いることが考えられる。しかし、これは目

的関数として離散的であるので、最適化しづらい。

より滑らかな目的関数とするため、行や列単位での一致度を考慮するなどの工夫を行う。

3. 高速化について

焼きなまし法では、試行回数を増やすことが重要であるため、操作の高速化が必要になる。ピースに割り当てられる数字が 0 から 3 の 4 種類のみであることに着目し、ビットボード高速化を用いる。これは盤面に対する操作を bit 演算により高速に処理する手法である。

また、適切に並列化を行うことで試行回数を増加させる。

4. 開発環境について

使用言語: C++, Python

エディタ: Vim

1. 概要

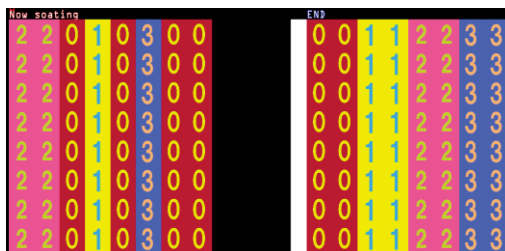
盤面をブロックごとに分けてから個別にそろえていくことで、確実にそろえられる手順をまず発見し、そこから探索範囲を広げてより少ない手数を探索していく。

2. 実装方法

4方向を探索するのではなく挿入方向を2方向に限定することで、探索範囲を狭める。ボードの右下端からブロックに区切っていく、最初は、すでにそろっているセルへの影響が大きくてもより完成盤面に近い抜き型を選択し、概ねそろってきたら、すでにそろっているセルへの影響が小さい抜き型を優先的に選択していく。1つ完成させることができる手順を見つけることができれば、条件を緩めてさらに少ない手数を探索する。

3. ビジュアライザ

探索中のボードの状態と完成状態のボードを確認できるようなビジュアライザを作成し、戦略を考える足掛かりとなるようにした。



4. 開発環境

使用言語: Rust, Python, C++

エディタ: Visual Studio, Visual Studio Code

ライブラリ: DxDLib

1. 概要

試合で有効となる回答は最後に受理された回答であることと、勝敗判定に優先順位があることを考慮して、2つのアルゴリズムを用いて試合を進める。

2. アルゴリズムについて

試合では、まずボードを一致させるアルゴリズムで回答を行い、次に手数を減らすアルゴリズムで回答を行う。

2.1 ボードを一致させるアルゴリズム

このアルゴリズムは手数を考慮せず、パズルの一致だけを目的としたものである。その手順は以下の通りである。

- (1) 1マスに注目し (以下、注目マスと呼ぶ)、同じ座標の現在の状態と最終状態の値が一致すれば次のマスに移動する。一致しなければ、(2) ~ (4) の手順に進む。
- (2) 注目マスに最も近い最終状態の値を持つマス (以下、対象マスと呼ぶ) の位置を求め、その x 座標と y 座標を保持する。
- (3) サイズ 1 の抜き型を使って、注目マスの x 座標と対象マスの x 座標が一致するまで、対象マスを移動させていく。
- (4) 同様に注目マスの y 座標と対象マスの y 座標が一致するまで、対象マスを上に移動させていく。
- (5) 最後の 1 マスになるまで (1) ~ (4) を繰り返す。

2.2 手数を減らすアルゴリズム

このアルゴリズムは手数を減らしつつ、パズルを一致させることを目的としたものである。この目的の実現のために A* アルゴリズムを適用する。ここで、A* アルゴ

リズムとは、与えられたスタート地点からゴール地点までの最適なルート探索のためのアルゴリズムである。

本プログラムでは、ボードの初期状態から現在の状態までの手数と、現在の状態から最終状態までの予測距離によって計算されるコストを設定し、コストが最小となるルートを探索していくことで最適なルートを求める。

3. GUI について

使用するアルゴリズムの切り替えをスムーズに行うため、図 1 のような GUI を用いる。



図 1 開発する GUI

[OS] Windows11

[言語] Visual C++, Siv3D

[IDE] Visual Studio

55 バナナより犬派

広島商船

東方田 匠真 (2年) 河原田 良明 (2年)
正宗 一康 (3年) 岩切 裕哉 (教員)

1. はじめに

今回私たちは強化学習を用いたシステムで、学習プロセスを効率化し、尚且つ安定したボードの揃え方をするような戦略を立てました。

2. 戦略

今回このゲームでは、最初に書いたように安定性を一番重視し、ボードを揃えていく事にしています。そのため、以下のような事に注意してボードを揃えていきます。

- ・値を合致させる順序の工夫: (特定の順番でマス揃えていくことで、値を合致させるのを出来るだけ簡単にしています。)
- ・報酬設計: (合致したマスによって報酬の与え方を変え、ボード全体の合致度に応じた評価で安全に目標達成を促せるような報酬の与え方をする。)

その他にも最適方策を減らし学習効率を上げるために、抜き型の使い方に制約を付けるなどし、問題を解決していく方法なども考えました。

これらの戦略によって、強化学習の効率性と安定性を両立させ、目的達成に向けた確実な進行を目指しています。

3. 意気込み

今回初めて Python や強化学習に触れてみたりし、慣れないことや、分からないことが沢山ありました。その為これまでの努力が無駄にならないように、最後まで楽しみます。

4. 開発環境

使用言語: Python

OS: Windows

56 よねぴーなっつ

有明

内田 陽太 (4年) 栗山 大樹 (4年)
米村 慶太 (4年) 森山 英明 (教員)

1. システムの概要

本システムでは、より短い手数でボードを完成させるための解答を決定し、出力する。これを実現するため、ボードを完成させることができるアルゴリズム (2. で後述) を用い、さらに探索アルゴリズムの 1 つであるビームサーチを用いてより短い手数の解答を探索する。

2. ボードを完成させるアルゴリズム

ボードを完成させるうえで、ピースをボードの端から順に揃えていく方法を採用する。このとき、型抜きの方法を工夫することで、一度揃えたピースを再び移動することなく、ピースを揃えていくことができる。例を図 1 に示す。図 1 ではボードの左上の端を起点にして①～⑧の順にピースを揃えている。その次に揃える座標 d は以下の手順で揃えることができる。Ⅰ) d に移動させたいピースを持つ座標 s_1 を決定する。Ⅱ) 右方向に型抜きを行い、 s_1 のピース

を d と同じ列である s_2 まで移動させる。Ⅲ) 上方方向に型抜きを行い、 s_2 から d まで移動させる。

このとき、型抜き後の下寄せと一度揃えたピースの座標への型抜きを行わないことで、一度

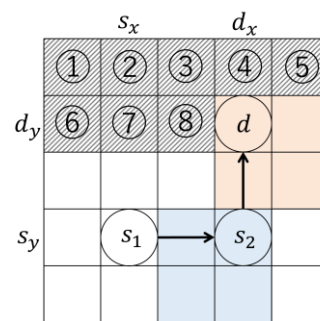


図 1: アルゴリズムの概要

揃えたピースを移動することなくピースを揃えることができる。

3. ビームサーチによる最適化

前述した方法でボードを完成させることはできるが、より少ない手数でボードを完成させる場合、 s_1 の選び方や s_1 から d までの型抜き方法を最適化する必要がある。この最適化には、ビームサーチを用いた探索が有効である。

57 "; DROP TABLE teams; --"

都立
(荒川)

高野 陽大 (5 年)
豊田 アディール (5 年)
三澤 進生 (5 年) 鈴木 弘 (教員)

1. はじめに

任意のピースを交換する二点交換を中心に、盤面を揃えるアルゴリズムを構築した。

2. アルゴリズム

2.1 二点交換

任意の二つのピースを交換するアルゴリズム。交換方法は二つのピースを囲める最小の範囲を盤面の角に移動して、抜き型が盤面からはみ出ても型抜き可能である性質を用いることにより、ピースの交換を行う。

2.2 その他のアルゴリズム

その他二点交換を行う際に手数をなるべく消費しないための補助的なアルゴリズム。

3. 実装方法

Python でアルゴリズムを実装し、React を用いて Web 上で盤面が揃っていく様子を可視化するツールを作成、アル

ゴリズムの検討に活用した。以下の図 1 に様子を示す。

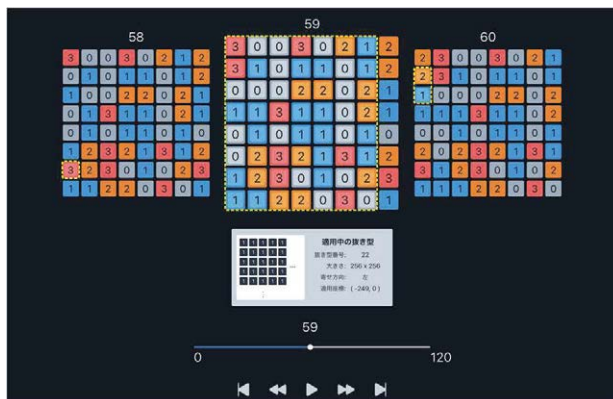


図 1. 盤面の進行の可視化

4. 開発環境

OS: Windows 11, macOS, EndeavourOS, Debian, Ubuntu

Language: Python, TypeScript

Library: NumPy(Python), React(TypeScript)

58 寿司が好きなコンピュータ、何か知ってる? 「データ巻き」ってね

神戸市立

杉本 遼羽源 (3 年) 明 幸音 (2 年)
北尾 匠望 (2 年) 朝倉 義裕 (教員)

1. はじめに

今回の競技部門では、少ない手数で盤面を一致させることが重要となってくる。弊チームは、縦横の移動を分離し、効率的に盤面を操作するアルゴリズムを考案したこのアプローチにより、計算量を大幅に削減することが可能になると考えた。

2. アルゴリズム

まず初めに、最終状態におけるそれぞれの行を構成しているピースを、順番を無視し種類ごとの数のみを揃える。その後、列方向に並べ替えて順番を揃えることで最終盤面とする。この手順により高速な探索が可能となる。

また、行・列ごとに要素を順に揃えるときにおいては、手数が少なくなるような手順となるよう工夫をした。ここでは、行の要素を最終盤面の下から順に見ていく時を

例に挙げる。まず、今見ている行の要素をボードの一番下の行に揃える。このとき、一番下の行に揃えればよいので、揃えるのに必要な要素を抜き型の一番下側に合わせ、上方向に寄せると簡単に揃えることができる。その後、定型抜き型 22 番のような広範囲の抜き型を一番左下の座標に使用し下方向に寄せることで、揃えた行を順に上に積んでいくという方法である。

3. UI 関係

React を用いて現在のボードの状態や手数、一致セル数などの表示を行う。

4. 開発環境

言語: TypeScript

エディタ: Visual Studio Code, IntelliJ IDEA Ultimate

1. Introduction

The goal of the competition is to create a computer program that can restore a puzzle board by strategically applying die cuts to rearrange puzzle pieces. The objective is to efficiently achieve the target puzzle pattern with the minimum number of moves.

2. Overview of the Development

The program is built using the Python language. It uses the Tkinter GUI toolkit and the Matplotlib library for numerical mathematics plotting. The program consists of two parts: the GUI and the Algorithms. The Algorithms part solves the problems we input, and the GUI allow users choose the best algorithms solution approach.

2.1 Graphic user interface (GUI)

The user interface that shows the question puzzle, the target puzzle pattern, and the board's information received from server.

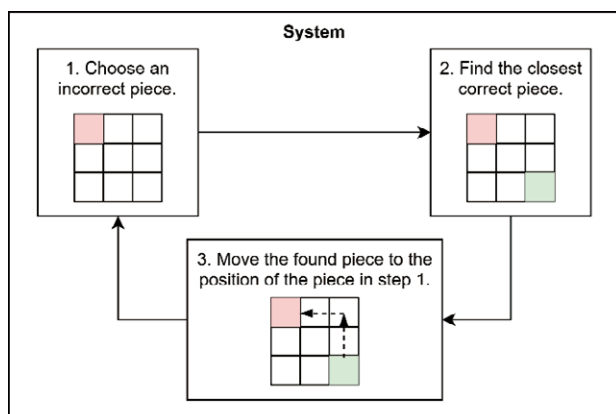
It displays the number of steps taken by different algorithms. This allows users to evaluate and choose the best answer. This comparison will enable user to identify the most suitable die cuts for each specific scenario, ensuring algorithms are effective and efficient.

2.2 Restores puzzle Algorithms

This method involves a systematic, step-by-step process where the board is adjusted sequentially from the top row to the bottom, and from left to right within each row. Applying die cuts of size one to make precise alterations to individual bits, ensuring that each piece of the board aligns perfectly with the designated final structure. An advanced search algorithm that is specifically designed to optimize move efficiency by larger-sized die cuts.

1. Overview

The following figure summarizes our approach to the competition problem, which consists of three main phases. Each phase corresponds to a module in our system. The key idea is iterating through incorrect pieces to restore each piece one by one while keeping the iterated region intact.



This approach focuses on ensuring that there are no pieces in disagreement with the goal board.

2. Details

2.1 Choose an incorrect piece

We traverse the start board row by row. For each row, we iterate through the pieces and if a piece is incorrect, we proceed with the next step.

2.2 Find the closest piece with the correct value

We use breath-first search (BFS) algorithm from the chosen location and stop right after finding a piece with the correct value.

2.3 Move a piece to the desired location

To move a piece to an arbitrary location, we move vertically and then horizontally. Moving one step can be done by placing 1x1 dies next to the piece in the desired direction. To move more than one step, we apply the 1-step move multiple times.

61 XStar

新モンゴル
高専

Tselmeg Tamir
Dulguun Zolzaya
Kyungchan Kim
Soronzonbold Nachin (教員)

1. Introduction

Our system calculates the shortest path from the initial state to the final state using a heuristic search.

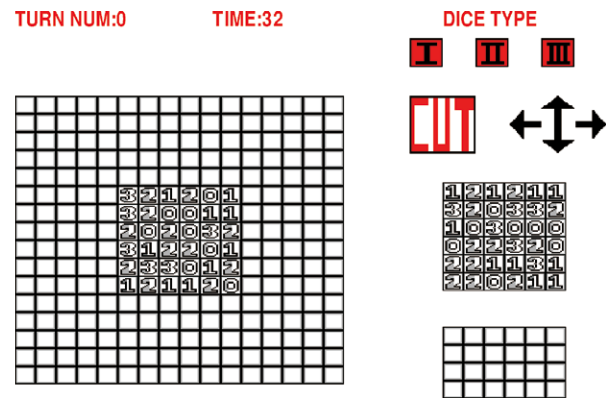
2. Overview of the Development

2.1 Algorithm overview

We use the A* search algorithm, which leverages the heuristic estimate $h(s, g)$ of the remaining cost to guide the search towards the goal. We also implement the reverse action to search from both ends. The DAVI reinforcement learning algorithm trains both the heuristic function and the action value function $Q(s, g, a)$, using a Deep Neural Network as the approximator. The heuristic function reduces search depth, while the action value function reduces the number of actions considered, optimizing both the depth and width of the search process.

2.2 GUI

We built a GUI to visually grasp the current state of the board.



3. Used tools

Language: C++

Libraries and Frameworks: TensorFlow, OpenSiv3D, nlohmann/json, libcurl

62 Ludwig van Beethoven

タイ高専

Phubadee Baimiden
Pranitan Chantawong
Saunghninpwint Oo (教員)

Introduction

Nara Prefecture is where consists of many World Heritages in Japan. This year's competition theme has been designed to be "World Heritage Revival". The board consisted of several pieces randomly and a smaller board called "Die". The idea of this competition is to "restore" the board to the ideal(original) form by using process called "Die-Cutting". This is our assignment. We need to find the shortest path to "restore" the board optimally.

1. Developing Environment

Software: Visual Studio Code, Postman

Devices: Two Laptops (Main and Back Up)

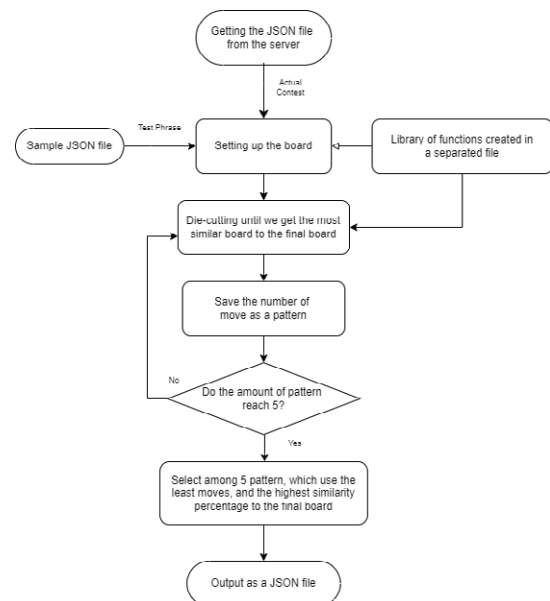
2. Program

We separated the program into 3 files

- 1) Main File: Getting the JSON file and running the program.
- 2) Board & Die Setup: Setting functions for the main file.

- 3) Sample JSON file: Use for testing the program.

4. Process of the program



1. Introduction

This project involves solving a die-cutting puzzle where the goal is to apply a series of predefined dies to transform an initial board into a final target configuration. The challenge lies in optimizing the sequence of moves using the available 25 dies.

2. Overview of the Development

Our team explored various algorithms and experimented with reinforcement learning to solve the die-cutting puzzle. Through the use of advanced RL techniques and iterative learning, we significantly improved the performance and accuracy of the algorithm in reaching the desired board configuration with fewer moves.

2.1 Algorithm and Reinforcement Learning:

We chose Reinforcement Learning (RL) because it allows an agent to learn strategies by interacting with the board and

receiving feedback (rewards) based on its moves. Unlike supervised learning, RL doesn't require labeled data and can discover the best sequence of actions through trial and error. It's particularly effective for board games and puzzle-solving where actions lead to long-term consequences. We chose Q-Learning, a model-free RL algorithm, to solve the problem. Q-Learning helps the agent learn which die to use and how to move pieces by evaluating the outcomes of actions. It balances trying new moves (exploration) and using the best-known ones (exploitation) to find the optimal strategy.

2.2 Tools and environment

Language: Python

OS: windows 10

Libraries: NumPy, TensorFlow, OpenAI Gym, Matplotlib

Environments: Anaconda3, Google colab

協賛企業広告一覧

第35回全国高等専門学校プログラミングコンテストでは、全国の企業より多くのご支援をいただきました。衷心より厚くお礼申し上げます。

(敬称は省略させていただきました。なお、数字は広告掲載ページです。)

アバナード株式会社	110	NTTコム エンジニアリング株式会社	166
株式会社NSD	112	国立研究開発法人理化学研究所 計算科学研究センター	168
株式会社セゾンテクノロジー	114	株式会社インテリジェント ウェイブ	170
株式会社日立製作所	116	株式会社ワコム	171
ピクシブ株式会社	118	アイフォーコム株式会社	172
チームラボ株式会社	120	富士ソフト株式会社	173
株式会社シーエーシー	122	アイ・システム株式会社	174
株式会社トヨタシステムズ	124	オーブンテクノロジー株式会社	175
さくらインターネット株式会社	126	TDCソフト株式会社	176
ネクストウェア株式会社	128	株式会社タブチ	177
株式会社FIXER	130	株式会社ワイズ	178
株式会社ブロードリーフ	132	株式会社クレスコ・ネクシオ	179
株式会社jig.jp	134	有限会社ユニバーサル・シェル・プログラミング研究所	180
株式会社NTTデータ フィナンシャルテクノロジー	136	株式会社CRI・ミドルウェア	181
株式会社アイ・エス・ビー	138	株式会社アイスタイル	182
アドバンスドプランニング株式会社	140	株式会社TOKIUM	183
株式会社インフォコム西日本	142	株式会社日本総合研究所	184
株式会社サイエンスアーツ	144	株式会社朝日ネット	185
合同会社DMM.com	146	株式会社オブティム	186
株式会社クレスコ	148	株式会社RKKCS	187
コスモリサーチ株式会社	150	株式会社TwoGate	188
株式会社デンソー	152	国立研究開発法人 情報通信研究機構	189
株式会社インフォコム東日本	154	株式会社ネットスプリング	190
株式会社サイバーエージェント	156	株式会社亀山電機	191
KDDIアジャイル開発センター株式会社	158	パナソニック インフォメーションシステムズ株式会社	192
株式会社アクセスネット	160	スパイラルセンス株式会社	193
株式会社ビッツ	162	株式会社ファインディックス	194
CompTIA 日本支部	164	株式会社電波新聞社	194

大会委員・プロコン委員・主管校実行委員・事務局等

大会役員

大会会長	一般社団法人	全国高等専門学校連合会会長	大塚 友彦	銅路工業高等専門学校長
副会長	一般社団法人	全国高等専門学校連合会副会長	東 健司	大阪公立大学工業高等専門学校長
副会長	一般社団法人	全国高等専門学校連合会副会長	小島 知博	サレジオ工業高等専門学校長
副会長	一般社団法人	全国高等専門学校連合会副会長	上田 悦子	鹿児島工業高等専門学校長
副会長	特定非営利活動法人	高専プロコン交流育成協会理事長	桑原 裕史	鈴鹿工業高等専門学校名誉教授、都城工業高等専門学校名誉教授
副会長	第36回大会次期主管校校長		和田 清	松江工業高等専門学校長
副会長	第35回大会主管校校長		近藤 科江	奈良工業高等専門学校長

プログラミングコンテスト実行委員会

委員長	近藤 科江	奈良高専	校長
副委員長	飯田 賢一	奈良高専	電子制御工学科（学生主事）教授
副委員長	尋木 信一	有明高専	創造工学科 教授
ブロック委員	三上 剛	苫小牧高専	創造工学科 教授
ブロック委員	千田 栄幸	一関高専	未来創造工学科 教授
ブロック委員	吉成 偉久	茨城高専	国際創造工学科 准教授
ブロック委員	小村良太郎	石川高専	電子情報工学科 教授
ブロック委員	井上 泰仁	舞鶴高専	電気情報工学科 准教授
ブロック委員	重本 昌也	大島商船高専	情報工学科 助教
ブロック委員	岡本 浩行	阿南高専	創造技術工学科 教授
ブロック委員	松野 良信	有明高専	創造工学科 教授
企業委員	久保 慎一	ネクストウェア株式会社	
企業委員	田中 達彦	Microsoft Corporation	
企業委員	伊藤 陽	LINE ヤフー株式会社	
企業委員	飯岡あゆみ	株式会社インテリジェント ウェイブ	
企業委員	伊藤 翼	株式会社 FIXER	
企業委員	リン メイチュ	チームラボ株式会社	
専門委員	伊藤 祥一	長野高専	工学科 教授
専門委員	長尾 和彦	弓削商船高専	情報工学科 教授
専門委員	小村良太郎	石川高専	電子情報工学科 教授〔兼任〕
専門委員	田中 寛	都立（品川）高専	ものづくり工学科 准教授
専門委員	小山 慎哉	函館高専	生産システム工学科 教授
専門委員	臼井 昇太	都城高専	電気情報工学科 教授
専門委員	森本 真理	秋田高専	共通教育系 准教授
専門委員	三上 剛	苫小牧高専	創造工学科 教授〔兼任〕
専門委員	吉成 偉久	茨城高専	国際創造工学科 准教授〔兼任〕
専門委員	布施川秀紀	群馬高専	電子メディア工学科 准教授
専門委員	中井 一文	鳥羽商船高専	情報機械システム工学科 准教授
専門委員	田中 諒	久留米高専	制御情報工学科 准教授
専門委員	寺元 貴幸	津山高専	総合理工学科 教授
専門委員	小保方幸次	一関高専	未来創造工学科 教授
専門委員	佐藤 秀一	長岡高専	一般教育科 教授
専門委員	松野 良信	有明高専	創造工学科 教授〔兼任〕
専門委員	黒木 祥光	久留米高専	制御情報工学科 教授
専門委員	出江 幸重	鳥羽商船高専	情報機械システム工学科 教授
専門委員	井上 泰仁	舞鶴高専	電気情報工学科 准教授〔兼任〕
専門委員	川本 真一	群馬高専	電子情報工学科 准教授
専門委員	重本 昌也	大島商船高専	情報工学科 助教〔兼任〕
専門委員	鈴木 宏	長野高専	工学科 嘱託教授
専門委員	松林 勝志	東京高専	情報工学科 教授
専門委員	田添 丈博	鈴鹿高専	電子情報工学科 教授
専門委員	小嶋 徹也	東京高専	情報工学科 教授
専門委員	都築 啓太	豊田高専	情報工学科 准教授
専門委員	山下 晃弘	東京高専	情報工学科 准教授
専門委員	サブコタ アチュタ	木更津高専	情報工学科 教授
専門委員	岡本 浩行	阿南高専	創造技術工学科 教授〔兼任〕
専門委員	千田 栄幸	一関高専	未来創造工学科 教授
専門委員	江崎 修央	鳥羽商船高専	情報機械システム工学科 教授
主管校実務委員	山口 賢一	奈良高専	情報工学科 教授
主管校実務委員	松尾 賢一	奈良高専	情報工学科 教授
主管校実務委員	岩田 大志	奈良高専	情報工学科 准教授
主管校実務委員	本間 啓造	奈良高専	情報工学科 准教授
主管校実務委員	高吉 康弘	奈良高専	学生課長
主管校実務委員	澤田 彩	奈良高専	学生係長
主管校実務委員	松岡 俊男	奈良高専	学生係主任
主管校実務委員	志方 優太	奈良高専	学生係
情報系有識者委員	奥田 遼介		
	福永 修一	広島工業大学	工学部 准教授
	大城 泰平	北海道大学	創成研究機構 特任准教授
次年度主管校委員	渡部 徹	松江高専	情報工学科 教授
次々年度主管校委員	重田 和弘	香川高専（高松）	電気情報工学科 教授
前年度主管校委員	斉藤 徹	福井高専	電子情報工学科 教授

主管校実行委員会（奈良工業高等専門学校）

委員長	近藤 科江（校長）	会場準備	山口 智浩
副委員長	飯田 賢一（学生主事 電子制御工学科 教授）	課題・自由部門	松尾 賢一、本間 啓造
事務局長	山口 賢一（情報工学科 教授）	競技部門	岩田 大志
総務	山口 賢一	学校PR・広報イベント	藤田 直幸、玉木 隆幸
受付・案内	鷗野 晃弘	企業ブース	中村 秀美
式典	高吉 康弘	学生交流会	飯田 賢一
輸送・弁当・警備	笹山 智仁	情報交換会	高吉 康弘
広報・記録	酒井 史敏	電源・ネットワーク	岡村 真吾
会計	澤田 彩		

大会事務局・委員会事務局

〒639-1080 奈良県大和郡山市矢田町 22
第35回全国高等専門学校プログラミングコンテスト事務局 奈良工業高等専門学校 学生課学生係
TEL: 0743-55-6034 / FAX: 0743-55-6039 / E-mail: jimu35@procon.gr.jp

高専プロコン交流育成協会（NAPROCK）事務局

〒541-0057 大阪市中央区北久宝寺町 4-3-11 ネクストウェアビル
TEL：06-6281-0307 / FAX：06-6281-0318
担当 事務局 木戸 能史 NAPROCK 事務局長
渡邊 博和 NAPROCK 事務局次長

【前回大会の様子】

第34回全国高等専門学校 プログラミングコンテスト

令和5年10月14日土 ~ 10月15日日

主管校 福井工業高等専門学校

会場 サンドーム福井



第35回全国高等専門学校 プログラミングコンテスト

日時 令和6年10月19日土~20日日

会場 なら100年会館 奈良市三条宮前町7番1号

お問合せ

第35回全国高等専門学校プログラミングコンテスト事務局 奈良工業高等専門学校 学生課学生係

TEL 0743-55-6034 E-mail jimu35@procon.gr.jp