

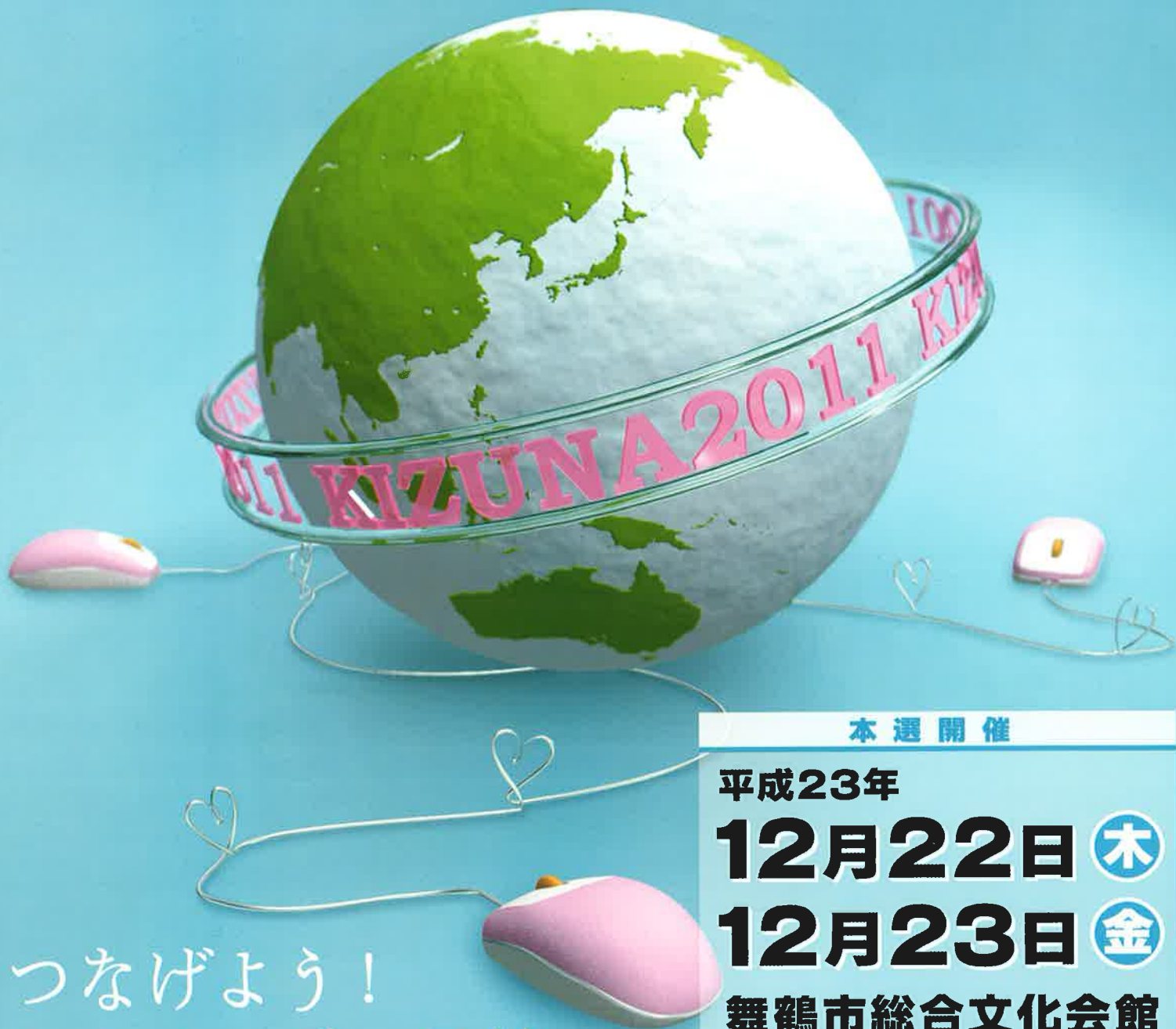
PRO

全国高等専門学校 第22回

プログラミングコンテスト

NAPROCK 3rd International Programming Contest

KIZUNA2011



つなげよう！
テクノロジーの絆。

本選開催

平成23年

12月22日 木

12月23日 金

舞鶴市総合文化会館

—京都府舞鶴市浜 2021 番地—

課題・自由部門

◎プレゼンテーション (概要発表)

日時 12月22日(木) 10:10～17:00

会場 課題部門：2階会議室

自由部門：小ホール

◎デモンストレーション (実演)

日時 12月22日(木) 10:40～17:00

12月23日(金) 9:00～14:00

会場 課題部門：2階ホワイエ

自由部門：展示室

競技部門

日時 12月22日(木) 14:00～17:30

12月23日(金) 8:30～14:00

会場 大ホール

プロコン公式サイト：<http://www.procon.gr.jp/>



国立舞鶴工業高等専門学校 (主管校)

- 主催** 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
舞鶴市
- 後援** 文部科学省、京都府、京都府教育委員会、舞鶴市教育委員会、舞鶴商工会議所、
社団法人コンピュータソフトウェア協会、一般社団法人情報処理学会、
社団法人電子情報通信学会、教育システム情報学会、独立行政法人情報通信研究機構、
NHK、舞鶴工業高等専門学校同窓会、舞鶴工業高等専門学校後援会

特別協賛 さくらインターネット(株)、東芝ソリューション(株)、(株)インテリジェントウェイブ、
(株)サプライ、チームラボ(株)、東京エレクトロン ソフトウェア・テクノロジーズ(株)、
(株)トヨタコミュニケーションシステム、日本マイクロソフト(株)、ネクストウェア(株)、
富士通(株)、(株)ブロードリーフ

一般協賛 アイフォーコム(株)、(株)NTTデータ、(株)オプティム、(株)サミーネットワークス、
セイコーエプソン(株)、トヨタテクニカルディベロップメント(株)、日本SGI(株)、
日本電気(株)、(株)BCN、メガソフト(株)、弥生(株)、(株)ワコム

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 「旅とコンピュータ」
2. 自由部門
3. 競技部門 「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）

応募期間 平成23年6月24日（金）～7月1日（金）

- 審査**
1. 予選（書類による審査）
 - 日時 平成23年7月30日（土）
 - 場所 舞鶴市総合文化会館（京都府舞鶴市浜2021番地）
 2. 本選（プレゼン・デモ等による審査、競技は対抗戦）
 - 日時 平成23年12月22日（木）～12月23日（金）
 - 会場 舞鶴市総合文化会館（京都府舞鶴市浜2021番地）

表彰 次の賞を授与します。

課題・自由部門

- 最優秀賞……………各1点（賞状及び副賞）
- 優秀賞……………各1点（賞状及び副賞）
- 特別賞……………各数点（賞状及び副賞）

※最優秀賞には、文部科学大臣賞、情報処理学会若手奨励賞が授与される。

競技部門

- 優勝……………1点（賞状及び副賞）
- 準優勝……………1点（賞状及び副賞）
- 第3位……………1点（賞状及び副賞）
- 特別賞……………数点（賞状及び副賞）

※優勝には、文部科学大臣賞、情報処理学会若手奨励賞が授与される。

ポスターデザイン 福井工業高等専門学校卒業 村田 真隆

※岩手県一関市での開催予定でしたが、震災のため、開催場所及び日程が変更になりました。

Nourishment Association for Programming Contest KOSEN, non-profit organization

NAPROCK

3rd INTERNATIONAL PROGRAMMING CONTEST

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. In this year, the NAPROCK 3rd international programming contest is held with the 22nd Kosen Procon. This contest aims to promote flexibility of thinking through programming, at a very high level. At the contest, students from kosens 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:** December 22nd (Thu.) – 23rd (Fri.), 2011

• **VENUE:** Maizuru City Culture Center (2021, Hama, Maizuru-shi, Kyoto-fu, Japan)

• **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 in a progressive tournament for victory.

• **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

The following prizes are awarded in the Competition Section.

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

PROCON WEB SITE:

NAPROCK Procon official site: <http://naprock.jp/IntProcon/>

Kosen-Procon official site: <http://www.procon.gr.jp/>

SPONSORS and SECRETARIAT:

Main sponsor: Nourishment Association for Programming Contest KOSEN

Co-sponsor: Technical College Association

Supporters: Nextware Ltd.

Toshiba Solutions Corporation

Other companies contributed to Kosen Procon support to this contest.

Supervising college: Maizuru National College of Technology

Judging Committee: Kosen Procon judging committee

Foreign Participants: Univ. of Engineering and Technology, Vietnam National University, Vietnam

Mongolian University of Science and Technology, Mongolia

Chengdu Neusoft Vocational Institute of Information Technology, China

King Mongkut's Institute of Technology, Thailand

National United University, Taiwan

Secretariat: NAPROCK

ご挨拶

大会会長挨拶

高等専門学校連合会会長
鹿児島工業高等専門学校長

赤坂 裕



第22回全国高等専門学校プログラミングコンテスト（高専プロコン）の本戦を、舞鶴市総合文化会館を会場として開催することになりました。

第22回高専プロコンは、一関高専が昨年来開催の準備を進めてきました。しかし、平成23年3月11日の東北地方太平洋沖地震により一関での開催が不可能となり、急遽、舞鶴高専に主幹校を担当していただくことになりました。快くお引き受けいただいた舞鶴高専の教職員の皆様、ありがとうございます。準備を進めながら中途で開催を断念せざるを得なかった一関高専の教職員の皆様、ご苦労様でした。

高等専門学校（高専）は、中学卒業後5年一貫の技術者教育を行う高等教育機関として1962年（昭和37年）に発足して以来、実践的・創造的な技術者を世に送り出し、産業界の発展に寄与してきました。時代の動向を見据えながら、専攻科やスーパー高専の設置等の組織改革、JABEEや機関別認証評価等の第三者評価への対応を行うなど、新しい時代に必要とされる人材の育成に邁進しています。

高等専門学校連合会は、高専教育の一層の振興を図るために、全国の国公私立高専のIT技術教育の成果をコンテスト形式で競うものとして、1990年（平成2年）から高専プロコンを開催してきました。この高専プロコンはIT技術に関するアイデアと表現力を競うものとして定着してきており、その独創性・創造性はIT業界や関連学会から高く評価され、今では、高専ロボコン、高専デザコン、高専プレコンと共に、連合会の主要行事に位置付けられています。

第22回高専プロコンは、復興への思いを込め主題を「KIZUNA2011」としました。部門は、例年通り、課題部門、自由部門、競技部門の3部門です。今大会も多数の応募があり、厳正な予選審査を実施して本戦参加チームを決定しました。今日と明日の本戦において、課題部門では「旅とコンピュータ」をテーマに20作品が、自由部門では独創的なアイデアを持つ20作品が、そして競技部門では「よみがえれ、世界遺産」と題した対抗戦に60作品が、アイデアとプログラミングの技術を競います。

ご来場いただいた皆様には、高専の高度で多様な教育内容をIT教育という切り口からご覧いただけることと存じます。高専学生の感性・創造性・技術力に直接触れていただき、高専の技術者教育の素晴らしさや面白さをご理解いただければ幸甚に存じます。

最後になりますが、ご後援いただきました文部科学省、京都府、京都府教育委員会、舞鶴市教育委員会、コンピュータ関連の協会、情報処理関連の学会、報道機関ならびに関係団体、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、高専プロコンを企画・運営されました実行委員会の先生方、高専プロコンに多面的にご支援いただいているNPO法人「高専プロコン交流育成協会（NAPROC）」、主幹校である舞鶴高専の太田校長はじめ教職員の皆様にご心より感謝申し上げます。

特定非営利活動法人 高専プロコン交流育成協会 (NAPROCK) 理事長挨拶

堀内 征治



3.11という歴史的なターニングポイントを刻んだ今年は、全国高専プロコンにとっても大きな試練に立ち向かう年となりました。そのような中、ここ舞鶴市において第22回大会が盛大に開催されますことに大きな感動を禁じえません。プロコンの力強い歩みに大きな期待を寄せるとともに、これまで大会に関わられた多くの皆様に心から敬意を表します。

私どもNPO法人高専プロコン交流育成協会(NAPROCK)は、伝統ある高専プロコンをさらに発展させ、情報産業のグローバル化に対応できる高度なエンジニア育成教育を充実させるために、産学官が連携して、社会に開かれ、かつ、国際化への対応を果たすことを目的に、2008年7月に正式に認可されました。そして、第15回大会(2004年)以来オープン参加で進めてまいりました国際チームの参加を、法人設立の翌年(2009年)からは、プロコンの正式な国際大会「NAPROCK国際大会」として、華やかにスタートさせることができました。本年度は、その第3回大会として、ベトナム、中国、モンゴル、台湾、タイそして日本の若者が集います。過去最多の参加であり、高度な技術を競う一方、幅広い交流を深める中で、充実した国際大会が開催されますことは、主催者として大変嬉しいことと存じております。

グローバル化が進む一方で、世界の経済状況は不透明さを増しております。このような中で、今年度も開催が可能となりましたのも、内外の関係者、ことに、ご協賛いただいた多くの企業の皆様方の並々ならぬご尽力の賜物と厚く御礼申し上げます。また、大会開催地として運営に当たられる舞鶴高専の皆様にも、多大なご配慮をいただきました。併せて、心より感謝を申し上げますとともに、若い方々が国際大会を通じて大いに飛躍されますことを祈念して、ご挨拶といたします。

プロコン委員長挨拶

舞鶴工業高等専門学校長

太田 泰雄



第22回全国高専プログラミングコンテスト大会(高等専門学校連合会主催)が舞鶴市で開催されますことは我々舞鶴高専にとりまして大きな荣誉とするところでございます。

現在、情報関連技術は産業の根幹となる技術であり、今後、訪れるであろう環境社会、スマートグリッド、再生エネルギーなどは、いずれもICT技術抜きには考えられません。そのような次世代技術の開発の中核となるのはICT技術に強い高専卒業生だろうと思います。産業界も高専・高専卒業生に対して熱い期待を持っています。そのためにも毎年開催されるこの大会の役目は大きいと考えられます。また、国際大会も同時に開催され、高専の国際化にも意味があると思われまます。

この大会は当初、一関高専を中心に一関市で10月に開催される予定でしたが、本年、3月に発生した東日本大震災により、京都府で開催されることとなりました。この大会には、被災をされた東北地区の高専生も参加の予定です。全国のICT高専生が山と海に囲まれた美しい舞鶴市に集い、日頃の学習成果を大いに発揮されるようお祈り申し上げます。

最後となりましたが、共催を頂きいろいろご支援・ご鞭撻を賜りました舞鶴市、また、ご後援頂きました皆様、さらに、審査委員、大会運営にご協力を頂きます皆様に重ねて御礼を申し上げます。

大会日程

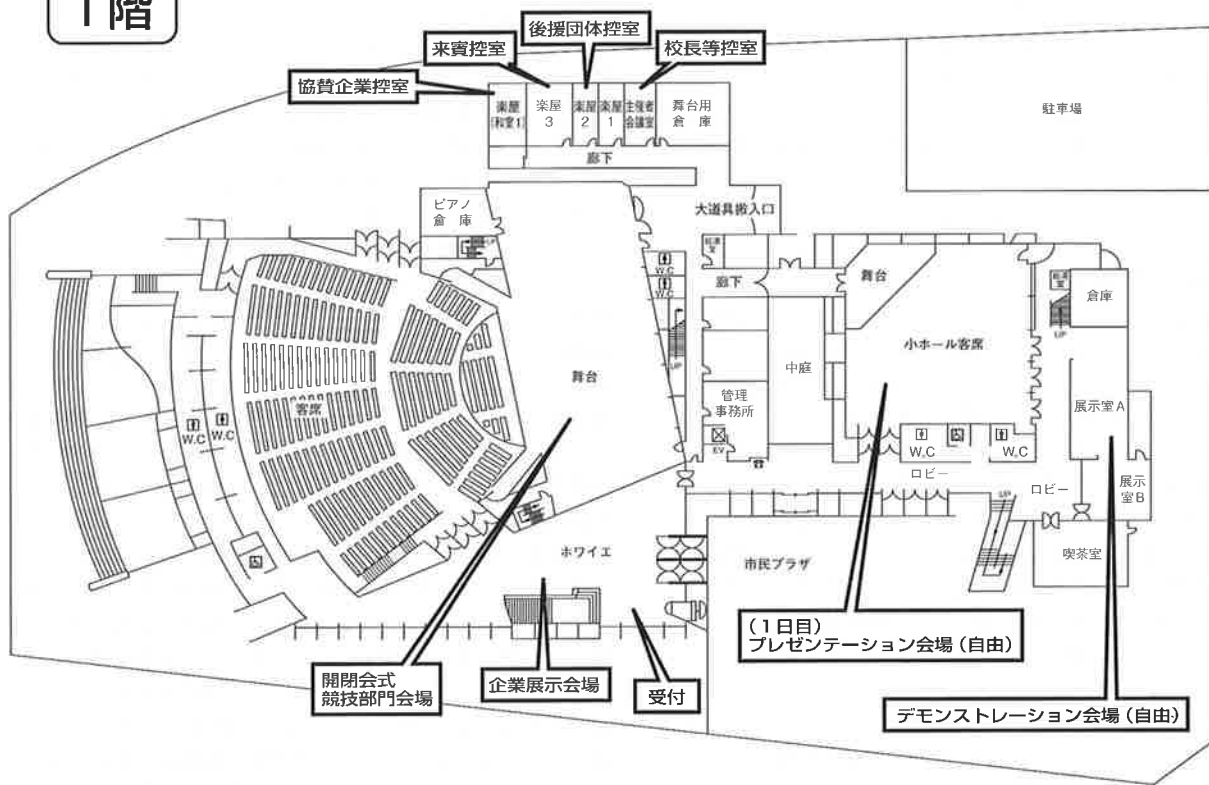
月日	会場	8	9	10	11	12	13	14	15	16	17
12月22日(木)	1階 ホワイエ	参加者 当日受付 8:00-8:30	連絡会議 (課題/自由) 8:30-8:40								
	大ホール		連絡会議 (競技) 8:40-9:00	開会式 9:30-10:00	予行演習(競技) 10:10-13:00			1回戦(競技) 13:00-17:30			
	会議室				プレゼンテーション審査(課題) 10:10-17:00						
	小ホール				プレゼンテーション審査(自由) 10:10-17:00						
	展示室 A・B		システム搬入チェック (課題) 8:40-9:15		システム セッティング (課題)	デモンストレーション一般公開(課題) 10:40-17:00					
	2階 ホワイエ		システム搬入チェック (自由) 8:40-9:15		システム セッティング (自由)	デモンストレーション一般公開(自由) 10:40-17:00					
12月23日(金)	1階 ホワイエ	連絡会議 (課題/自由) 8:00-8:10									
	大ホール	連絡会議 (競技) 8:00-8:20	敗者復活戦・準決勝・決勝(競技) 8:30-14:00						閉会式 14:45-16:00		
	展示室 A・B	システムセッティング (課題) 8:10-8:40	デモンストレーション審査・マニュアル審査(課題) 8:40-12:00			デモンストレーション一般公開(課題) 9:00-14:00			システム 細包		
			デモンストレーション審査・マニュアル審査(自由) 8:40-12:00			デモンストレーション一般公開(自由) 9:00-14:00					
2階 ホワイエ	システムセッティング (自由) 8:10-8:40	デモンストレーション審査・マニュアル審査(自由) 8:40-12:00			デモンストレーション一般公開(自由) 9:00-14:00						

審査委員

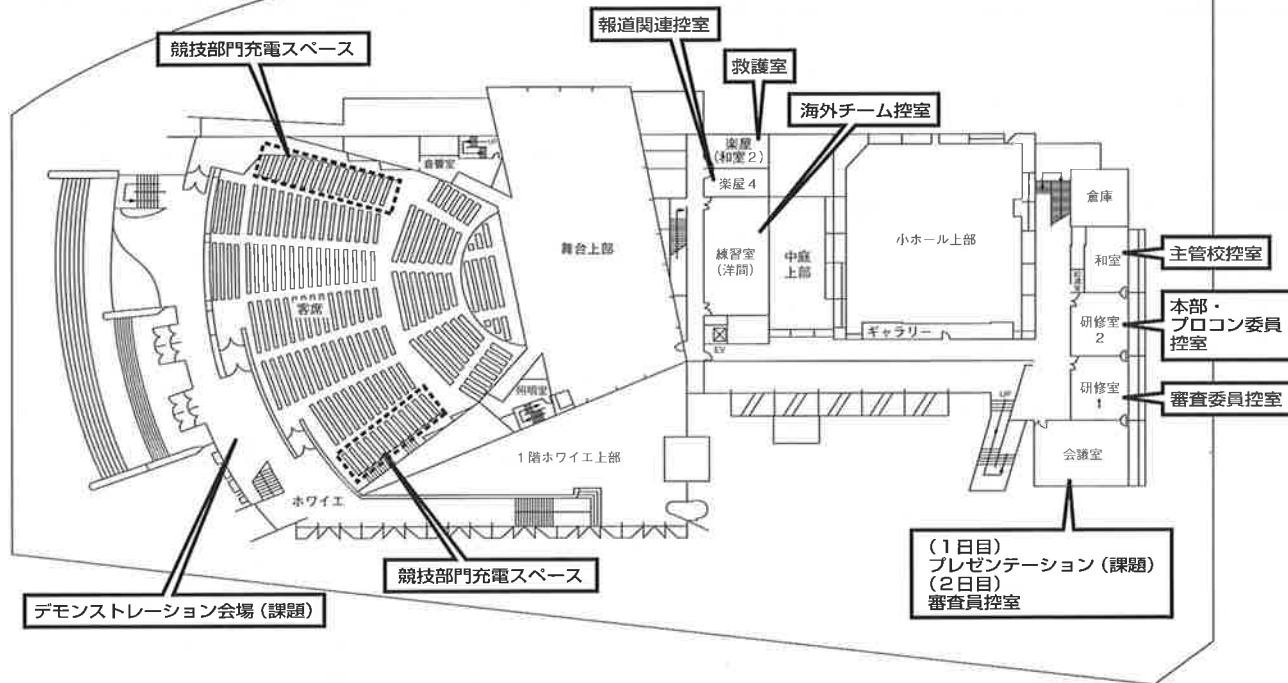
審査委員長	神沼 靖子	情報処理学会フェロー
審査委員	白井 支朗	理化学研究所 脳科学総合研究センター ニューロインフォマティクス技術開発チーム チームリーダー/ 神経情報基盤センター長
	遠藤 直樹	東芝ソリューション(株) 技術統括部 技監
	折居 正広	(株)サプライ 代表取締役 社長
	木下 博行	富士通(株) 文教ソリューション事業本部 シニアディレクター
	嶋田 豊秋	NHK放送センター 放送技術局 報道技術センター ニュース・ネットワーク部 部長
	杉田 泰則	長岡技術科学大学 電気系 助教
	竹俣 信榮	(株)トヨタコミュニケーションシステム ビジネスシステム本部 BS2部 部長
	田村 哲也	チームラボ(株) 取締役
	津田 格	メガソフト(株) 専務取締役
	前川 徹	社団法人 コンピュータソフトウェア協会 専務理事
	松澤 照男	北陸先端科学技術大学院大学 情報社会基盤研究センター 教授
	宮地 力	国立スポーツ科学センター スポーツ科学研究部 副主任研究員
	矢野 剛	(株)ブロードリーフ 事業本部 執行役員 副本部長
	山岡 美之	ネクストウェア(株) 専務取締役
	山本 祥之	(株)インテリジェントウェイブ 代表取締役 社長執行役員
	鷺北 賢	さくらインターネット(株) さくらインターネット研究所 所長
	渡辺 弘之	日本マイクロソフト(株) デベロッパー&プラットフォーム統括本部 アカデミックテクノロジー推進部 アカデミックエバンジェリスト
(マニュアル)	久保 慎一	ネクストウェア(株)
(マニュアル)	津曲 潮	デザイン・クリエイション(株)

会場案内図

1階



2階



プログラミングコンテスト発展の経緯

本コンテスト（俗称プロコン）は今回で22回を迎えます。60高専から応募があり、今回も盛大に開催できます。これも、関係各位のお陰と感謝申し上げます。

プロコン発展の経緯について説明させていただきます。

本コンテストの主催団体は高等専門学校連合会です。この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す。平成元年8月、この委員会（当時は、情報処理教育協議会という名称）の常任委員会で全国の高専の学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願いもありました。

第1回コンテストは、1年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。第1回の成功以後、回を重ねるごとに規模も大きくなり、内容も充実してきています。

当初、課題・自由の2部門でスタートしたコンテストですが、第5回から競技部門を設け、3部門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、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計って参りました。主催団体である連合会も、教育プロジェクトとしてのプロコンの役割を重視し、下部機関としてプログラミングコンテスト委員会を独立に発足させ、プロコンのさらなる充実を図っています。

第2回からは文部省からもご後援を賜り、第4回からは念願の文部科学大臣賞を、第6回からは競技部門を含む全部門で文部科学大臣賞をいただけるようになりました。更に、第18回課題部門最優秀作品が第3回ものづくり日本大賞（内閣総理大臣賞）を受賞しました。プロコンの教育効果に対する評価も各界からいただいています。プロコン委員が、日本工学教育協会から工学教育賞を、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。プロコンは第1回より(株)日本パーソナルコンピュータソフトウェア協会（現(株)コンピュータソフトウェア協会）から後援をしていただき、絶大なご援助をいただいております。第1回は6社からスタート

した協賛ですが、最近では20数社に及ぶ大きなご支援をいただけるようになりました。また、マスコミからもご後援を頂戴しております。このように、本コンテストの趣旨や意義がますます社会的に評価されてきたものと、喜ばしくまた有り難く思っています。更にプロコンを支援する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回本選から共催団体として加わりました。これも、後援団体および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回ではオーストラリア・リッツ大学へ、第10回では韓国への課題部門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、ハノイ工科大学、モンゴル科学技術大学、大連東軟情報学院、モンゴル国立大学、ベトナム国家大学、國立聯合大学(台湾)を本選に迎えています。第20回大会より、国際プログラミングコンテストを同時開催しています。海外チームの招聘には、国際化にご協力いただいている企業およびNPO法人高専プロコン交流育成協会にご支援いただいております。

今年度は舞鶴高専が主管校となり、舞鶴市総合文化会館を会場に本選が開催されます。なお、3月まで主管校をご担当頂いた一関高専には競技システムをはじめ大変お世話になりました。今回は過去最多の163チームの応募があり、予選を通過した課題部門20チーム、自由部門20チーム、競技部門60チーム、海外から6チームの参加で本選が行われます。

コンテスト本選では、いずれの部門も高専生のエネルギーと創造力を皆さんの肌で感じることができると思います。学生の逞しいエネルギーと皆様のご支援を糧として、常に一歩進んだプロコンを目指して努力したいと考えております。

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	一関高専/舞鶴高専	舞鶴市総合文化会館

第21回 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	高専名	指導教員	作成学生
文部科学大臣賞 最優秀賞	べらぐる -VariousLanguagesGourmet-	松江	福岡 久雄	川上 西, 吉村 拓哉, 岩成 達哉, 小川 貴子, 荒川ひかり
優秀賞	タビは道づれ	舞鶴	井上 泰仁	小出 健司, 新保 智暁, 金澤 朋寛, 金田 卓也, 陰山海一郎
特別賞	すれちがいステーション	鳥羽商船	江崎 修央	奥浦 航, 木下 実優, 藤原 正希, 島影 瑞希, 河崎 早苗
特別賞	旅coccolo	東京	松林 勝志	赤松 駿一, 安部 清貴, 大川 水緒, 田畑 愛実, 原 みさと
特別賞	文学探訪 -Literature Navigation-	香川(詫間)	宮武 明義	佐藤 誠, 石田 恵理, 川江 遼太, 原田 晃輔, 宮崎 雄也
特別賞	バーチャリ・トラベラー! -エアロバイクで旅に行こう!-	舞鶴	船木 英岳	辻 春樹, 小林 賢太, 西村健一郎, 高山 凱久, 中川 慎哉

■ 自由部門

賞の名称	タイトル	高専名	指導教員	作成学生
文部科学大臣賞 最優秀賞	MINAMO STYLE -水面流-	香川(詫間)	金澤 啓三	安藤 英里, 合田 貴博, 佐野 純一, 北原 康佑, 十鳥 恭平
優秀賞	ねこみゅ〜	松江	和田守 美穂	中島 千尋, 今田 智大, 大澤 昇, 品川 有輝, 岸野 博明
特別賞	Connect Beat!	長野	伊藤 祥一	矢口 裕也, 齊藤 匠, 小林 将史, 若林 哲宇, 奥本 隼
特別賞	救急Res:Q -携帯端末を用いた音声反応式記録システム-	米子	河野 清尊	秋山 晴彦, 左久間一幸, 柴田 泰樹, 鷺見 知洋, 舟越 大
特別賞	ガリガリはんがリアン -コンピュータではじめる木版画-	徳山	高山 泰博	平櫛 貴章, 三坂 奨, 八木 俊樹, 松本 八希, 重岡 達哉
特別賞	弓道MASTER -理想の射型をアドバイス-	熊本(熊本)	島川 学	星野 涉, 三池 伽奈, 宮崎 靖之, 吉本 聡美

■ 競技部門

賞の名称	タイトル	高専名	指導教員	作成学生
文部科学大臣賞 優勝	インピーダンスZ -アドミタンスの逆数-	石川	小村 良太郎	中嶋 悠介, 藤江 拓哉, 本多 達也
準優勝	大地水浸し計画 -お前はもう、沈んでいる-	松江	橋本 剛	竹下 幸裕, 森山 弘基, 児嶋健太郎
第三位	高専の治水基礎B	長野	鈴木 宏	佐藤 匡, 浅沼 仁, 松井 澄
特別賞	瓶瓶クラブ -亀がいるだけで-	近畿大学	宇田 隆幸	高嶋 大翼, 清原 達成, 宇井 拓臣
特別賞	まったり水撒き	群馬	小幡 常啓	味曾野雅史, 味曾野智礼, 土屋 俊貴
特別賞	水亀 -みずゲーム-	豊田	岡部 直木	小川 拡, 赤川 卓也, 藤井 直道

NAPROCK PROCON2010 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
Grand Prize	べらぐる -VariousLanguagesGourmet-	松江	福岡 久雄	川上 茜, 吉村 拓哉, 岩成 達哉, 小川 貴子, 荒川ひかり
Second Prize	タビは道づれ	舞鶴	井上 泰仁	小出 健司, 新保 智喝, 金澤 朋寛, 金田 卓也, 蔭山海一郎
Special Prize	Easy To Choose (ETC)	ハノイ国家大学	Nguyen Phuong Thai	Pham Phuc Anh, Pham Xuan Truong
Special Prize	Dreamy Wish Wall	大連東軟 信息学院	Zheng Chuniun	Dong Jiehao, Wang Liang
Special Prize	Travel Journal	モンゴル 科学技術大学	KHUDER AltangereI	CHINZORIG Dandarchuluun, MARAL-ERDENE Tumursukh

■ 競技部門

賞の名称	タイトル	学校名	指導教員	作成学生
Champion	インピーダンスZ -アドミタンスの逆数-	石川	小村 良太郎	中嶋 悠介, 藤江 拓哉, 本多 達也
Second Prize	Passion	ハノイ 国家大学	Nguyen Phuong Thai	Hoang Thanh Tung, Nguyen Huu Cuong
Special Prize	Hydrogen & Oxygen Artificial Intelligence	大連東軟 信息学院	Sun Lei	Zheng Min, Sun MingShen

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の作品を募集しています。今大会のテーマは、昨年度に引き続き「旅とコンピュータ」となっています。インターネットを通して、世界中の情報がいながらにして入手できるようになった現在、旅の過程や目的地の魅力、面白さをどのように伝え、体験できるシステムを提案するのか。高専生が感じる「旅」の魅力を提案する新しい作品が期待されます。

今大会では、課題部門に45作品の応募をいただき、7月に舞鶴市で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。これに海外から台湾國立聯合大學を加えた21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) プログラミングリストのチェック

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。)

近年のパソコンの高度化、無線を含むネットワークの普及、携帯型情報端末の多様化により、コンピュータの利用方法が大きく拡大しています。本部門では、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提案された独創的な作品が期待されます。過去の自由部門の優秀作品がIPA未踏ソフトウェアに採択されるなど、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58作品の応募をいただき、7月に舞鶴市で行われた予選審査において、書類選考により20作品が本選に選抜されました。これに海外から台湾國立聯合大學を加えた21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。

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。

今年の競技は、京都や平泉のような優れた文化遺産の復元をモチーフとしており、与えられた画像の復元が目的となります。はじめに1枚の画像がフィールドに示され、複数のスタンプが用意されます。スタンプをフィールドに適用するとスタンプと

一致したフィールドが反転します。うまくスタンプを適用して、フィールドを目標とする最終画像に近づけます。できるだけ早く初期画像を最終画像に「修復」できたチームが勝利する競技です。

●今大会の競技内容

「よみがえれ、世界遺産」

第22回大会の競技は、最大12チームの対戦を行います。競技時間は2～20分とし、1試合に2問出題されます。各チームはネットワークから問題をダウンロードし、最適なスタンプの適応方法を考えます。解答が見つければ、サーバへアップロードします。勝敗は以下のルールで決定します。

- ① 最終画像との一致度が高い方が勝ち。
- ② 修復手数が少ない方が勝ち。
- ③ 最後の回答を早く提出した方が勝ち。
- ④ じゃんけんで勝った方が勝ち。

12チームによる競争ですが、チーム間の干渉はないので各チームのアルゴリズムの優劣がそのまま勝敗に繋がる競技になっています。

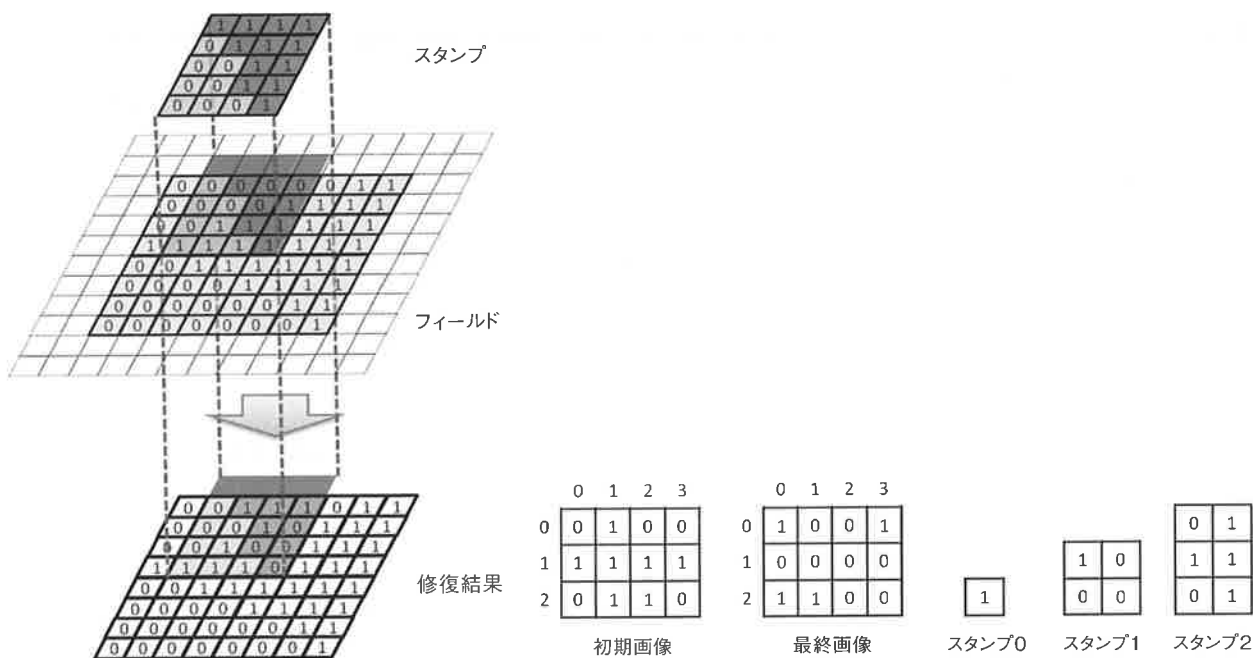


図1 画像復元の原理 (左：スタンプの適応、右：問題例)

課題部門本選参加テーマ

発表順	タイトル	高専名	指導教員	参加学生
1	MAPHIS —歩いて広がる歴史の世界—	松江	福岡 久雄	荒川ひかり, 難波 隼也, 津田 達也, 索手 一平, 内田 譲大
2	たびどあ!	石川	越野 亮	林 真利奈, 孫田 耕作, 川端 優紀, 山田 貴之, 月田小百合
3	とらりす —ノートを書き終わるまでが旅なのだ—	舞鶴	中川 重康	新保 智暁, 小林 賢太, 金澤 朋寛, 高山 凱久, 桐井 良太
4	旅ズバツ!	奈良	岩田 大志	麻田 優真, 上岡 真也, 梶原 雄士, 島岡 大遥, 恒川 凌
5	SnapSwapTrimap —携帯端末による景観再現・創造—	米子	河野 清尊	梶間 光法, 亀井千佳良, 鷺見 知洋, 近藤 醇, 田中 雄介
6	カンファイン —簡単カンファチェックイン—	長野	伊藤 祥一	奥本 隼, 若林 哲宇, 宮崎 大智, 大和 優介, 矢口 裕也
7	地域まとめーしょん	鳥羽商船	江崎 修央	島影 瑞希, 小林真佐大, NOROVERDENE BATZAYA, 服部龍太郎, 野村 昭好
8	ShuWord —手話で旅するお手伝い—	沖縄	神里 志穂子	比嘉 早紀, 上江洲有希, 米須 樹, 大城 聖也, 比嘉健太郎
NUU	Travel Round the World Virtually	国立聯合大学	Chin-Chuan Han	Jhuo Xyun-Wu, Hao-Wen Yao
9	スマートアンブレラ	香川(高松)	重田 和弘	矢野 正人, 助安 涼, 米井 裕紀, 田口 拓明, 長谷川雄太
10	旅Navi —これ一つであなたの旅を快適サポート—	木更津	坂元 周作	秋葉 匡伸, 七条 聖哉, 富重 博之, 則友 一磨
11	ぐるりん —スマートフォンで観光情報を簡単発信—	熊本(熊本)	島川 学	白井 武史, 加藤 孝史, 坂本 譲二, 大和 康平
12	BOLT —BOdy Language Translator—	新居浜	先山 卓朗	稲田 利亀, 新宅 隆弘, 飛鷹 智浩
13	TRGP —TravelerRoadPlayingGame—	福井	斉藤 徹	夢田 一貴, 青山 昂平, 玉木 義孝, 山腰 貴大
14	MUSIVEL —TPO JUKEBOX—	長岡	竹部 啓輔	小林 樹矢, 高橋 悟, 武田 修平
15	旅管 —“イマドキ”の団体旅行“超”管理システム—	沼津	鈴木 康人	加藤 史也, 庄司 直樹, 湯山 孝雄, BALJINNYAM
16	PICT COMMY	舞鶴	中川 重康	辻 春樹, 蔭山海一郎, 芦田 陸, 大利 晃司, 中田 康貴
17	僕らのShareWalk —地元の良さを知る旅支援システム—	八戸	釜谷 博行	坂上 拓哉, 山下 和志, 管 貴志, 釜谷 諒悟, 花牟礼茂人
18	Floor→Flow Navigator	一関	小保方 幸次	吉田 琢朗, 舘崎 優人, 大関 啓太, 佐々木拓也, 鈴木 拓弥
19	旅人 —あなたと私で二人旅—	熊本(八代)	村田 美友紀	濱崎 瞳, 坂上 孝平, 梅川 孝佑, 岩本 優, 関原 至音
20	パシャナビ! —現在位置確認システム—	金沢	藤澤 武	堀岡 瞳, 花谷 恵里, 夢田 朋生, 長谷川一平, 浅井 一真

1

MAPHIS 一歩いて広がる歴史の世界一

松江

荒川ひかり(3年) 難波 隼也(3年)
津田 達也(3年) 索手 一平(3年)
内田 譲大(5年) 福岡 久雄(教員)

1. はじめに

旅行で初めて訪れる街では、その土地の歴史に興味を持つ人が多いのではないのでしょうか？私達が行ったアンケート調査によると、一般人 125 人中 110 人が「興味がある」と答えました。しかし、観光パンフレット等に基づいた固定的な観光ルートを進むだけでは、旅の楽しさが今ひとつ欠けてしまうと考えました。そこで、旅行者に自由な行動を促すことによって、探検心を掻き立てるような歴史観光を可能にするシステム「MAPHIS」を提案します。

2. システム概要

「MAPHIS」は図 1 に示すように、4 つの機能から構成されます。旅行者は訪れたい歴史スポットを選択し、MAPHIS が提示するヒントを頼りに街中を散策しながら、目的地を目指します(マフィサポ)。この時、目的地までのルートを詳細に指定しないことによって、自由な散策を促します。目的地の歴史スポットに辿り着くと、それに関する歴史解説が閲覧できます(マフィスト)。この解説情報は地域住民が登録することを想定しています(マフィレジ)。旅先で得た歴史情報と散策した軌跡は自動的に記録され、後で振り返ることができます(マフィップ)。

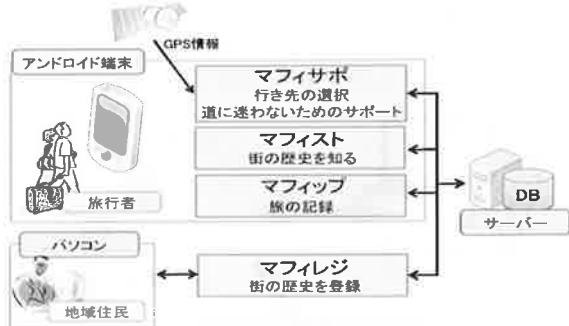


図 1. システム構成図

3. システムの機能

3. 1 自由な散策の促し (マフィサポ)

MAPHIS では目的地の歴史スポットまでの詳細なルートをあえて表示せず、ヒントを与えるのみです。例えば、歴史スポットの画像をジグソーパズルの形で表示します(図 2)。街歩きを開始する時、歴史スポットとの距離に応じた数のピースが埋まっています。歴史スポットに近づくときピースが増

え、逆に遠ざかると減っていき、辿り着くとパズルが完成します。こうして感覚的に距離を掴むことができる一方、辿るルートには制限を付けず、自由な散策を促すことで、目的地の歴史スポットへ誘導することができます。



図 2. パズルのピースによる距離の遠近

3. 2 寄り道の促し (マフィサポ)

目的地に至る散策途中の地図には、目的地と関連する他の歴史スポットが表示されます。そこに寄り道すると歴史解説を閲覧できます。目的地に至ることだけを目標とするのではなく、寄り道もしてみることで自然と街並みに目を向けることができます。

3. 3 街の歴史を解説 (マフィスト)

歴史スポットに辿り着いた時、その場所の歴史解説を閲覧できます。街中で様々な歴史スポットを訪れることでポイントが溜まり、一定の数を満たすと地域住民のみが知る街の歴史も閲覧できます。教科書のように淡々と語られたものではなく、地域住民の視点で書かれています。

3. 4 旅の記録 (マフィップ)

マフィストで得た歴史解説や歩いた軌跡は旅行後でも振り返ることができます。旅行者が訪れた街のみを地図上に表示することにより、一人一人独自の地図が形成されます(図 3)。



図 3. 街を歩くと拡大する地図

4. おわりに

MAPHIS を利用した旅行者が、様々な場所の歴史を発見することで、街を巡る楽しさを感じてもらえると幸いです。

1. はじめに

「たびどあ！」は、旅に出たいと思っても、いま一歩踏み出せない人たちを支援することが目的の Web アプリケーションです。

ユーザはスマートフォン及び PC からサイトにアクセスすることで「たびどあ！」を利用することができます。ユーザにはそこで「たびどあ！」が尋ねるいくつかの質問に答えて貰います。その質問の結果に基づいて「たびどあ！」はユーザにオススメの観光地を提案します。そのため、ユーザは自分にとってオススメの観光地を手軽に知ることができます。

2. オススメの観光地を知る

① ユーザはサイトにアクセスします。すると、図 1 の画面が表示されます。



図 1:サイトのトップ画面

② ボタンをクリックして画面を進むと、図 2 のように、質問とその質問に対する選択肢が画面に表示されるので、ユーザは表示されている質問に対して、直感的に選択肢を選びクリックして答えていきます。

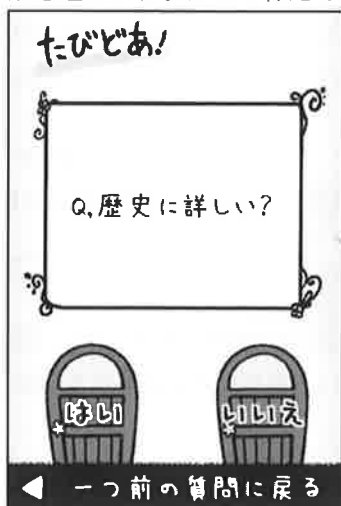


図 2:質問画面

③ 質問に答えていくと、図 3 のように結果画面が表示されます。結果画面にはオススメの観光地、観光地の情報・写真、大まかな地図が表示されます。

また、ユーザには、表示された観光地が気に入らなかった場合、「たびどあ！」の精度の向上のために、一覧から自分の気に入った観光地を選んで貰います。



図 3:結果画面

3. システムの実装について

オススメの観光地をユーザに提案するために、決定木というグラフを用いています。テスト段階でテストプレイヤーに質問に答えて貰っており、そのデータを元にオープンソースのデータマイニングツールである「Weka」の API を利用し、決定木を作成しています。

決定木作成に必要なデータ、観光地のデータはサーバに保管します。システムの構成を図 4 に示します。

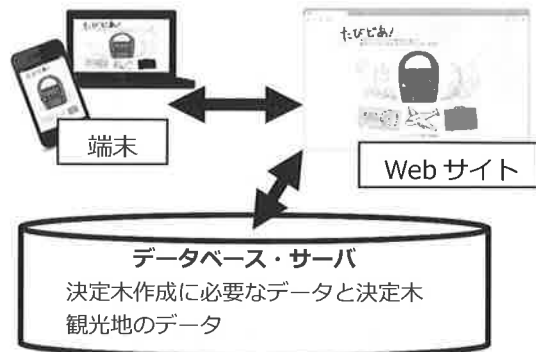


図 4:システム構成図

4. おわりに

みなさんは旅行に行くことがしばしばあるでしょう。ですが、いざ旅に出ようと思いついても、どこに行こうかと悩んでしまうことがあると思います。

私たちの「たびどあ！」が、そんな人たちの助けになれば幸いです。

3

とらりす

一ノートを書き終わるまでが旅なのだー

舞 鶴

新保 智喝(4年) 小林 賢太(4年)
 金澤 朋寛(4年) 高山 凱久(3年)
 梅井 良太(3年) 中川 重康(教員)

1. はじめに

皆さんは過去に数々の旅をされてきたでしょう。計画・調査し、非日常な土地と時間に期待を膨らませた思いや旅先での体験、情景。それらはその時、その瞬間にしか得られません。しかし、人の記憶は劣化し、大切な思い出も日常に上書きされて消えてしまいます。

そこで、旅の計画や思い出をノートに書いて旅をしてみませんか？本システム「とらりす」では、アナログなノートでは出来ない「快適な旅の計画・下調べ・思い出の保存」を提供します。

2. システムの目的と内容

本システムの目的は、スムーズな旅行の設計・実行の実現と、思い出を記録して記憶を復元出来るようにすることです。

現在、手帳等に旅の計画や思い出を記す「旅ノート」というものがあります。旅ノートには、スケジュール・写真・絵・文章等を書き、快適な旅の実現と、思い出の保存を可能としています。しかし、旅ノートの利用を始めるには、デジカメで撮影した写真の印刷・切り貼り等の煩わしさがあり、手軽に始めることが困難です。そこで、Android タブレット端末で動作するアプリとして「旅ノートを実現することで、さらに便利な「旅ノート」の手軽な利用を実現しています。

3. 機能紹介

3.1 とらりすノート

本機能では、ノートの作成を行います。ノートには次の4つのページ（スケジュール、下調べ、思い出、フォトアルバム）があります。

スケジュールページでは、目的地等に関する最低限の情報をブロック単位で入力し、旅のスケジュールを自動計算します。また旅の途中での予定変更も再計算による再構築が可能です。

下調べページ・思い出ページは、旅先の情報や思い出を文字やキャプチャ画像、写真として入力します。

3.2 とらりすビューア

この機能は、とらりすノートの閲覧機能です。基本的な見開きではなく、「帯形式」(図1)という独自の閲覧方法で、快適な閲覧を提供します。

この「帯形式」とは、スケジュール、下調べ、思い出、フォトのページを上から順に並べたものです。また、横方向は時間軸となっています。この閲覧方法により、日付ごとの情報を縦に一括で閲覧できます。

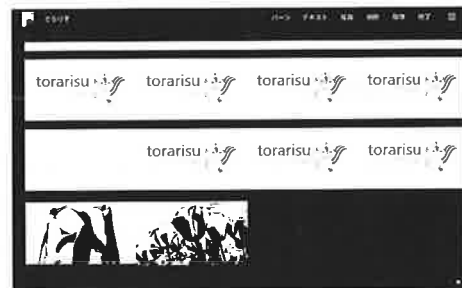


図1 とらりすビューアの開発中動作画面

3.3 とらりすパーツ

この機能は、絵が苦手なユーザや、手軽にノートを書きたいユーザ用のノート装飾パーツ提供機能です。パーツは、パーツ作成機能による簡易パーツの作成や、PCで作成した物の利用が可能です。

3.4 とらりすオンライン

この機能は、インターネット上のwebページです。ここでは、ノートやパーツのアップロード・ダウンロードや、ユーザ同士のコミュニケーションが可能です。

4. おわりに

本システムを利用により旅に関する情報の一元管理が可能となり、旅のスムーズな設計と思い出の記録による記憶の復元が可能となります。

将来の展望としては、とらりすノートをパンフレットとして作成してとらりすオンラインで公開し、観光事業の活発化を目的とした利用方法等が期待できます。

4

旅ズバツ！

奈良

麻田 優真(専1年) 上岡 真也(5年)
 梶原 雄士(2年) 島岡 大遥(2年)
 恒川 凌(1年) 岩田 大志(教員)

1. はじめに

旅をもっともっと楽しくしたい。だれもが思うことです。しかし、楽しい旅のためには、煩雑で面倒な準備作業がともないます。

旅ズバツ！は自由部門の Qcloud と連携し、クラウドに存在する様々な情報を活用することで、あなたの旅を賢くサポートします。目的地と条件を旅ズバツ！に入力すれば、自動的に旅のプランの候補を提示します。候補の中からプランを選ぶと、旅ズバツ！が電車やホテルの予約まで自動的に行います。今までバラバラに行っていた旅のプランの考案や電車の予約などが、すべて旅ズバツ！にまとまるのです。

旅ズバツ！は、旅をより楽しみたい、すべての人をターゲットにしたシステムです。

2. システムと動作

2.1 動作環境

本システムは、タッチパネルディスプレイが搭載されており、自由部門で提案する Qcloud と呼ばれるフレームワークが動作する端末で使用することができます。Qcloud が動作する環境であれば、パソコンであるかスマートフォンであるか、OS は何であるかは問いません。したがって、様々な環境で動作させることができ、活用の幅が広がっています。

2.2 旅ズバツ！でできること

①目的をもとにした旅のプランの提案

パソコンやスマートフォンで、旅ズバツ！に目的地や出発地、宿泊日数などの条件を入力します。入力された条件をもとに、旅ズバツ！が自動的にルートや料金、宿泊施設などをまとめた旅のプランを提案します。ユーザは、提案されたプランから気に入ったものを選ぶことができます。

②電車や宿泊施設の予約

鉄道会社や宿泊施設が予約を行う API を用意していれば、Qcloud アプリを通して、ユーザが選択した旅のプランをもとに電車や宿泊施設を予約できます。従来、別々の Web サービスで行っていた煩雑な予約作業が、旅ズバツ！により、ズバツとワンアクションでできるようになります。

③旅のプランの告知

複数人で旅行する場合、各メンバーに旅のプランについて告知する必要があります。従来、この作業は面倒でした。

旅ズバツ！は、クラウドを用いる強みを生かして、

さまざまな Web サービス (Facebook や Twitter, メールなど) と連携し旅のプランの告知を行うことができます。

旅のプラン作成から、予約、プランの告知まで、一連の作業をズバツとひとつのシステムで行うことができるのが、旅ズバツ！の特長です。

2.3 インターフェース

旅ズバツ！は主に2つのインターフェースが切り替わり動作します。

①プランモード

プランモードでは、目的地や条件を指定します。クラウドにアクセスすることで、ネットワーク上に存在するデータを用いて最適なプランを提案します。予約や告知なども、この画面で行います。

②確認モード



確認モードでは、移動中や現地でスケジュールやルートの確認ができます。また、現在の交通情報や観光地に関する情報を調べることもできます。プランの変更は、プランモードに戻って行います。

2.4 公共端末の利用

旅ズバツ！で用いる Qcloud は、様々な端末に組み込まれることを想定しています。そのため、旅ズバツ！を公共端末に組み込むこともできます。

その土地、その場所に合わせたアプリ (以下ローカルアプリと表記) を、公共端末の提供者が導入することにより、旅行者に従来よりも一歩進んだコンテンツを提供することができます。

たとえば、観光地に関連したゲームを用意することなどが挙げられます。また、クラウドを用いているので、雑記帳を模したローカルアプリに書き込んだメッセージを、家に帰って見返して、旅の思い出にひたるという活用例も考えられます。

5

SnapSwapTrimap

—携帯端末による景観再現・創造—

米 子

梶間 光法(5年) 亀井千佳良(3年)
鷲見 知洋(3年) 近藤 醇(3年)
田中 雄介(3年) 河野 清尊(教員)

1. はじめに

旅先の観光地で、天候が悪くて、見れたであろう景色を楽しめない、あるいは工事中で建造物を見れない、などの残念な経験をしたことはないでしょうか。このような旅先での悩みを解消することを目的に、多機能携帯端末を用いた景観再現・景観創造システム『SnapSwapTrimap』(以下、本システムと呼ぶ)を開発することにしました。

2. システムの概要

本システムは、観光地のある地点から見える風景・景色の画像をあらかじめ撮影しておき、その画像を方位・季節・時間帯を自動検出して iPad2/iPhone に表示することにより、上述の旅先での悩みを解決しようというものです(景観再現)。また、景観再現画像およびユーザが撮影した画像に、人・物の合成や装飾・音の効果の付加などを行うことにより、新たなコンテンツの創造を行えるようにしようというものです(景観創造)。本システムにより、旅人に観光地の良さ・魅力を訴求することができ、地域の観光振興に役立てることができそうです。

2.1 システム構成

本システムは、図1に示すように、iPad2/iPhone およびインターネットに接続されたサーバから構成されます。

- ①まず、iPad2/iPhone から位置・季節・時間帯を指定して、サーバに景観再現画像を要求します。
- ②サーバから取得したその画像を、iPad2/iPhone の向いている方位に合わせて表示します。(景観再現)
- ③また、iPad2/iPhone で撮影した画像やサーバから取得した景観再現画像に、
- ④画像合成や効果の付加などを行います。(景観創造)

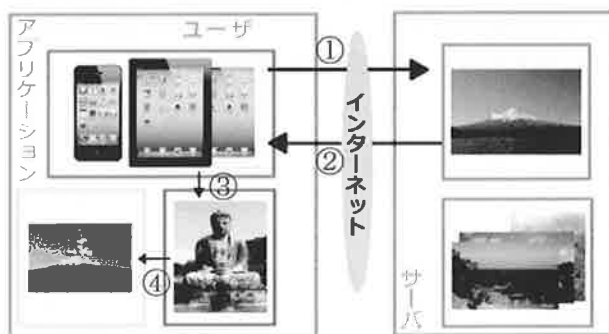


図1 システム構成

2.2 システムの機能

本システムは、観光スポットに iPad2 を置いて訪れた観光客に使ってもらう『観光地設置型』と、個人所有の iPad2/iPhone で使ってもらう『旅人携帯型』の2つの用途があります。それぞれに以下のような機能を持たせます。

(1) 観光地設置型

【景観再現機能】 その観光スポットにおける

- ①現在の風景と観光情報の表示
- ②方位(8方向)・季節・時間帯を自動検出し、対応した景観再現画像の表示(図2)
- ③季節・時間帯・時代を指定して、方位(8方向)を自動検出し、対応した景観再現画像の表示

【景観創造機能】 景観再現画像に対しての

- ①画像の合成
- ②効果の付加

(2) 旅人携帯型

観光地設置型に加えて以下のような機能を実現します。

【景観再現機能】

- ①マップの表示と景観再現したい観光スポットの選択
- ②任意の場所において、ユーザが iPad2/iPhone で撮影した8方向の画像の景観再現への登録と削除

【景観創造機能】

- ユーザが iPad2/iPhone で撮影した画像に対しての
- ①画像の合成
 - ②効果の付加
 - ③画像の保存

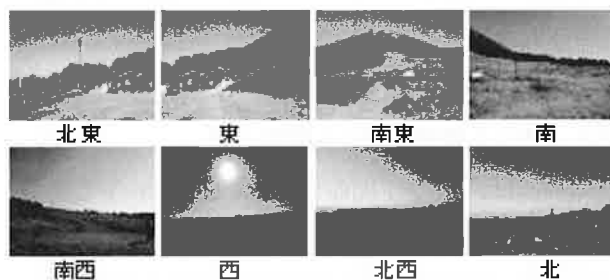


図2 景観再現の画面(8方向)

3. 実現にあたっての課題と解決策

【課題1】景観再現画像の確保

現在、観光スポット5ヶ所の夏と秋の画像および1ヶ所の秋の朝昼夕の画像を撮影しました。

【課題2】画像への位置・方位・季節・時間帯情報の付加
これらの情報をファイル名に含めることにより識別できるようにしました。



図3 360°パノラマ撮影したシームレス画像

4. まとめ

今後は、景観再現を図3に示すような360°パノラマ撮影したシームレス画像、あるいは360°レンズを用いたシームレス画像で行えるようにします。また、iPad2/iPhone で撮影した画像のアップロード機能を追加することにより、景観再現画像や景観創造画像をユーザ間で共有できるようにし、双方向型システムへの発展を目指します。

6

カンファイン

—簡単カンファチェックイン—

長 野

 奥本 隼(2年) 若林 哲宇(2年)
 宮崎 大智(2年) 大和 優介(2年)
 矢口 裕也(5年) 伊藤 祥一(教員)

1. はじめに

全国各地で、様々なイベントが行われています。遠隔地からのイベント参加者も多く、いまや、イベントへの参加=「旅」と言えるでしょう。しかし、告知方法や、参加確認、受付での出席確認をすべて手作業で行うのはイベントの規模によってはとても大変なことです。その作業を簡単にするシステム、それが「カンファイン」です。

2. カンファイン

カンファインの主な機能は以下の通りです。

2.1 システム概要

本システムは、イベントの作成、告知、参加、受付などをすべてサポートするシステムです。これらのサポートにより、イベントの主催者の負担を一気に軽減することができます。

2.2 イベント開催サポート機能

本機能は、イベント開催当日までをサポートします。

2.2.1 イベント作成から参加登録まで

始めに、主催者が本システム上にイベントページを作成します。イベントの内容は後からでも編集できます。イベントページ作成後は、主催者から Twitter 等を利用して告知を行います。告知を見た参加希望者は、イベントページから、簡単なユーザー登録を済ませた後、参加登録を数ステップで終わらせることができます。参加登録終了時、当日受付用の QR コードが発行されます。

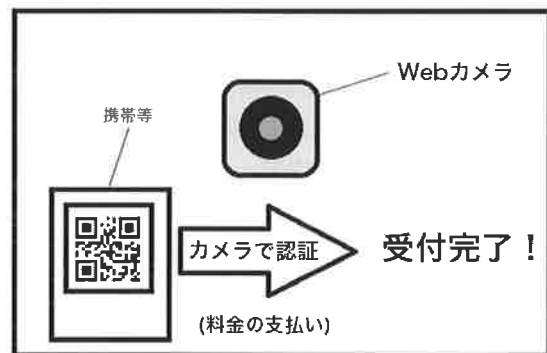
2.2.2 イベント参加登録後

参加登録後も、「メール一斉送信機能」を使用することにより、主催者からの一斉連絡が可能です。参加者に対しては、「リマインダーメール機能」により、イベントの開催日を確実に知らせることができます。参加者がイベントについて不安があっても、「Q&A 機能」

により、主催者に質問をすることができます。

2.2.3 イベント当日

イベント開催当日は、携帯電話などに保存した QR コードを用いて、受付システムによって簡単にチェックインができます。受付システムは、PC と Web カメラ、インターネット環境さえあれば利用できるため、受付の準備に手間取ることもありません。



当日の受付

3 既存システムとの比較

本システムでは、既存システムにない QR コードによる参加者受付機能により、参加者にとっても主催者にとっても受付が楽になります。イベント終了後も、主催者のコメントなどをまとめた「思い出機能」により、振り返りができます。

	本システム	既存システム
イベント告知	◎(Twitter 利用)	○
当日の受付	◎(QR コード)	×
コミュニケーション	◎(Q&A)	△もしくは×
イベント終了後の支援	◎(思い出機能)	×

4 まとめ

本システムにより、主催者の煩雑な作業が軽減し、参加者は、イベント当日までの不安をなくし、イベント終了後は振り返りができます。

7

地域まとめーしょん

鳥羽商船

島影 瑞希(3年) 小林真佐大(3年)
 NOROVERDENE BATZAYA(3年) 服部龍太郎(2年)
 野村 昭好(2年) 江崎 修央(教員)

1. はじめに

日本各地にはさまざまな観光団体があり、地域を盛り上げる為にイベントの企画や実施に奮闘していますが、思うように行かない事もあります。そこで私達はスムーズに企画立案と情報共有ができるシステム「地域まとめーしょん」を開発しました。本システムの利用により旅行者に向けて有益な情報配信ができます。

2. システムの概要

本システムはイベントの企画ページを作成し、そこで議論を進めます。ある程度意見が出たら「まとめ機能」により立案者の任意のタイミングで要約することができます。完成した企画の情報は各団体に配信され閲覧可能になります。

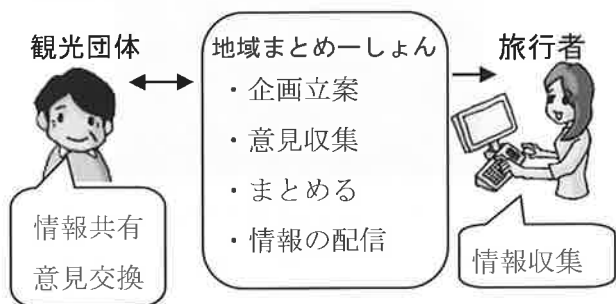


図1 システムの概要

3. 企画立案と意見収集

新しい企画を投稿する際、主催者は内容を図2の項目ごとに入力します。企画ごとに検討内容は異なるので、画面下部に用意された「新しい項目の追加」ボタンから追加します。本システムではラベルとテキストボックスを1つのユニットとして扱い、まとめる単位にします。

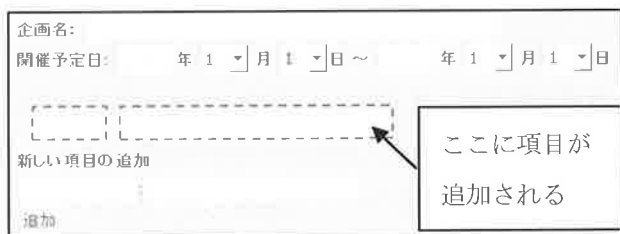


図2 新規記事の作成画面

企画が立案されると関連団体を交えて議論が開始されます。意見の書き込みは、「画像貼付け」「手書き」「付箋」機能を利用します。例えば、「付箋」を用いる場合は意見を記入したい項目をクリックすれば付箋が表示され、利用者は自分の意見を記入することができます。

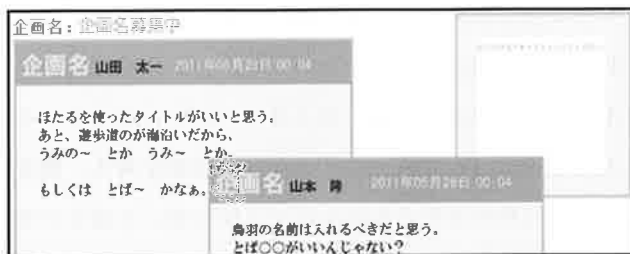


図3 意見書き込みの例

4. まとめと情報配信

まとめ機能では、各ユニットごとに集まった意見を要約します。ユニットの内容から頻出単語をカウントしランキングを付け、表示します。主催者はこれを利用し企画を編集、要約します。



図4 まとめ画面

ホームページとしてまとめて配信する他、RSS 等を利用して各団体やニュースサイトとも連携できます。



(a)ホームページの例 (b)ポスターの例

図5 情報提供の例

5. おわりに

本システムでは関係団体が直接意見交換できるので、意思疎通と情報共有が迅速に行えます。旅行者にとっても情報収集が容易に行えることで、魅力ある観光地になるのではないのでしょうか。

1. はじめに

現在、日本政府による聾者人口は重軽度合わせ 30 万人を超える数にある。多くの聾者にとって、社会生活におけるサポートは、十分とは言い難く、不慣れた場所で道を尋ねることや、病院などで十分なコミュニケーションが取れないなど普段の生活などでも不便を強いられる場面が少なくない。そのため、旅に出かけることへの不安は、健常者よりも大きいと考える。そこで私たちは IT 技術での社会福祉の貢献を考え、聾者にとって旅や普段の生活の中で不便を感じる場面に置いて活躍する手話変換ツールの開発を目指している。

2. システム概要

本システムでは、図 1 に本システムの構成を示す。本システムは、聾者の手話をデータとして計測するグローブ上のデバイス部分とその手話を文字として変換し健常者が確認できる Android 端末から構成されており、聾者と健常者との会話をサポートするツールとなっている。加速度センサ・曲げセンサのついたグローブが手話の動作を感知し、Bluetooth で Android 端末に送信される。そこであらかじめ用意しておいた手話データベースを参照し、最も適切だと思われる語を画面に出力するという仕様になっている。

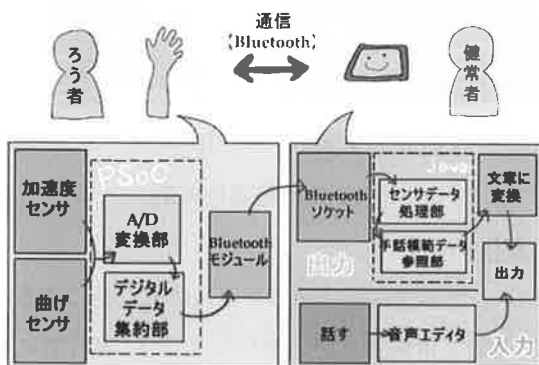


図1 システム構成

3. 機能説明

まず、手話者はデータグローブを両手につけ、道などを尋ねる会話を行う。その手話による指などの関節角度情報が Bluetooth を介して Android 端末に送信される。Android 端末ではそれを文字として出力する。これによって、手話を理解できない人にも質問や気持ちを簡単に伝えることが可能になると考える。

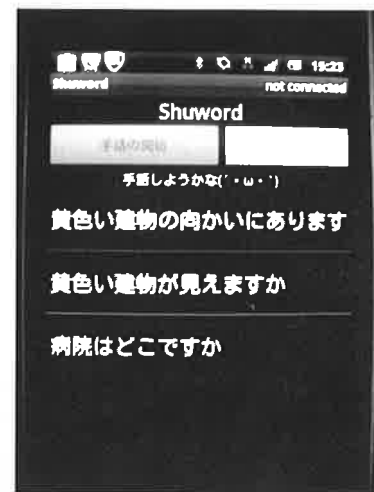


図2 Android 端末画面

手話者からのメッセージに対して健常者が返信する際には、音声による入力を行う。まず、Android 端末の画面にある返信ボタンを押す。すると音声入力画面がポップアップされ、そのまま端末にむけて返信を伝えることで文字として出力される。もし音声入力に誤認があった場合は、表示された出力内容をタッチすると、削除の有無を聞くウィンドウが表示されるようになっている。

4. システムの特徴

このシステムはデータグローブと音声認識によって、聾者と健常者の双方向でコミュニケーションをとることが可能である。また、画面からの動作認識ではなく、データグローブのデバイスを利用することによって、相手の動作や表情を確認しながら、会話を行える点がこのシステムの利点である。

1. Motivation

Google street-view service provides panoramic images by which users could view the surrounding scenes along the streets. In this study, a virtual traveling system is developed based on the ‘Google Earth’ platform and the commands are identified using the Kinet input device. People explore the world from the Internet virtually in natural interaction. In addition, they can take a picture and record the video during the journey.

2. System Architecture

The traveling system consists of two modules: the action identification(AI) and the scenery exploration(SE) modules as shown in Fig. 1.

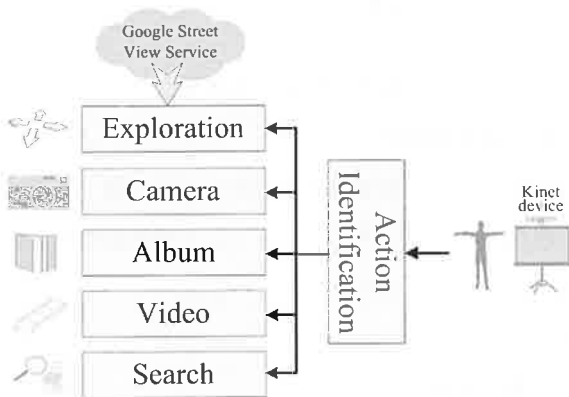


Fig. 1: System Architecture

2.1 Action Identification (AI) module

An RGB color image and a depth-based sensor image are extracted for action identification. The identification rules are designed using the 15 joint coordinate features of human body. Eleven actions, six hand gestures and five body postures, are classified by using the relationship among these 15 joint features.

2.2 Scenery Exploration (SE) module

The APIs of Google street view service can move the street views free in SE. Human objects are segmented from the color images and are embedded in the street images.

Furthermore, a video is obtained from the continuously grabbed human objects and the serial street images for the complete remembrance.

2.3 System Flowchart

Users stand at the frontal of a projected screen. Kinet device identifies the human postures, or action instead of the mouse-based or keyboard-based inputs. Using the Kinet-based commands, users see the scene free by the Google street-view service as shown in Fig. 2. In addition, they can take a picture or record a video for remembrance.

3. System functions

Five functions are developed in this system:

- 1) **Exploration:** Users exhibit four postures to move forward, move backward, turn left, or turn right. Using them, users move free in the Google street view service.
- 2) **Camera:** When two hands applaud, the system will count down in 5 seconds. Users can exhibit any pose to take a picture.
- 3) **Album:** Users browse and manage the album using the gesture commands.
- 4) **Video:** The system like a private cameraman records the complete journey using videos.
- 5) **Search:** The system jumps to the chosen high spot to begin the exploration.



Fig. 2: System Implementation

9

スマートアンブレラ

香川(高松)

矢野 正人(専2年) 助安 涼(専1年)
 米井 裕紀(専1年) 田口 拓明(5年)
 長谷川雄太(5年) 重田 和弘(教員)

1. はじめに

旅に悪天候への備えとして傘を持っていくが、その傘もただ雨避けというだけではもったいない。ところで、旅するなかでは道案内や天気予報などの情報を知りたいものである。

そこで旅にとって便利な機能を組み込んだ傘型ガジェット「スマートアンブレラ」を開発した。

2. システム概要

2.1 特徴

スマートアンブレラは傘に様々なアクセサリを組み込み、スマートフォンで制御することで雨の日の旅行をサポートするガジェットである。ハンドルに内蔵したボタンを利用することで、傘を差しながら操作が行える。また、露先(傘の骨の先端部分)に取り付けられた8個のカラーLEDランプの発光パターンとイヤホンからの音声により情報を得ることができる。画面を見る必要がないため歩行中でも安全に使用できる。

2.2 システム概要

スマートアンブレラはスマートフォンの磁気センサや加速度センサ、GPS機能、インターネットなどを用いて道案内や天気予報などの機能を露先のLEDの点灯などによって提供する。

スマートアンブレラとスマートフォンはBluetooth、スマートフォンとインターネットはWi-Fiまたは3G回線を用いて接続される。

3. 機能説明

スマートアンブレラはコントローラや音声入力を用いて機能を切り替え、操作できる。

以下で主な機能を紹介する。

3.1 道案内機能

スマートフォンで設定した目的地までの道案内を行う機能である。GPSで現在位置を取得し、目的地の方向に対応するスマートアンブレラのカラーLEDランプを点灯させる。

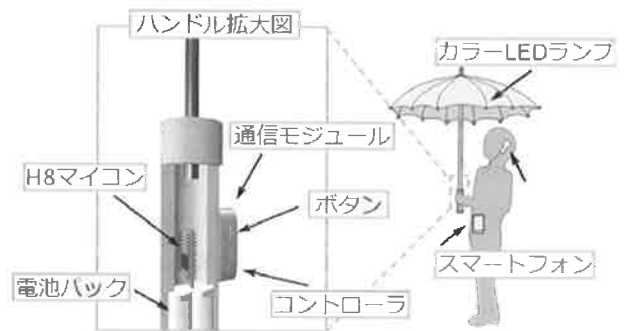


図1 システム構成図

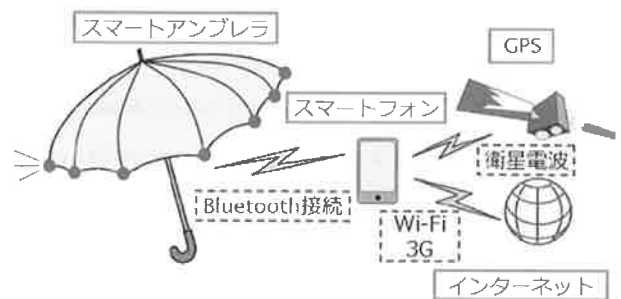


図2 システム概要図

3.2 天気予報機能

これからの天気を使用者に知らせる機能である。GPSで現在位置、インターネットで天気情報を取得し、今後の天気に合わせて色でカラーLEDランプを点灯させる。

3.3 通話機能

ヘッドセットなどを利用し、電話帳に登録された相手と通話することができる。

4. 既存システムとの比較

携帯端末の地図アプリや天気予報アプリは、雨の日には傘や荷物で両手がふさがっている状態では使用しづらい。スマートアンブレラは傘そのものが機能を提供するので雨の日でも手軽に使用できる。

5. まとめ

スマートアンブレラは道案内や天気予報といった機能を使用することができる傘型ガジェットである。歩行中でも安全に使えるため、雨の日の観光を安全・快適にサポートする。

1 はじめに

近年のインターネット利用者の増加に伴い、生活を豊かにするウェブサービスが多く誕生している。最近急速に普及しつつあるスマートフォンにおいては、そうしたウェブサービスを利用したアプリケーションも数多く登場している。旅行についても例外ではなく、旅行者をサポートするアプリケーションが多数見受けられるが、様々なウェブサービスを一元化するものは存在しない。

そこで我々は各種ウェブサービスの統合により、スケジュール管理やナビゲーション、アルバム生成といった種々の機能を実現する、AndroidOS 対象のアプリケーション開発を行うこととした。本アプリケーションが、旅行前から旅行後まで楽しめる、名アシスタントとして活躍することを期待する。

2 システム概要

本システムでは図 1 に示すような複数のウェブサービスの API を利用することで、複数のアプリを操作できるよう工夫した。使用した API を以下に示す。

- ・ Yahoo ローカルサーチ API…キーワードからの目的地検索システムに使用。
- ・ GoogleAPI…Google Map の表示、ナビゲーション機能に使用。
- ・ Twitter4J…java 用の twitter API。twitter との連携に使用。
- ・ wikipediaAPI…名所案内システムに使用。



図 1: システムの概要

3 機能説明

3.1 目的地の設定

はじめにユーザーはキーワードまたは地図から目的地を設定する。目的地訪問の順番を設定すると、Google の経路検索機能によってナビゲーションが可能となる。

3.2 ナビゲーション機能

Google のナビゲーション機能へアクセスし、交通機関や経路の表示を行う。

3.3 Twitter 投函

アプリ内から Twitter 投函が行えるので、ブラウザを起動して Twitter にアクセスする手間が省ける。投函した際、現在位置が Android のデータベースに保存される。この位置データは下記に述べるアルバム生成時に利用される。

3.4 名所案内

各目的地に対して、Wikipedia から情報を取得、成形し表示する。

3.5 アルバムの自動生成

Twitter への投稿 (投函時の現在位置データを含む) や各観光地で撮影された写真の情報などは Android のデータベースに自動保存される。時系列にそって自動でアルバムが作成され、簡単に思い出を振り返ることが可能である。

4 おわりに

本アプリケーションの特徴は、現地での行動、旅の記録整理まで旅行に必要な様々な機能へのアクセスを集約した点である。これを利用することで、スマートフォン初心者でも簡単に旅を楽しむことができると期待される。

1. はじめに

インターネットによる観光情報の発信手段として、旅行者自身が訪れた観光地の写真や情報を寄せたり、利用した宿泊施設を評価したりするなど、利用者参加型が増えています。最近では、FacebookなどのSNS利用者同士で情報が広がる「クチコミ効果」を利用するものもあります。本システム「ぐるりん」は、スマートフォンの機能を活用して、誰もが簡単に観光情報(特に写真情報)を提供し合い、共有できるサービスを提供します。

2. システム概要

観光地の風景写真などをスマートフォンで撮影し、インターネット上のサーバーにアップロードして公開するだけならよくあるサービスですが、本システムはスマートフォンの機能活用に特徴があります。

2.1 写真撮影機能

最近のスマートフォンの中には、GPSで現在位置を取得するだけでなく、電子コンパスや加速度センサーによって向いている方角まで取得できる機能を備えているものもあります。本システムでは、この機能を利用して、スマートフォンを構えたままその場で「ぐるりん」と一周するだけで全8方位の写真を撮影できる機能を実装します(図1)。

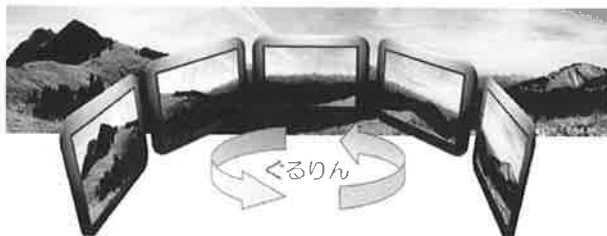


図1 写真撮影の動作

2.2 撮影した写真の公開

撮影した写真はスマートフォン内のメモリーに保存されますが、そのデータをサーバーにアップロードして、他の利用者からも閲覧できるように公開することができます。また、アップロードする際には、撮影場所や撮影日時、タイトルやコメントなどの情報を付加することができます。

2.3 閲覧機能

スマートフォンに保存されている写真データや、サーバーで公開されている写真データを、スマートフォンやパソコンなどから閲覧することができます。スマートフォンで閲覧する場合、写真撮影の場合と同じようにその場で「ぐるりん」と一周すると向いている方角に合わせた写真を閲覧することができます。

3. Facebookとの連携

本システムは、図2に示す処理の流れのようにFacebookとの連携を行います。情報を発信しようとする利用者はFacebookでユーザー認証を行った上で写真データを投稿します。自分が投稿したデータを修正したり削除する場合にも、Facebookでユーザー認証を行います。これにより、本システムのサーバーはユーザー管理をする必要がなくなります。写真データを閲覧する一般の利用者がFacebookユーザーであれば、「いいね!」ボタンをクリックするだけで、友達と情報を共有することができ、クチコミで観光情報が広がっていく仕組みを利用することができます。

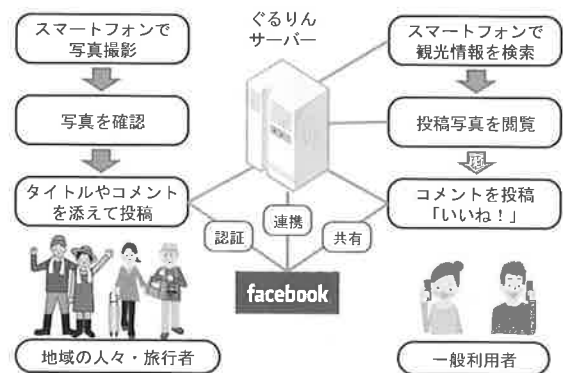


図2 処理の流れとFacebook連携

4. おわりに

本システム「ぐるりん」は、スマートフォンがあれば誰でも利用でき、誰もが観光情報を発信できます。スマートフォンを情報収集のための道具としてだけでなく、観光情報を発信する道具として活用できます。地元で暮らす人々にとっては普段の生活の中の何気ない風景であっても、他の地域の人から見れば魅力的な場所かもしれません。本システムが、そのような埋もれた観光資源の発掘に貢献できれば幸いです。

1. はじめに

旅先で異文化交流をする際、ボディランゲージは有用な手段です。しかし、これらには国や宗教、言語により多くの差異が存在し、それが原因でトラブルが発生することがあります。本システムは、対象者間でボディランゲージを翻訳することで、トラブルを未然に防ぎ、文化交流を促進することを目的としています。

2. システム概要

本システムは「翻訳機能」「共有機能」「学習機能」の3つのサブシステムから構成されます。

2.1 翻訳機能

翻訳時のシステム構成図を図1に示します。

ユーザは、自分の宗教、言語、地域、または登録IDを入力し、Web上から動作の翻訳辞書をダウンロードします。システムが起動すると、画面に自分の母国語で相手の動作の意味とそのログ、それと自分の動作の翻訳結果が表示されます。動作の入力には、Kinectセンサーを、翻訳にはBing Translation APIを用います。

この機能で、もし自分が相手の価値観でいう失礼な動作をしたとしても、相手が自分の側での意味を知ることができ、大きなトラブルを防ぐことができます。

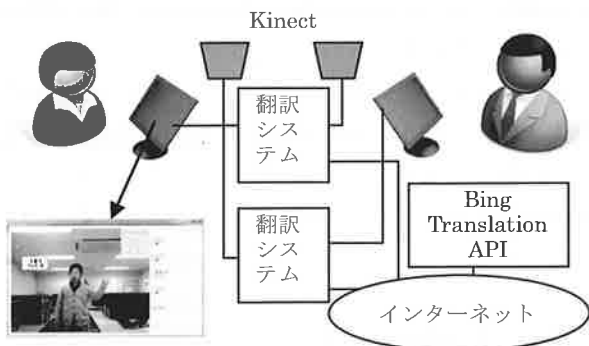


図1 翻訳機能システム構成図

2.2 共有機能

共有機能は、辞書をWeb上のデータベースで宗教などの情報とともに管理します。また、必要に応じ辞書を再構成し、翻訳プログラムへ返します。この機能で、旅先での利用が可能になるほか、辞書の不足を補い、未登録者にもシステムを利用できるようにします。

2.3 学習機能

学習機能は3つの作業でユーザ独自の辞書を作成します(図2)。まず動作入力段階で、実際に体を動かして動作を入力します。次に動作分類段階で自動分類された動作を、必要に応じて手動で訂正します。最後に意味入力段階で、それぞれの動作について意味を入力

します。学習機能を3つに分けることにより、たくさんのデータをより簡単に入力しやすくしました。

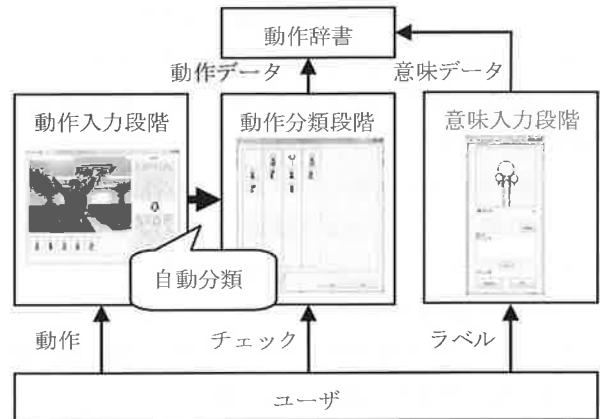


図2 学習システム構成図

3. 利用者へのアンケート

プロトタイプを作成後、一般の方に利用していただき、アンケートを取りました(図3)。

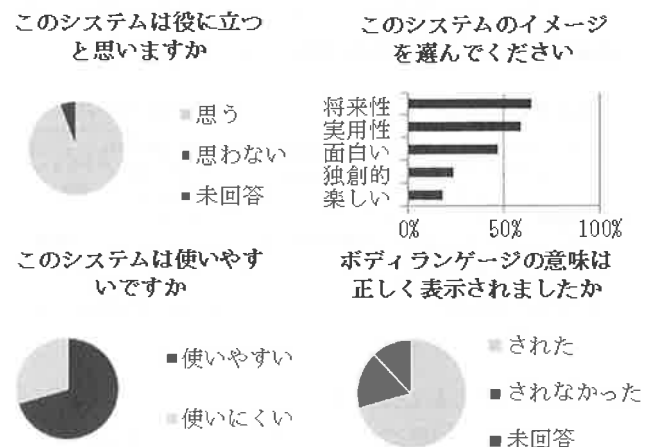


図3 アンケート結果(一部)

良い面として、有用性を広く認めていただけたことが挙げられます。また、半数が将来性や実用性を感じています。改良点として、3割近くの人がシステムに使いにくさを感じている点があり、特に学習機能について課題が残っています。さらに、2割近くの人が翻訳結果を「正しくない」と答えており、正確性を増すためにより改良を加える必要があります。

4. おわりに

本システムによって、コミュニケーションをとる上での不要な心配を取り除き、旅先での異文化交流が促進できたら幸いです。

1. はじめに

今まで旅を支援するシステムがいくつも存在してきましたが、その多くは端末や機械と向き合っただけで冷たく無機質なものでした。旅行者が堪能すべきものは押し付けられた技術ではなく、**旅先そのものであるべきです**。私たちが提供する「TRGP」の使命は、旅行者が本来楽しむべきである旅先の魅力や出会いを旅人本位、つまり技術を使いつつもそれにあまり依存することがない形で提供することです。TRGPを構成する3つのアイテム、Androidアプリケーション「トラベラーズナビゲータ」、布に縫い付けて扱うLilypadArduinoを基に液晶ディスプレイや加速度センサ、振動モータやスピーカを搭載して作られた、身に付ける独自デバイス「トラベラーズバッチ」、地域の魅力を伝えるためのwebサービス「Missionセンター」。これらを用いて実現される様々な仕組みによって旅先の魅力を十分に伝え、味わい、一期一会の出会いを楽しめることを目指し、21世紀の旅にあるべき姿を提案します。

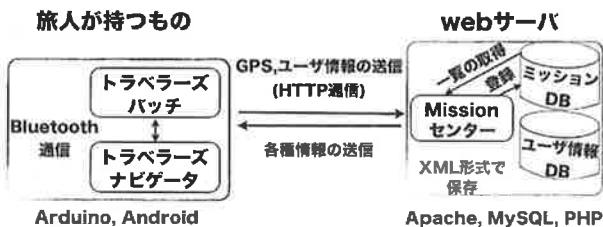


図1: システム構成図

2. 「Mission」という新発想

「登録 → 遊ぶ」で、伝える、伝わる。

TRGPでは、「その地域の魅力を楽しく知りたい。」という旅人の思いと、「地元の魅力伝えたい。」という地域住民の思いを満たすMissionという仕組みを提供します。Missionは、旅人に与えられる旅先独自の指令やクイズのようなものです。「ある場所へ向かえ!」といったものや「〇〇が売っている店の名前は?」「△△が出来た時代は?」などといった指令を与える事で、旅人はこれに答えるために実際に観光地へ向かったり、回答の過程で調査したりする事で、その地の歴史や魅力をまるでゲームのように楽しく知る事が出来ます。このMissionは、魅力を伝えたい人が専用webサービス「Missionセンター」上でテンプレートに沿って位置情報と共にデータベース

に登録します。Missionの内容によって旅人に情報を伝えたり、特定の場所に誘導したりする事が可能です。登録されたMissionは旅人の持つAndroidアプリケーションに対して、GPS情報を参考に旅人が今いる地点の周辺に登録されているものが送られます。

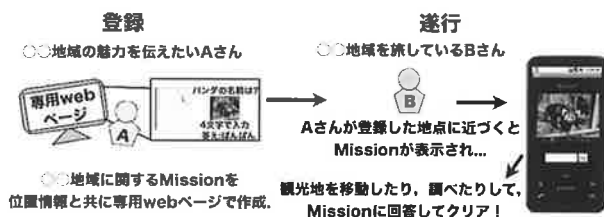


図2: ミッションの流れ

3. 独自デバイス「トラベラーズバッチ」

触れない。見て、分かる。

旅先で、ずっと端末とにらめっこ。そんな寂しい思いはもうありません。TRGPではトラベラーズバッチを身につけることで、Androidを操作したり画面と向き合う時間を少なくし、**旅そのものを楽しめる仕組みを提供します**。バッチは搭載された各種センサ、ディスプレイにより、ジェスチャー認識による自分の感情や情報の入出力、近くに旅人やMissionに関する情報があることを通知、といったことを実現し、自分の今の感情や状況を目の前の人に直感的に伝えて交流の円滑化をはかったり、MissionをAndroid端末に依存しない自由なスタイルで楽しんだりする事が出来ます。画面を見るのは、また、操作をするのは必要なときだけ。それがこのバッチのコンセプトです。

4. 旅人同士の新しいコミュニケーションのカタチ

トラベラーズバッチとトラベラーズナビゲータ。この2つを用いて、普段はなかなか味わえない旅人同士の交流のきっかけと場を提供します。近くにいる旅人を振動や音で伝えたり、その旅人の情報を得られたりする機能により、**他の旅人のことをその場で知ることができ**、初対面の人との円滑な交流が可能になると考えています。また、登録されたプロフィールから自分との共通点がわかったり、旅人同士でMissionを共有できる仕組みによって、交流のきっかけが更に広がります。さらに、旅人が旅先に「宝」と呼ばれる、「思い出スポット」や「お勧めスポット」を位置情報と共に登録でき、それを検索する機能により過去に訪れた旅人のことを感じることが出来ます。

1. はじめに

旅に出かけても感傷に浸りきれない、また長い月日を経ることで思い出が薄れてしまう…なんて経験はないでしょうか？そこで、場所や気分に合わせて音楽を聞きながら旅をすることで、旅をもっと楽しく、印象深いものにしてはいかがでしょうか？

我々は人の選曲が自身の感情と周りの情景から影響をうけると考え、若い人から年老いた人まで、ユーザを問わず誰でも選曲の手間を考えず、感情と情景に合わせた曲を聞くことができるようなシステムを開発しました。

2. システムの概要

システムは、OSにAndroidを搭載した小型携帯端末上から利用することが出来ます。ユーザはAndroid端末上から専用ソフトを起動し、現在の自分の感情をソフトに入力するだけで楽曲を視聴することができます。操作上の手間を少なくするために、ユーザがソフトに対して最低限しなければならない操作はこれだけです。

ソフトは感情の入力を受け取ると、自動的にGPSによる位置コードをサーバに送出します。位置コードを受け取ったサーバはこの位置コードと、入力された感情情報に基づいて選曲を行います。選曲結果は、インターネットを通じてサーバからAndroid端末に送出され、ユーザは楽曲の再生を実現します。

3. 機能説明

3.1 感情グラフィタッチ

ユーザはスマートフォン上から図1のような感情グラフにタッチすることで、自分の感情をソフトウェアに入力することができます。自分の感情を正確にソフトに伝えることによって、その時の気分にあった選曲を行い、旅を盛り上げる手助けをします。

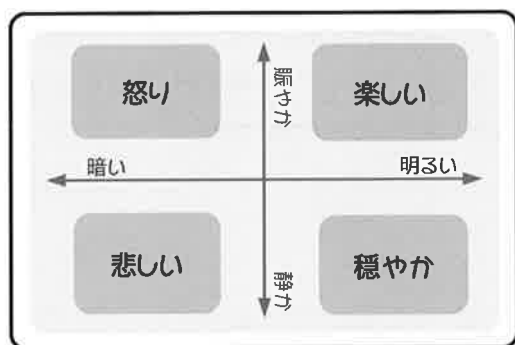


図1: 感情タッチ画面(イメージ)

3.2 位置コード取得

システムは起動時と、曲の切り替わりの時に自動的にAndroidのGPSから位置コードを受け取ります。受け取った位置コードから、時間、天気、その場における自然度、ユーザの移動速度、四季など合計5つの要素を解析します。アプリを起動するだけで自動的にこれらの処理が行われるため、ユーザに手間がかからないだけでなく、ユーザが普段気にすることのないようなその場での細かな情景まで読み取ることができます。

3.3 楽曲の管理

3.3.1 曲の登録

楽曲にはそれぞれの曲に適した「情景情報」、「感情情報」そして「曲調情報」の3種類の情報が付随して保存されています。新たに曲を登録する際には解析によって得た曲調情報の各要素を、データベース内のすべての楽曲の曲調情報の各要素と比較し、その結果を基に新規の曲の情景情報と、感情情報の要素の値を決定していきます。

楽曲情報		
感情情報	情景情報	曲調情報
<ul style="list-style-type: none"> ・楽しい ・種やか ・怒り ・悲しい 	<ul style="list-style-type: none"> ・時間 ・天気 ・自然度 ・移動速度 ・四季 	<ul style="list-style-type: none"> ・BPM ・迫力因子 ・金属製因子 ・美的因子 ・和音進行 ・アタックタイミング

図2: 楽曲情報

3.3.2 曲調情報

曲調データの要素には、BPM、迫力因子、金属製因子、美的因子、和音進行、アタックタイミングなど、6つの要素を解析します。人間には解析できないような多くの要素を自動で解析するので、様々な状況に対応させることができます。

4. おわりに

アニメをはじめとしてドラマや映画などでは場所の情景、キャラクターの感情に合わせたBGMを流すことによってその場面の雰囲気を盛り上げ、視聴者により印象深くより楽しんでもらおうと表現しています。

本システムを用いることで、みなさんもあたかも自分が主人公になったかのような印象深い旅を体験してみてください!!

1. はじめに

修学旅行などの団体旅行に於いて、引率者が特に気にするのは、迷子になったり、集合時間に遅れたりする人が出たりしないか、ということです。これらの不安を解消し、より楽しい団体旅行を提供するために、私たちはWebアプリケーションシステム「旅管」を提案します。

2. システム概要

本システムでは、図1に示すようなシステムを用いて、団体旅行を進行します。団体旅行の主催者と参加者は旅管のクライアントアプリケーションを使用します。クライアントはGoogle Mapsや旅管のアプリケーションサーバと通信しながら、次に示すような独自の機能を提供します。

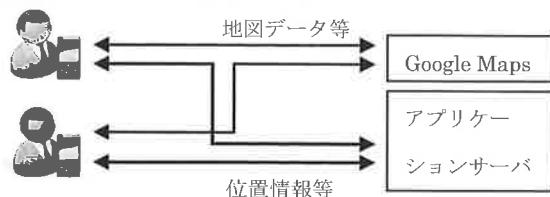


図1 システム構成図

3. 機能説明

以下に、本システムが持つ5つの独自の機能を示します。

3.1 連絡機能

旅行中、「他の人たちと連絡をとりたい」「他の人たちが何をしているのか知りたい」と思うことがあります。そのような場合に、文章や位置情報をやり取りできる強力な連絡機能があります。

3.2 集合時間の通知

団体旅行において、集合時間を守ることはとても大切なことです。集合時間に遅れてしまうと、その後の日程に大きく影響を与えてしまうからです。電車やバスなどの公共交通機関に乗り遅れなどしてしまったら大変です。その対策として、旅管には現在地と集合場

所との距離を元に移動に必要な時間を計算し、集合場所に必ず集合時間内に到着できるようにユーザーに集合を促す通知をする機能があります。これにより、団体旅行の日程を滞り無くスムーズに進行することができます。

3.3 全員の位置の把握

団体旅行に於いて、引率者の懸案事項は迷子者が出ないようにすることだと思います。旅管を使用すれば、引率者は地図に表示される、参加者の位置を表すピンを見て全員の位置を把握できるので、迷子者が出る心配もありません。

3.4 最適な集合場所の決定

旅行中に天候が急変してしまった場合など、全員がすぐに集まらなければならない状況に陥ることもあるかもしれません。そのような場合にも備えて、旅管には全員が同じ時間で到着できる、最適な集合場所を自動的に決定する機能があります。これにより、突然のトラブルにも対応することができるようになります。

3.5 アルバム作成機能

旅行から帰ってきた後に、参加者各個人のカメラや携帯電話などの写真データを交換するのは大変です。そんなときのために、旅管には旅行中にサーバーにアップロードされた画像データを纏めて、アルバムを作成する機能があります。これにより、旅行後に疲れた状態で、面倒なアルバム作成作業をすることがなくなります。

4. 終わりに

本サービスを使用することで、スムーズな団体旅行を行うことができます。ぜひ旅管を活用して、団体旅行をもっと楽しんで頂きたいです。

1. はじめに

日本の海外旅行者は先進国の中で最も少なく、この背景には言語の違いに対する不安が大きな要因となっていると言われています⁽¹⁾⁽²⁾。私たちは、外国の人ともスムーズにコミュニケーションが取れるようになれば海外旅行にも行きやすくなるを考え、コミュニケーションルール「PICT COMMY」を提案します。

2. システムの紹介

2.1 システム概要

PICT COMMY は“絵を使った会話”を行うことが出来るソフトです。図1の様に言語の異なる人にも理解できる「絵」を使って自分の意志を相手に伝えます。



図1 画面の例 (パンの値段が知りたい)

2.2 システムの利用者

本ソフトの利用者は海外旅行者です。また、このソフトを使って旅先で会話するには、現地の人にも操作してもらう必要があります。そのため外国の方にも使いやすいようにピクトグラムを採用しています。

3. 機能説明

3.1 画像検索機能

画像検索機能では、内部データベースにある約1200個の名詞の画像データ、動詞100個の動画データの中から場面に合わせた画像を表示することができます。ネットワーク環境に接続できる場合、Bing Search APIの画像検索を使って画像を表示することでネットワー

ク上の膨大なデータも利用できます。

3.2 ペイント機能

本システムに登録されている画像だけでは自分の意志が伝えられない場合、ペイント機能を利用します。ペイント機能では、タッチパネル上に触ることで自分で絵を描いて伝えることが出来ます。

3.3 ストレージ機能

海外ではこのソフトの性能を100%発揮できないことがあります。海外では“ネットワークに接続できないこと”があるからです。この問題に対しPICT COMMYは図2の様に“ストレージ機能”を実装することによって解決します。旅行の出発前に大量のネットワーク上のデータを12000個まで収集し、旅行中は集めたデータを使用して会話を行います。

ネットワーク環境あり

1200個 + 100個 + 大量

名詞データ 動詞データ ネットワーク検索

ネットワーク環境なし

1200個 + 100個 + 12000個

名詞データ 動詞データ ストレージ機能

図2 ネットワーク有無の比較

4. おわりに

コミュニケーションの幅を広げることのできるこのソフトは、ツアーやガイドブック中心の“旅のあり方”を、地元の人と会話する、生の観光地を味わえる旅に変えることができます。このソフトを使うことで海外旅行でのコミュニケーションが手軽に行えるようになり、旅行者が増加すれば嬉しく思います。

5. 参考文献

- (1) 世界銀行 『WDI Online』 2007.8.3
- (2) 国土交通省 『海外旅行者満足度(2008年)』

1. はじめに

現在この日本では少子高齢化や地方での過疎化が問題となっています。私達はこの問題を解決するきっかけとして「地元学」というものに着目しました。

「地元学」とは改めて地元を見つめ直し、地元やそこに住む人々の持っている力を引き出し、これからの生活づくり、地域づくりに活かそうというものです。

私達はこの「地元学」に基づいて Android 端末を用い、地元の方々が地元の“良いトコロ”を提供する旅、観光者とその地元の“良いトコロ”を探索する旅を実現するシステムを開発しました。

2. システム概要

本システムは、GPS やカメラ機能を備えた Android 端末を利用します。その構成を図1に示します。

利用方法はまず、Android 端末の初期起動時にユーザーIDとパスワードを入力します。その後3.で示す2つの機能の選択、実行となります。

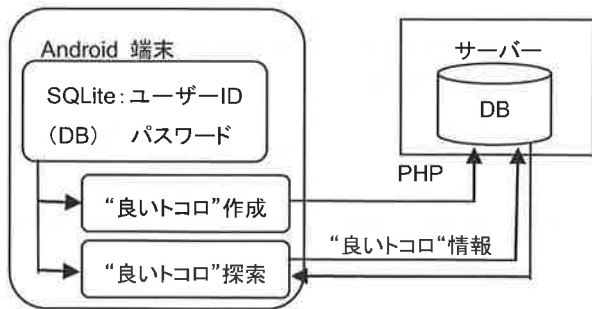


図1 システム構成

3. 機能説明

(1) “良いトコロ”作成

図2に作成画面を示します。ユーザーが他の人に勧めたい“良いトコロ”の写真とおすすめコメントを入力、どういった場所なのかというカテゴリ分けを行います。GPS より取得する現在位置と投稿日時は自動的に入力されます。これらを“良いトコロ”情報としてサーバー側のデータベースへアップロードします。

(2) “良いトコロ”探索

図3に表示画面を示します。まず、GPS より取得された現在位置情報に基づいて、サーバー側のデータベースより取得した“良いトコロ”情報の位置を現在地周辺マップ上にマーカーで表示します。ユーザーがマーカーをタッチすることで、目的地の“良いトコロ”情報やその投稿者情報を閲覧できます。そして、その目的地までの距離、方角を示したナビゲーションが可能です。また、目的地に着いた際にはその場所の評価

をすることも可能です。この評価は他ユーザーが探索を行う際の目安となります。



図2 作成画面(例)



図3 表示画面(例)

4. システムの特徴

(1) 旅ポイントの採用

本システムでは旅ポイントという得点を採用しています。このポイントは次の時に得られます。

- ・ “良いトコロ”情報をアップロードした時
- ・ “良いトコロ”探索を行い、目的地へ到着した時 (歩いた歩数に応じてポイントが計算される)
- ・ 目的地を評価した時

この旅ポイントによって、ユーザーの本システムの利用率、つまり、地元への貢献度を数値化できます。また、“良いトコロ”情報を作成したユーザーの旅ポイントを見ることで、そのユーザーの情報の信頼度が分かります。

(2) 旅ポイントによるシステムの信頼性向上

コモンユーザーは初め“良いトコロ”作成ができません。“良いトコロ”探索を行い、歩く、評価する等をして旅ポイントを貯めていきます。図4に示すように一定値以上でミドルユーザーとなり“良いトコロ”情報を作成可能となります。この仕組みによって“良いトコロ”情報の信頼性を向上させています。

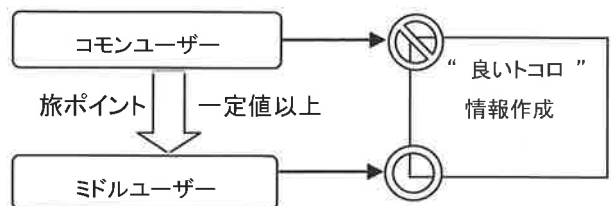


図4 旅ポイント利用について

5. おわりに

本システムを通して、地元の良さを見つめ直す旅、観光者に地元の良さを知ってもらい旅を楽しんでもらい、少しでも地域の活性化に役立てていきたいです。

1. はじめに

博物館や美術館などの展示施設を訪れた際、皆さんはどのように作品を見て回りますか？

一通り全ての作品を見回ったら、それだけでその施設の全てを知ったような気持ちになってはいないでしょうか。私たちは、本アプリを通し、ただ見て回るだけではない博物館・美術館の楽しみ方を提案します。

本アプリが重視しているのは、「作品と作品の繋がり」です。一見何も関連なく見える作品同士でも、よく見てみると何か共通点があるかもしれません。そのような共通点に光を当て、作品同士の関連性という観点から展示施設での作品観賞をサポートするアプリが、この Floor→Flow Navigator です。

2. 機能

本アプリは、Android 3.1 以降を搭載したタブレット端末上で動作する施設観覧案内アプリです。

2-1. フロアナビ機能

フロア内の作品の配置や作品の説明を表示することで、ユーザの施設観覧をサポートします。

さらに、観覧前から気になる作品のチェックリストを作成することもでき、見ておきたかった作品を見逃すことを防げます。



図 1. フロアナビ機能

2-2. リレーションマップ機能

作品同士の関連を浮かび上がらせ、ユーザに広い視野で施設観覧を楽しんでもらう機能です。

各々の作品に対し、その作品に関連する言葉(時代、人物、場所、出来事など)を「タグ」として作品と関連付けます。タグには、施設管理者が予め設定する「固定タグ」と、施設を訪れたユーザが自由に設定できる「ユーザタグ」があります。同じタグが付与されている作品同士は関連があると考え、その数が多いほど関連度が高いとします。

本機能は、このようにして求めた関連度を元に作品同士の結びつきを視覚的に表示します。



図 2. リレーションマップ機能

2-3. コミュニティ機能

同じ施設を訪れたユーザ同士で感想を交流する機能です。作品に対し、ここが良かった、ここが興味深かったなど、コメントを残すことが出来ます。

3. まとめ

皆さんも、Floor→Flow Navigator で新たな視点から施設観覧を楽しんでみませんか？

今まで気づけなかった意外な繋がりが見えてくるかもしれませんよ。

※ 図中の画像はパブリックドメインのものを使用しています。

19

旅人

—あなたと私で二人旅—

熊本(八代)

濱崎 瞳(4年) 坂上 孝平(3年)
梅川 孝佑(2年) 岩本 優(2年)
関原 至音(1年) 村田美友紀(教員)

1. はじめに

『なぜ人は「旅をしよう」と思うのでしょうか?』それはその人が旅先に興味を持っているからです。

興味は魅力を知ることから生まれます。旅先の魅力はインターネットやガイドブックを利用して知ることができますが、それらの内容のほとんどは大人向けで、低年齢の子供たちにとっては分かりにくいものです。

そこで私たちは低年齢の子供にも旅先の魅力を楽しく知ってもらえるアプリケーションソフト「旅人」を開発しました。

2. システム概要

本アプリケーションでは、図1に示す3つの機能によって各県の魅力を知ることができます。アプリケーションは端末にインストールされており、ユーザの情報は「TABIBITO」という名前のフォルダがあるUSBメモリに保存されます。

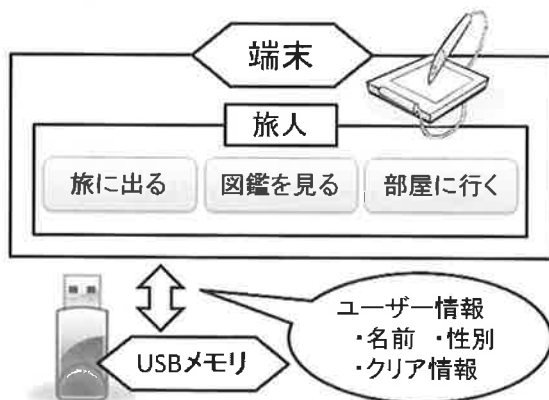


図1 システム構成図

3. 主な機能

3.1 旅に出る

日本地図から1つ県を選択すると、図2のようなマップが表示されます。ユーザはこのマップの各所においてある宝箱を開けていきます。宝箱には県の特産品や名所の情報（以下、県情報）が入っていて、ガイドキャラクターの「なつきちゃん」が紹介してくれます。

配置されている全ての県情報を集めるとゴールが出現し、その場所に行くとき県情報に関する簡単なクイズが出題されます。クイズに正解すると「県のマップをクリアした」ということになり、更に「部屋に行く」機能内で飾ることのできるアイテムを取得できます。

3.2 図鑑を見る

「旅に出る」機能内で集めた県情報を自由に閲覧することができます。図3のようにアイコンは写真なので、名前が思い出せないものでも簡単に探すことができます。

3.3 部屋に行く

「旅に出る」機能内で取得したアイテムを図4のように部屋に配置することができます。アイテムには壁紙や置物などがあり、それらを組み合わせることで自分だけの部屋を作り上げることができます。

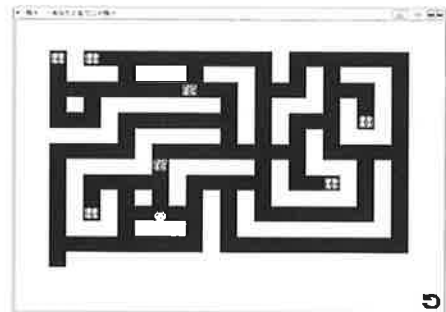


図2 マップ

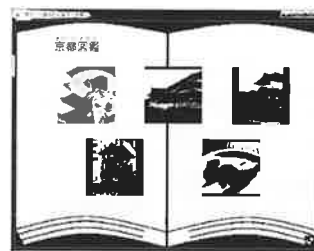


図3 図鑑



図4 部屋

4. おわりに

多くの人が「旅人」を日本各地の魅力を知るきっかけとして、実際に旅に出て、本当の『旅人』になってくれると嬉しいです。

1. はじめに

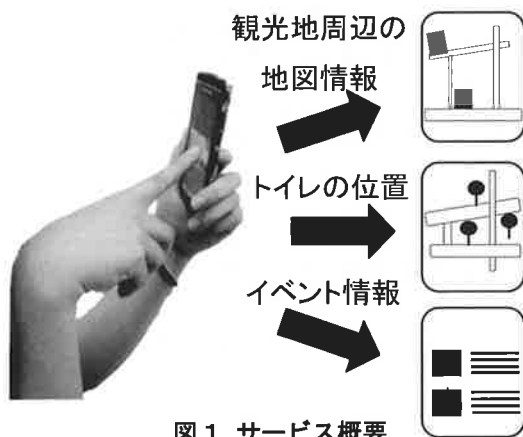
日本人および外国人観光客が旅行中に不便に思うことは何でしょうか？例えば以下のようなことがあると思われま

- ・ ガイドの情報が古い
- ・ 情報誌に掲載されていないスポットが知りたい
- ・ バス停の場所がわからない
- ・ 今いる場所の詳しい地図が見たい
- ・ 近くの観光案内所が知りたい
- ・ トイレはどこだろう？
- ・ 英語・中国・韓国語で詳しい情報が知りたい

このような問題を解消し、旅を楽しんでいただく手段として観光スポット周辺の情報を多言語でかつ、簡単に入手できるサービス「パシャナビ」を提案します。

2. サービス概要

本サービスは、スマートフォンで観光地のランドマークやポストカードを撮影することで場所を特定し、周辺の地図情報や観光地のイベント情報など旅に役立つ情報を表示します。スマートフォンに設定されている言語に応じて表示する言語が切り替わるので外国人観光客でも簡単に利用できます。(図1)



3. 機能説明

本サービスですが、スマートフォンで動作するアプリケーションと、画像検索を行うサーバ、データベースサーバで構成されており、スマートフォンアプリと各サーバはネットワークを介して通信を行います。以下に、それぞれの機能や特徴について説明します。

3.1 スマートフォンアプリ

スマートフォンアプリは撮影した写真を画像検索サーバに送信し、サーバから返ってきた候補となる画像の一覧を受け取ります。候補画像の中から画像を一つ選択すると、関連する情報が表示されます。



使い方は簡単です。

1. カメラで対象物を撮影
2. 一覧表示された画像から一つ選択

以上の操作で情報が表示されます。



3.2 類似画像検索サーバ

スマートフォンから送られてきた画像を予め登録されている画像と比較して類似度が高い順に画像を選び候補の画像の一覧を作成します。また、サーバ内の画像は撮影された天候と時間帯にグループ分けされており、スマートフォンから画像と一緒に天候や時間が送られてくれば、より高い精度で検索ができるようになっています。

3.3 データベースサーバ

データベースサーバには類似検索サーバに登録されている画像に関連する様々な情報、たとえば観光情報やWiFiスポットの場所、トイレの位置などが多言語で登録されています。

4. 終わりに

本サービスを活用することで現在地周辺の情報や、旅に役立つ情報が簡単に入手できます。これにより、多くの人の旅行がより快適になれば幸いです。

自由部門本選参加テーマ

発表順	タイトル	高専名	指導教員	参加学生
1	すまいる・りふれいん! —ボランティア情報共有システム—	八 戸	久慈 憲夫	竹林 優樹, 島守 勇毅, 笠原 圭太, 榎 優人, 竹林 雅人
2	ひろイケ!キネクト交響楽団 —指揮者・楽器演奏体感システム—	米 子	河野 清尊	花井 浩介, 山根 剛, 廣池 颯人, 貝田 祐輔, 下尾 賢二
3	micro花vision	久 留 米	黒木 祥光	武田 聖平, 竹中 孝介, 井本 至, 伊瀬知洗平, 原 拓良
4	Book Information Share —書籍情報共有—	木 更 津	大枝 真一	長沼 孝弥, 幸田 大智, 宮坂 結花, 根本 明, 池田 正隆
5	Let's クツ王 —レッツ クッキング—	徳 山	重村 哲至	小林 励志, 廣中 詩織, 有馬 諒, 近藤 佑樹, 小松 弘人
6	うえきもち	鳥羽商船	江崎 修央	木下 実優, 西岡 壮太, 西村 貫, 藤原 正希, 坂下 雄摩
7	iFarm —次世代農作業システム—	香川(高松)	村上 幸一	細野 敬太, KURISU, 井上 健太, 山根 佑樹, 清水 健作
8	習紙 —目指せ!!折り紙マスター!!—	長 岡	竹部 啓輔	長谷川侑太, 高橋 広樹, 竹内 優也
NUU	Intelligent Environment Detection System Integrating Wireless Sensor, Mobile cell, Google Map and Speech Module	國立聯合大学	Feng-Long Huang	Ming-You Zou, Shu-Nan Ke
9	contactone	沼 津	長谷 賢治	山梨 航輝, 松陰 瞭, 石原 賢太, 小澤 和也, 増田 恭介
10	Miraimi	長 野	伊藤 祥一	吉野 翼, 小林 慎治, 桐山 裕平, 渋谷みさき, 久保田実咲
11	放射能早期警戒表示システム(RWADS)	茨 城	杉村 康	京谷 繁明, 池田 綾香, 西連地礼奈, 宮本 奈緒
12	Qcloud —生活に溶け込む新世代クラウドサービス—	奈 良	松尾 賢一	小川 夏輝, 中村 匠, 矢鋪 知哉, 今吉翔太郎, 関 一至
13	ぶち・やさいふあくとりー —ステレオカメラで野菜を育てよう—	鳥羽商船	白石 和章	杉田 敢, 木下 翔太, 浅井 成充, 柴原 基希, 江口 耀平
14	iMASATO —おしゃべり冷蔵庫内管理システム—	広島商船	岩切 裕哉	當麻 美久, 奥津 尚子, 濱畑 愛香, 土井 節美, 渡辺 康太
15	いとをかし —織人しらす—	弓削商船	長尾 和彦	長尾 詩織, 山下 弘晃, 桑原 裕也, 中本 真司, 肥田 琢弥
16	Crowd Flip —群衆を取り払え!—	舞 鶴	中川 重康	古林 俊祐, 稲垣 洸雄, 高橋 洸樹, 川崎 航平, 藤原健太郎
17	space school —宇宙遊泳を夢見るあなたへ—	小 山	小堀 康功	飯島 悠介, 高平 寛之, 海老原俊輔, 福田 裕貴, 中内 宙志
18	Motion Session	沖 縄	正木 忠勝	兼城駿一郎, 宇江城貴仁, 世嘉良 紳, 藏屋沙那恵, 金城 楓太
19	All Lights! —可視光通信による省電力照明システム—	東 京	松林 勝志	大川 水緒, 田畑 愛実, 赤松 俊一, 榎原 裕章, 中川 理恵
20	HOSO! —ホットなお知らせ・すばやくお伝え—	東 京	小嶋 徹也	安藤 大輝, 菅原 政義, 松石 浩輔, 石橋 諒一, 後藤 健太

1. はじめに

3月11日に発生した東日本大震災により、多くの人々が被害を受け、さらには台風による水害などで未だ多くの支援を必要としています。ボランティア団体ではそのような被災者を支援しようと多くの人々を募集しています。

しかし、被災地では携帯電話しか使えない、ボランティアが帰ってしまうとその後は被災者との個々のコミュニケーションができない、などの問題があります。

そんなボランティアと被災者とボランティアセンター（以下ボラセン）のニーズに応え、問題を解決するものが私たちが提案する「すまいる・りふれいん！」—ボランティア情報共有システム—です。

2. システム概要

本システムは主にボランティアと被災者とボラセンを対象とし、ボランティアと被災者間の情報の共有をより簡単に行うためのシステムです。図1に示すようにサーバーにデータを保存し、WEB ページより閲覧します。

本システムにはスケジュール共有、ありがとう伝達、ブロードキャストの3つの機能があります。

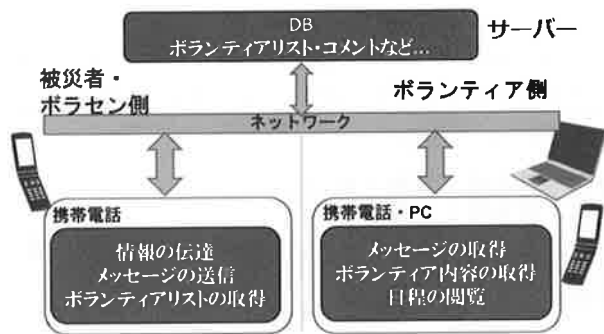


図1. システム構成図

3. 機能説明

・ スケジュール共有機能

個々のボランティア団体がスケジュールを共有して活用するための機能です。被災者・ボラセンがボランティアに依頼する仕事内容、希望人数、日時などを登録し、予定表として図2のようなカレンダーに公開します。

ボランティア側は、カレンダーから登録された内容、参加する予定の他のボランティア団体名、ボランティアの充足率を確認し、都合のいい日程を選択します。

・ ありがとう機能

被災者側がボランティアを行ってくれた方々へのお礼をWEB上で公開する機能です。伝える方法は「寄せ書き」（図3）、「音声」、「動画」から選択します。



図2. スケジュール例

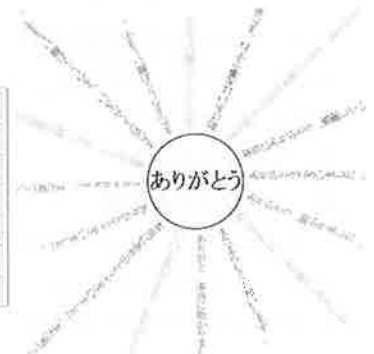


図3. 「寄せ書き」例

・ ブロードキャスト機能

このシステムに登録しているボランティア団体にメールによりボランティアを要請する機能です。これによりボランティア不足の時や大規模作業などを行うときにボランティアの人数を確保しやすくします。

4. 本システムの効果及び利点

- ・ 携帯電話を用いるため、どこでも要望を書き込むことができ、その情報を閲覧することができます。また、WEBサイトなので、ネット環境さえあればどこでも閲覧することができます。
- ・ サーバーを介して、被災者がボランティアをしてくれた方々に「ありがとう」の気持ちを伝えることができます。これがボランティア団体との心の共有、意欲の向上につながり継続的な支援を可能にします。
- ・ 被災地のボラセンに行ってから作業内容を割り当てられるのではなく、事前に予定や内容の把握が可能になり、事前準備や効率化につながります。

5. まとめ

このシステムを通して、ボランティアと被災者間での情報の共有、「ありがとう」の感謝の心の伝達が可能となります。本システムを活用し、東日本大震災や台風被害などの復興やボランティア活動の広がり役立てていきたいと思っています。

1. はじめに

世の中には音楽が好きで、楽器を演奏したり、オーケストラで指揮をしてみたいと思ってる人は多いと思います。しかし、指揮や楽器の演奏などには専門的な知識と技術が要求され、楽器も非常に高価なものが多く敷居が高いです。そこで、音楽に興味のある人であれば、誰でも簡単にオーケストラの指揮者や楽器の演奏を体験できるシステム『ひろいけ！キネクト交響楽団』（以下、本システムと呼ぶ）を開発することにしました。

2. システムの概要

2.1 システム構成

本システムは、図1に示すように、パソコン、Kinect(キネクト)およびディスプレイ(スピーカー内蔵)で構成されます。また、Kinectを動作させるために、Prime Sense社のドライバとNITEのライブラリを使用しています。

使用者はディスプレイの前に立ち、手で指揮者の動きや楽器を演奏するような動きを行います。使用者の「前腕」をKinectを用いて検出し、その動きを手の動きとして識別し、プログラムを動作させます。

曲のデータは、楽器ごとに分けてMIDIファイルで保持しておき、C++ライブラリDirectXAudioを用いて同時再生します。これによりパートごとに制御することが可能になります。

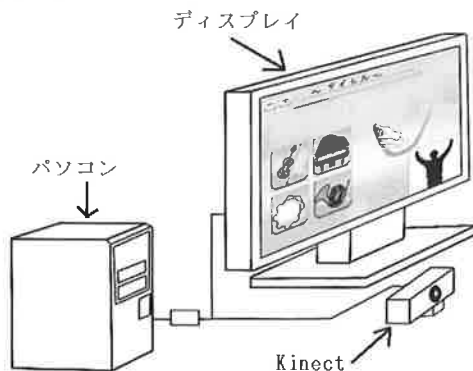


図1 システム構成

2.2 システムの機能

本システムには、指揮者を体感する『オーケストラモード』と楽器演奏を体感する『演奏モード』があります。

(1) オーケストラモード

このモードは左右の手で異なる役割を持っています。

【右手の役割】(指揮棒を持っている手として扱われる)

- ①手を頭の上まであげた状態から、振り下ろすことで演奏が開始されます。
- ②演奏中は手を振る速さで曲のテンポを調整します。

- ③手を頭の上まであげて止めることで演奏を停止させます。また手を振り始めると演奏が再開されます。

【左手の役割】

- ①演奏中に画面に表示されている楽器に手をかざすと、そのパートの楽器の音を強調します。
- ②メニューの操作や音量ボタンで音量の調整できます。



図2 オーケストラモード画面

(2) 演奏モード

演奏を体感できる楽器として、ピアノとドラムを実装しようと考えています。

ピアノの場合は、使用者が画面内のカーソルを鍵盤に合わせることで音が出るようにします。ドラムの場合は、ドラムに手をかざすことで音が鳴るようにします。他にもカスタネットやタンバリンといった打楽器の演奏もできるようにする予定です。



図3 演奏モードの画面 (左:ピアノ 右:ドラム)

3. 本システムの特徴

本システムの特徴としては、使用者の動きを認識する3DセンサーとしてKinectを用いていることが挙げられます。Kinectを用いる理由は、Kinectが人の体の動きを認識することが容易かつ安価に入手可能であることです。使用者はコントローラ等を使う必要がなく、これにより直感的な操作を実現することができます。

4. まとめ

本システムでオーケストラを体感し、音楽を奏でることを楽しんでください。

1. はじめに

ジオラマは綺麗な景色を再現した素晴らしいものです。私たちはそのジオラマに、何か動きのあるものを入れるとさらに面白いものになるのではないかと考えました。そこで、動きのあるものとして拡張現実(AR)を使用してジオラマに綺麗な花火を打ち上げるシステム「micro花vision」を制作しました。

2. システム概要

2.1 システム構成

micro花visionのシステム構成は次のようになっています。

- ジオラマ
- カメラ付きヘッドマウントディスプレイ
- AR マーカー
- 砲台型ミニチュア
- パソコン (micro花visionソフトウェア)

2.2 打ち上げの流れ

1. 設定モードでジオラマの大きさと鑑賞モードを選びます。
2. ジオラマの四隅にARマーカーを置きます。
3. 花火を打ち上げたいところに砲台型ミニチュアを置きます。
4. 鑑賞開始ボタンを押すと花火が上がり始めます。
5. 今までにない視点で花火をお楽しみください。

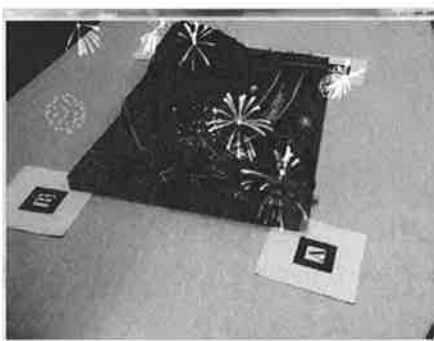


図1 花火観賞のイメージ

2.3 砲台について

花火鑑賞では、砲台型ミニチュアをジオラマ上に置

くことで花火の打ち上げ場所を決めます。砲台は花火の種類ごとに色分けされており、ジオラマ上に何個でも配置することができます。

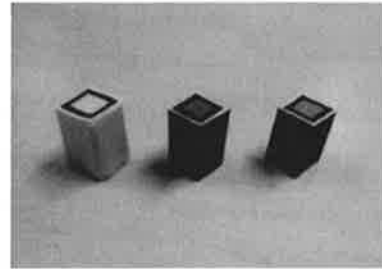


図2 花火観賞モードで使用する砲台型ミニチュア

3. 各モード紹介

3.1 通常花火鑑賞モード

通常花火鑑賞モードでは、ジオラマ上の砲台型ミニチュアから打ち上がる花火を楽しむことができます。好きな位置に砲台型ミニチュアを設置して花火大会を作りましょう。

3.2 音楽花火鑑賞モード

音楽花火鑑賞モードでは自分の好きな音楽ファイル(MIDI)を読み込ませ、その音楽に合わせて打ち上がる花火を楽しむことができます。音の音色や音程、リズムに合わせて花火の種類や色、打ちあがり方も変化します。

4. こだわり

micro花visionの制作にあたっては特に音にこだわりました。「そこに花火が上がっている」感覚を出すために、花火大会で録音した実物の花火の音を使用しています。また、打ち上がる花火の動きも実物の映像を研究して作り込んでいます。

5. おわりに

micro花visionは花火とジオラマの新しい楽しみ方を提供します。ぜひ触れてみて、小さな花火大会をお楽しみください。

1. はじめに

「この本は面白かった」「この専門書は役に立つ」などの、本を読んで思ったことは重要な情報です。それを多くの人と共有できれば便利だと考えました。また、最近ではデジタル化が進み本の利便性の理解が失われかけています。そこで私達は多くの人に本を利用してもらい、本の利便性に目を向けてもらうために Book Information Share(以降 BIS と略)というインターネット上で本の情報を共有できるシステムを提案します。

2. システム概要

本ソフトは図1に示す通り、大きく分けて3つの機能があります。(1)ブックレビュー機能：ユーザーが本の情報を Web 上にアップロードして、本の詳しい情報を他のユーザーに提供します。(2)本の相関図機能：本と本の関係性を二次元上に相関図として表示して表示します。これにより、一つの本から多くの他の本を知り、数多くの本の存在を短い時間で知ることが出来ます。(3)別メディアとの関連付け機能：本と別メディア(動画や音楽など)と趣向的に関連付けることによって、本を読む人が増えることが期待されます。



図1. BISの概要

3. 機能説明

以下に、本システムの機能を説明します。

3.1 ブックレビュー機能

ブックレビュー機能は、本を読んだユーザーが書いたブックレビュー(本についての感想)を Web 上にアップロードして、それを他の多くのユーザーに見ても

らうことが出来ます。例えば、この機能により数多くある参考書を選ぶ際に、ブックレビューを見ることでどの参考書が自分にあっているかがわかります。

3.2 本の相関図機能

本の相関図機能は本と本の関係性をわかりやすく相関図にして表示することにより、一目で多くの本の存在を知ることが出来る機能です(図2)。SF系などのジャンルで関連性を持つ本どうしは近くに表示され、ミステリー系の本とアクション系の本は余り関係性が見られないと判定され、その本どうしは遠くに表示されます。また、ブックレビュー機能と組み合わせることにより、効率的に本探しを行うことができます。

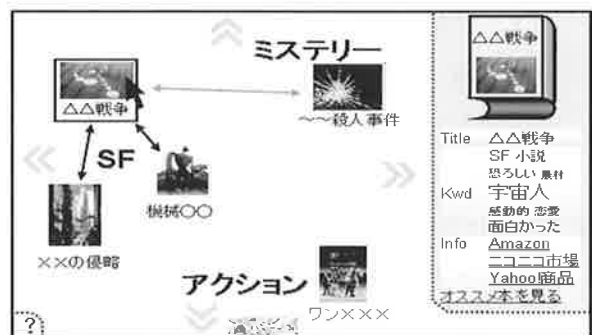


図2. 本の相関図

3.3 本と別メディアとの関連付け機能

本と別メディアとの関連付け機能では、ユーザーが「この動画が好きな人は、この本を面白いと感じるはずだ」のように、動画や音楽などの別メディアと本を関連付け、本に対して興味を持ってもらう機能です。この機能により多くの本を読むきっかけを作り、本をよく読んでいる人だけでなくあまり本を読んでいない人にも本を読む機会を与えることができます。

4. 終わりに

本システムを活用することで、本探しを効率的に行うことや、自分のものだけでしかなかった本に対する自分の感想が他の多くの人に役に立ててもらうことができます。これにより、多くの人に本のすばらしさや面白さが伝われば幸いです。

1. はじめに

おいしい食事は生きるための大きな支えである。しかし、食べることと対照的に作ることは初心者からすれば非常に難しい。そこで、料理初心者の手助けをするため、従来の料理の本の欠点である”ページをめくる動作”を解消した調理支援システム、Let's クッ王、(以下クッ王)を開発した。

2. システム概要

図1は、クッ王を利用した料理のイメージである。ユーザは、料理と調理手順を決定し、料理を開始する。



図1 クッ王使用時の概観



図2 クッ王の表示イメージ

料理を行っている最中、USBカメラが取得した画像はPCへ送られ、調理器具を認識することより現在の進行状況を把握する。そして、それに応じた調理手順がカメラ画像と合成されてディスプレイに表示される。図2に表示イメージを示す。ユーザは、表示を見て、フットペダルと音声によってクッ王を操作する。これに

よりユーザは調理から手を離す必要はなく指示に従うだけでよく、スムーズに料理を行うことができる。

画像の合成はいわゆるAR(拡張現実)であるが、マーカではなく、SURF(高速特徴量認識)を基に認識を行っている。SURFによって認識された調理器具から進行状況を把握し、適切な調理手順を表示することができる。

3. その他機能

3.1 レシピ検索・推薦機能

クッ王は、データベースから食材などの条件検索でユーザが望むレシピを探し出すことができる。また、料理初心者はそもそも何を作るべきかで悩むことがある。そのため、クッ王はレシピ推薦機能を備えており、食材の栄養価をもとに、健康的なレシピを提案する。

3.2 調理手順最適化機能

クッ王はユーザの負担を軽減するために、“器具の使い回し”、“所要時間の短縮”、“各料理の同時完成”、の3つの条件を考慮した最適な調理手順を組み立てる。料理の本では対応が難しい、複数のレシピを並行して調理するとき効果を発揮する。

3.3 レシピ追加機能

ユーザは、調理工程や分量を自分の好みに合わせて入力し、新たな料理を登録することが可能である。また、レシピサイトのURLを指定することでレシピ入力の項目を自動的に埋める機能も有している。

4. おわりに

本システムで日々の食事が彩り豊かになれば幸いである。

5. 参考文献

藤吉弘亘, 山下隆義: “物体認識のための画像局所特徴量”, 「-CVIM チュートリアルシリーズ- コンピュータビジョン 最先端ガイド2」, アドコム・メディア株式会社, pp. 1-59

3日で作る高速特定物体認識システム:

<http://d.hatena.ne.jp/aidiary/20091018/1255862734>

1. はじめに

私たちは至る所で観葉植物を目にします。しかし、「誰が世話しているのか」「いつ水をやったのか」などの疑問をもつことがあります。そこで私たちは、植物とのコミュニケーションをサポートするシステムである「うえきもち」を開発しました。また、植木鉢を通した人と人とのコミュニケーションも支援します。

2. システムの概要

本システムではAndroid OSを搭載した携帯端末を用いて現在の植物の状態を確認します。植物状態は植木鉢に挿し込まれた「うえきもちユニット」により自動的に収集され、図1に示すように画面上で状態を確認することができます。



図1 「うえきもち」の使用例

3. うえきもちユニットの構成

「うえきもちユニット」は植物が置かれている環境値を自動的に測定し、内部のデータベースへ蓄積することで、植物にとって良い状態を評価します。携帯端末は「うえきもちユニット」内のプラグコンピュータとBluetoothによって接続します。また、「うえきもちユニット」の全面に取り付けたARマーカへ携帯端末付属のカメラをかざすことによって、拡張現実として植物の表情を表示します。

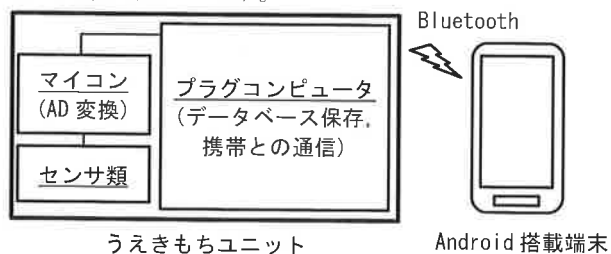


図2 システムの構成

4. システムの機能

4.1 植物とのコミュニケーション

植物の現在の気持ち（水が足りない、寒いなど）をセリフ、表情によってグラフィカルに知ることができます。この情報から水をやったり、日当たりの良いところに移動させることが可能になります。



図3 植物とのコミュニケーション画面

4.2 人とのコミュニケーション

ユーザは植物の写真を撮影し、コメントを付けて「うえきもちユニット」へ保存することができます。このコメントは「うえきもち」を利用する他のユーザからも端末を通して見ることができます。また、保存と同時にtwitterやfacebookなどのSNSへ自動投稿することも可能ですのでより多くの人に植物のお世話状況を知らせ、共有することができます。

このように「うえきもち」によって、至る所にある観葉植物とコミュニケーションをとれるだけでなく、植物が人との間にもコミュニケーションを生むツールとなります。

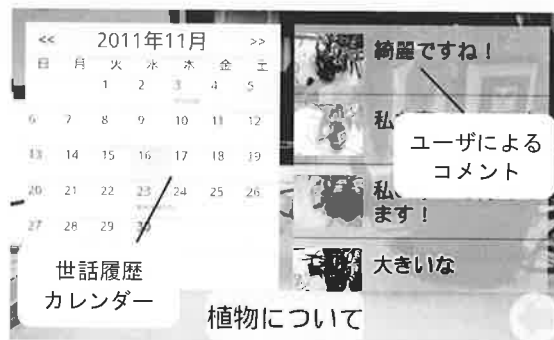


図4 お世話記録の閲覧画面

1. はじめに

現在、多くの香川県内の営農者から、圃場管理、栽培管理ソフトウェアへのニーズが寄せられている。そこで、私たちはこのニーズに応えるために「農業のIT化」をコンセプトに本システムを開発した。

2. システムの概要

本システムは、iPhone、iPadなどの通信・カメラ機能を持ったモバイル端末を利用する。圃場の写真撮影が行え、撮影時のGPS情報、気象情報と連動した農作業日誌を作成することができる。クラウドサーバでは農作業日誌などの情報を管理し、それらのデータから、農作業における被害や作業内容を営農者に知らせてくれる。

2.1. 構成

営農者が圃場での作業の後に作業内容を簡潔にまとめることが可能な農作業日誌を作成する「農作業日誌作成機能」および、農作業日誌を管理し、連作障害や追肥のタイミングなどを通知する「農作業情報管理機能」から構成される。

2.2. 機能

本システムは、以下の2種類の機能から構成される。

① 圃場管理機能

圃場管理機能とは、気象データや過去の農作業データを取り込み、連作障害の可能性や追肥のタイミング、定植時期などを携帯電話などの端末に通知してくれる機能である。

② 栽培管理機能

栽培管理機能とは、品種・種まき・定植日・除草剤・肥料などをいつ、どれくらい与えたかを記録し管理する機能である。

2.3. 処理の流れ

本システムの処理の流れは図1に示すように、携帯端末とクラウドコンピューティングを使用する。



図1 システムの概要

- ① 携帯端末で圃場の状態を撮影
- ② 撮影された画像を元に営農者が農作業日誌を作成し、クラウドサーバに送信する。
- ③ クラウドサーバで農作業日誌の管理を行い、連作障害や追肥などの農作業のタイミングを営農者に通知する。

3. 既存のシステムとの比較

本システムは iPhone、iPadなどのモバイル端末を使用することで、営農者にとって煩雑な定型データ入力の手間を省き、最低限の情報入力のみで、農作業日誌を作成することができる。モバイル端末側の機能に加え、サーバシステムを利用することにより、サーバ側での高機能・高負荷の処理を可能とする。さらに、クラウドコンピューティング、ならびにオープンソースを利用したシステム実装を行うことにより、営農者に対して既存のシステムに比べて、低価格でサービス提供を行うことができる。

4. まとめ

本システムにより「農業のIT化」を実現し、農作業の効率化を図ると共に、よりよい作物を私たちの食卓に届けることができる。

1. はじめに

折り紙は幅広い年代に親しまれている娯楽の一つです。実際に折り紙の折り方を紹介した本は多数出版されています。しかし、折り紙を折っている途中で難しく投げ出してしまいうこともあると思います。これは、折り方が本で紹介されているため、折られる過程の描写が省略されているからと考えられます。また、視点が固定されており、多面的な見方ができないことも挙げられます。

この問題を解決するには3Dアニメーションを用いた折り紙の教材を使うのが適切と考えられます。

しかし、3Dアニメーションの作成には専門の知識が必要であり、誰でも簡単に作れるものではありません。

そこで私たちは、誰でも簡単に3Dアニメーションを用いた折り紙の教材を作成し公開できるシステム「習紙」を考えました。

2. システム概要

システムの全体像を図1に示します。本システムはエディタとWebサイト、サーバに分けることができます。

エディタで作成された作品はエディタのアップロード機能によって、サーバにアップロードされ、DBに作者名などの作品の情報が登録されます。Webサイトでは作品の閲覧や作品へのコメント投稿を行うことができます。

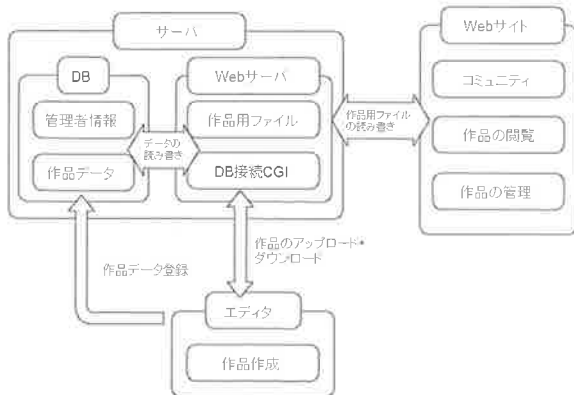


図1: システム全体像

3. システム説明

3.1 エディタ

エディタは図2のような画面となっており、折り紙アニメーション作成・編集機能、アップロード機能、アニメ再生機能の3つの機能があります。

折り紙アニメーション作成・編集機能

操作はマウスでのドラッグアンドドロップと折れ線描画の2つを基本としています。さらに、かぶせ折りや

花卉折りなど、折り紙でよく用いられる基本的な折り方が予めボタンとして用意されており、複雑な折り方もボタン1つで簡単に再現することができ、誰でも簡単に折り紙の3Dアニメーションの作成が可能となっています。また、後述のWebサイトに投稿された作品をダウンロードし、編集することも可能となっています。

アップロード機能

自分が作成した作品をサーバへとアップロードし、作品情報をDBに登録することができます。

アニメ再生機能

自分が作成した折り紙の3DアニメーションやWebサイトからダウンロードした作品を再生することができます。また、視点変更や拡大、縮小を行うことで、自分が見やすい視点で折り紙が折られていく様子を見ることができます。

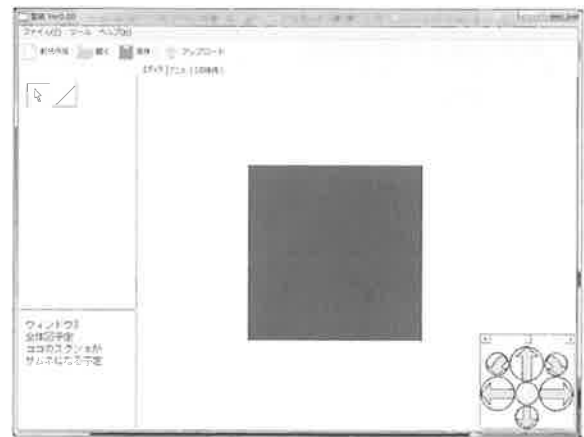


図2: エディタ画面（開発中の画面です）

3.2 Webサイト

Webサイトでは、投稿された作品を閲覧することができます。その他に、作品への一言コメントの投稿や、掲示板での意見交換を行うこともできます。また、悪質な作品が投稿された時のために、1作品に対して1回ユーザが通報を行うことができます。通報された回数が多い作品は、管理者によって削除されます。

4. おわりに

本システム「習紙」を使うことによって、いろいろな折り紙に触れ、折り紙の楽しさを学び、みんなで折り紙マスターを目指しましょう！！

1. Introduction

In current living environment, the disasters, such as fire accident, earthquake, air pollution and so on, often occur and lead to huge life being and cost loss. How to prevent and reduce the occurrence of such disasters is very important issue. In this project, we develop an intelligent disaster preventive and detective system by integrating various technologies, such as cloudy computing, wireless Sensor Networks and Googles Map.

2. System Description

2.1 The Architecture

Our system consists of following modules:

- **Wireless Sensor Networks,**
- **Googles map.**
- **Mobile Cell and**
- **Speech synthesis.**

As shown in Figure 1, all the above modules are integrated in Internet. At first, the wireless sensors will detect and collect the data in environment and pass into the gateway (front-end server) through the ZigBee protocol. Secondly, all the data in gateway will be transmitted to back-end server in remote site.

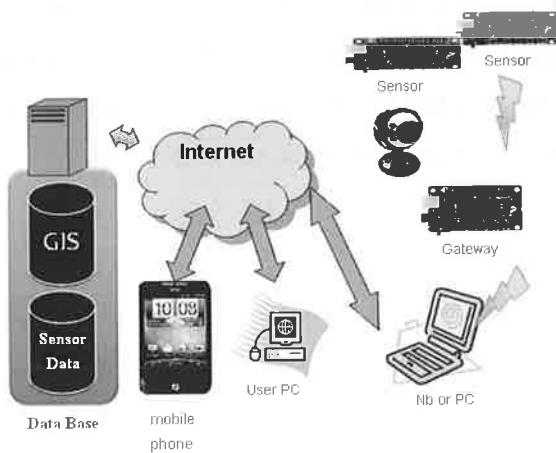


Image1. System Structure

Figure 1: System Architecture

In the back-end, there are two databases for environment and Google Map information. All the clients (users) can monitors the environment by Web Cam and browsers.

2.2 Module Functions

- 1)**Wireless Sensor Networks** : the ZigBee based sensors collect all the data and pass into front-end server.
- 2)**Googles Map** : provide the GIS information related with the detecting site and display it on the map.
- 3)**Mobile Cell**: users can control adjust the view of web cam in remote site and monitor real time the environment synchronously.
- 4)**Speech synthesis**: while an extraordinary situation is detected, the module will generate the synthesized speeches related the detecting data and notify the disaster center and some relative users. It is a cloudy computing technology.

2.3 Intelligent disaster preventive and detective system

We have developed an intelligent disaster system to prevent and detect living or foctory environment. The goal is reduce the occurrence of some disasters. In our system, ZigBee based sensors can detect kinds of environmental features, such as the earthquake, fire accident, changes in air, temperature, Geographic site, and location moving. Upon the change are detected automatically, we will notify related authorized unit or people by many manners , such speech, e-mail and web can picture, to adopt in time necessary procedures to reduce the loss.

2.4 Conclusion

In our detective system, we have developed an intelligent disaster preventive and detective system integrating various technologies. The sysetem provide visual user interface and Web Cam to enhance flexibility and prformance. We hope the system can recude huge loss in real world.

1. はじめに

私たちは日々の生活の中で手を使います。鉛筆で文字を書いたり、荷物を持ったり、何か組み立てたりとその用途は非常に多く、生活の多くは手を使うことで成り立っています。このように非常に私たちの生活に欠かせない器官である手ですが、もし手が触覚以外の感覚を感じることができるようになったら私たちの生活はどのように変わのでしょうか。そこで私たちは手に色を判別する色覚を追加し、手の機能を拡張することで日々の生活の中に音楽を取り込むことができる「ContacTone」を提案します。

2. システム概要

ContacTone では触れた色の判別を耳による聴覚を用いて識別します。つまり ContacTone が読み取った色を音に変換し、耳で違いを確認することになります。手に装着するため ContacTone の形は手袋型で、色を読み取る部分を指先に集め、手としての機能は失わないようになっています。また色を読み取るだけでなくモニターに読み取った色をコミカルに再現することで、老若男女問わず誰でも楽しんで使うことも可能です。

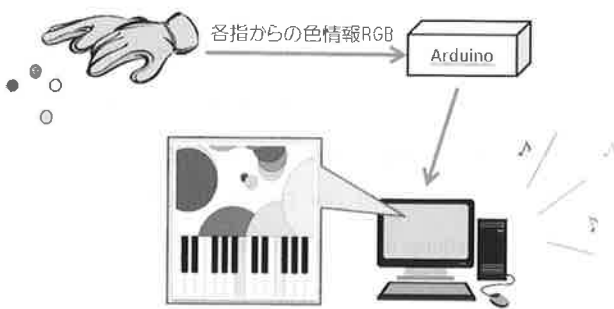


図1: システム構成図

3. 機能説明

以下に本システムの基本的な機能を説明します。

3.1 色の識別について

色の識別は ContacTone にとって最も重要な機能です。指先に取り付けたセンサーで触れた色を

RGB (Red, Green, Blue) の組み合わせで読み取り、その値から音の高さ、種類を変えることで使用者は高度な色の識別が可能です。また指先のセンサーが読み取る光は対象物の反射光だけに絞るために、それ以外の外の光がセンサーに入らないようにすることでより正確な色の情報を読みとることができます。

3.2 音の選択について

カラーセンサーから受け取った色情報から色相、光度、彩度を計算して音の高さや種類を変えていきます。

3.3 モニター上での色の再現

ContacTone では触れた色をモニター上で視覚的にも楽しむことが可能です。図1では出した音の鍵盤を、触れた色で色付けて背景にも同じようにその色の水玉を表示しています。こうすることで今自分がどんな色に触れているのか、その色はいったい何の音を奏でるのがすぐに理解できます。

4. 使い方の一例

身の回りにある様々な色のついたものを集めて簡単な演奏をするなど、楽器として楽しむ以外にも対象物を視ないで触れて、その色をモニターで確認して何を触っているかあてるゲームなどができます。

4. 終わりに

ContacTone を多くの方に使っていただき、一人でも多くの方の生活に彩ることができれば幸いです。

1. はじめに

朝、私たちは歯を磨いたり髪を乾かしたり、様々なことを行っています。そしてそれらを行うのは、ほとんどが鏡の前です。1つのことだけに時間をかけるのではなく、その時間をもっと有効に使えたらいいと思いませんか？そこで、鏡の前での時間を有効に使うべく私たちはこのシステム“Miraimi”を提案します。

2. システム構成

2.1 構成

本システムは図1のような構成となっています。

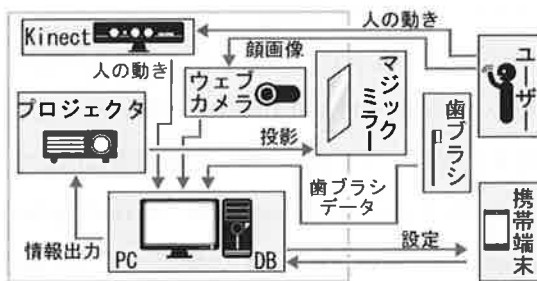


図1：システム構成図

2.2 入力デバイス

このシステムでは Microsoft が開発したデバイス“Kinect”を使用しています。Kinect は骨格などから人間の動きを認識することができ、ユーザーは自身の体だけで直感的な操作を行えます。

2.3 出力デバイス

出力デバイスはマジックミラーを用いた鏡ディスプレイです。前に立つと通常の鏡と同じく自分の姿が映ります。Miraimi が起動すると、マジックミラーの裏側に設置されたプロジェクターが映像を投影し、鏡面上に情報が映し出されます。

3. 機能

洗面台の前に立つと、様々な情報が浮かび上がる鏡が本システムの中心となります。

3.1 標準機能

本システムでは忙しい朝に焦点を当て、特に必要とされる日時、天気、ニュースを表示します。インターネットに接続することにより RSS を取得し、常に最新の情報を流すことができます。

3.2 My Miraimi

“My Miraimi”とは、鏡に表示する内容をその人に合わせて設定できる機能です。設定はインターネット接続が可能なパソコンや携帯端末を使って行います。

●予定・TODO リスト

その日の予定、やらなければならないこと(TODO)を鏡に表示させます。予定表、TODO リストは Google カレンダーを表示させることも可能です。

●表示内容の拡張

使いたい機能を追加、使わない機能を削除、表示する位置・内容を変更するなど、鏡に映るものを自由にカスタマイズすることができます。

3.3 認証方法

My Miraimi は事前に専用サイトへのアカウント登録をすることで利用できます。認証方法には①顔認証、②歯ブラシ認証の2つがあります。顔認証は鏡の前に立つだけ、もし周りに人がいるなどの場合は、専用スタンドから自分の歯ブラシを持ち上げることで個人を認識し、My Miraimi を起動することができます。

3.4 操作

Miraimi は通常、自動で起動・終了するので特別な操作は必要ありません。詳細を見る場合などはジェスチャーで操作を行います(図2)。Miraimi 上のポインタは人の手と同じように動くので、映し出されているアイコンに合わせるように手を動かすと詳細が映し出されます。また、ジェスチャーの内容によって表示内容が変わります。



図2：使用している場面

4. 終わりに

Miraimi は、洗面台などについている鏡に様々な情報を映すことで、今まで無駄になっていた時間を有効に使うことを目的としています。本システムにより忙しい朝を過ごす人々にゆとりのある朝を提供します。

1. はじめに

NHK テレビでの以下の報道をご存じでしょうか？

- (1) 3月11日 21時23分:官房長官より、福島第一原発より3km以内の住民へ避難指示が発令された。
- (2) 3月12日には、上記(1)の避難地域を3kmから20kmに変更が指示された。
- (3) 4月11日には、更に20kmを超える地域の一部を1カ月以内に避難が必要な区域に指定された。

これらは、放射能の影響から住民を守るために必要な処置と考えられます。

では上記以外の地域は安全なのでしょう？残念ながら「東京の5つの浄水場で、乳児の摂取制限を超える放射性物質検出」という3月23日のNHKテレビ報道は、そうではないことを示しています。しかし、放射能の強さは私達の目では判りません。そこで、異常があったらすぐ知らせてくれるシステムがあれば!!と考えました。

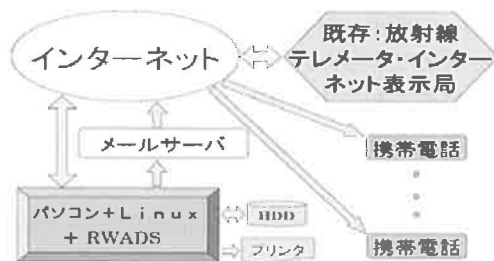


図1. システム構成図

日>	<日	月>	<月	年>	<年	時>
0時	1時	2時	3時	4時	5時	6時
10時	11時	12時	13時	14時	15時	16時
20時	21時	22時	23時	00分	10分	20分
Zone1	Zone2	Zone3	Zone4	Zone5	Zone6	Zone7
Now	Fault	Print	DL01M	DL12M	1M	GM

図2. 操作ボタンの一部例

RWADS 2011年11月16日16時00分

No.測定局	[nGy/h]	風向	[m/s]	年累積 [uGy/y]
1 石神	101	N	1.5	884
2 島岡	97	NNW	2.0	849
3 舟石川	66	NNE	1.2	578
4 押延	77	NNW	0.9	674
5 村松	80	NNW	1.5	700
6 横塚	70	NNE	0.5	613
7 門部	73	C	0.0	639
8 菅谷	67	N	1.9	586
9 浜島	66	NNE	0.7	578
10 馬淵	101	NNW	1.3	884

図3. 放射能値等の一覧表の一部例

2. 機能とシステム概要等:当放射能早期警戒表示シ

ステム(アールワッズ;RWADS;RadioactivityWarningDisplaySystem)をインストールして起動すれば、下記の

- (1)が可能となり、更にマウスによるクリックの操作(図2.参照)だけで、下記の(2)や(3)が可能となります。
- (1) 茨城県の27箇所の放射能レベルを10分毎に監視して異常等を検出すれば、指定の携帯電話等に電子メールで自動通知し、10分毎の現在の放射能レベルや風向き等を表と図(図3.,図4参照)で視認。

- (2) 上記(1)の表や図を印刷をする。

- (3) 2011年3月11日以降の任意の月日時分以前の4日間の10分毎、又は190日間の1日毎の放射能レベルを5色のグラフで自由に描いて視認や印刷をする。(「<http://www.housyasen-pref-ibaraki.jp/eartquake/doserate.html>」での公開データ」のRWADS内部データへの変換結果も使用しています。)

2.1 システム構成: 図1.の通りです。図中の放射線テレメータ・インターネット表示局のURLは、<http://www.housyasen-pref-ibaraki.jp/present/result01.html>であり、公開されています。

2.2 システム開放URL:当システムの操作説明書やソースプログラム等を無料(GPL;The GNU General Public Licence<http://www.sra.co.jp/public/doc/gnu/gpl-2j>)で、下記のサイトで開放していますので、貴方もそれらを利用できます。

<http://www.ibaraki-ct.ac.jp/ece/yas/rwads.htm>



図4. 地図や風向等の図示の一部例

1. はじめに

現在、街には数多くの公共端末が普及していますが、目的ごとに独自のシステムが個別で動作しています。さらに、提供される情報は公共性の高い情報が多く、ユーザーに合わせた情報はほとんど扱われていません。

Qloud は、生活に溶け込む新世代クラウドサービスです。情報をクラウドに統合し、ユーザーに適したサービスを提供するだけでなく、公共端末を視野に入れた設計により、既存の公共端末に置き換わる可能性を持ったクラウドサービスです。

2. システムと動作

2.1 動作環境

本システムは「Qt」で開発しているため、「Qt」がサポートするあらゆるプラットフォームで動作します。

公共端末での利用を想定しているため、タッチパネル、非接触 IC カードリーダーなどの周辺機器をベースとしたシンプルで安価なハードウェア構成で実現可能となっています。

2.2 システム構成

Qloud は、ユーザーデータを保存するネットワーク部、アプリに基本機能を提供するフレームワーク部、アプリとフレームワークを結びつける API、これらを利用して動作するアプリケーションの4つで構成されます。

2.3 アプリ形態

Qloud のアプリ形態は、以下の3つに分類されます。

- ①全画面アプリ：画面全体を占有して動作します。
- ②ウィジェット：ホーム画面で情報を表示します。
- ③エージェント：アプリの情報を通知します。

2.4 ローカルアプリ

公共端末の設置者からの情報は、ローカルアプリを通じてパブリックに提供されます。ローカルアプリは、現地で必要となる情報を提供できる、公共端末の強みを生かしたアプリです。例えば駅に設置された端末では列車の発車時刻など、現地だからこそ便利な情報を表示させることができます。

2.5 インターフェース

Qloud はタッチパネル動作を前提としたシンプルなユーザーインターフェースを備えています。



図：インターフェース外観

2.6 サードパーティによる開発

Qloud はサードパーティにフレームワークが公開され、サードパーティがアプリ開発者となってアプリを開発、公開することができます。

Qloud アプリは、QML(Qt Meta-Object-Language)と呼ばれるプログラミング言語で開発します。QML は CSS ライクな文法で UI をデザインし、JavaScript で動作を記述します。加えて、Qloud が提供する API を利用することができるため、少ないコストで高品質なアプリの開発が行えます。

2.7 クラウドサービスの統合

現在、多くのクラウドサービスがリリースされています。しかし、それらのサービスは独立していることがほとんどで、サービスが相互に連携することは容易ではありませんでした。

Qloud はプラットフォームでありながら、クラウドサービスの統合を目指した新しいソリューションです。サービスが連携し、ユーザーにとって便利な公共端末を提供することができます。

2.8 認証方法

Qloud は非接触 IC カードによる認証を行います。手持ちの IC カードを Qloud に登録するだけで、IC カードがあなたの鍵になります。

2.9 利用シーン

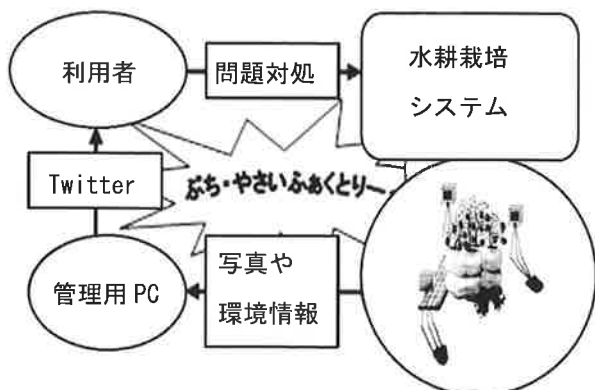
設置場所に特化した地域情報を提供する電話帳アプリ、周辺情報に詳細な地図アプリ、観光地に設置して旅行をアシストする旅アプリなど、公共端末の強みを生かした利用が可能です。

1. はじめに

家庭菜園をしたいと思っても、何をしたら良いかよく分からない、野菜の管理をする時間があまり無い、このような理由から家庭菜園を諦めていませんか？確かに、知識や経験がない初心者ではおいしい野菜を作ることはとても難しいです。そこで私たちは、知識や経験がなくても野菜を効率良く育てることができる水耕栽培システム「ぷち・やさいふぁくとりー」を開発しました。

2. システムの概要

本システムは水耕栽培キット、ステレオカメラデバイス、管理用 PC から構成されているネットワークシステムです。デバイスで撮影したステレオ写真や温度等の環境情報を、管理用 PC で解析し、抽出された問題を利用者が Twitter で受け取り、利用者はツイートの内容をもとにその問題を解決します。



3. 主な機能

3.1. 摘葉

葉っぱの量が多いと日射量や栄養が減り、収穫量の低下に繋がります。デバイスで撮影したステレオ写真から葉っぱの量を検出し、余分な葉っぱを切りとるよう画像を交えて Twitter で知らせてくれます。

3.2. 摘花

花が多いと栄養が拡散し、野菜の味が落ちてしまいます。このシステムでは花の量を検出し、多いと取るよう画像を交えて Twitter で知らせてくれます。

3.3. 害虫発見

害虫を検出します。害虫の種類と対処法を、撮影した画像を交えて Twitter で知らせてくれるので、簡単に害虫を駆除できます。

3.4. 環境情報

デバイスのセンサで室温を測定し、それらに異常がある場合に Twitter で知らせてくれます。

4. 管理用ソフトウェア

管理用ソフトウェアには3つのモードがあります。

4.1. 二視点形態

デバイスで撮影したステレオ写真と室温、育ち具合のグラフと、予定を書き込んだり記録を付けたりするためのカレンダーを表示します。



二視点形態

4.2. 一視点形態

デバイスで撮影した写真と発見した問題の種類、対処法を表示します。



一視点形態

4.3. 三次元形態

デバイスで撮影したステレオ写真から 3D グラフィックスを再現します。更に管理用 PC で 3D グラフィックスを解析し、葉の茂り具合から野菜の育ち具合を数値化します。



三次元形態

5. おわりに

この「ぷち・やさいふぁくとりー」を使用することで、システムの補助により、誰でも簡単に野菜を育てることが出来ます。あなたもこのシステムを使っておいしい野菜を育ててみませんか？

1. はじめに

あなたや、あなたの家族は、毎日の献立を考えることに苦労をしていませんか？冷蔵庫を開けてみると、材料の賞味期限が切れていたり、腐っていたり…。これは、おそらくたくさんの人が経験したことがあることだと思います。そこで私たちが考えたシステムは「iMASATO—冷蔵庫内管理システム—」（以降iMASATO）です。

2. システム概要

iMASATO は、冷蔵庫に取り付けた顔に向かって話しかけ、操作します。複雑な操作もないため、機械などが苦手な方でも簡単に利用していただけます。iMASATO のシステム構成を図1に示します。

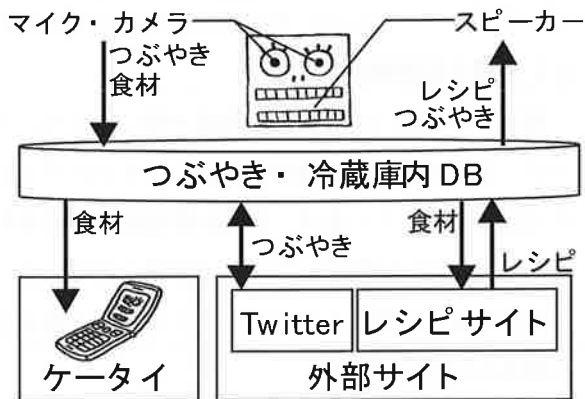


図1 システム構成図

3. iMASATO の主な機能

(1) 冷蔵庫内管理

音声によって入力され情報をもとに、食材の数量や賞味期限等を管理します。賞味期限が近かったり、食材の量が少なくなると、しゃべって教えてくれます。冷蔵庫内の食材のデータを携帯に送ることができるので買い物の時などに便利です。

(2) 離れている家族や友達とのやりとり

Twitterを通して気軽にやりとりができます。本体に内蔵されているカメラで、人物の識別が可能なので、その人宛てのメッセージを読みあげることができます。家庭内での伝言も可能です。

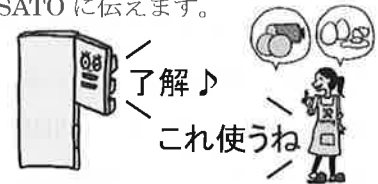
冷蔵庫が一定期間使われないと、家族などに緊急連絡ができるので、単身世帯の見守り機能としても活用できます。

(3) レシピ検索

キーワード（食材、メニューなど）をもとに、外部レシピサイトでレシピ検索できます。冷蔵庫内の食材データをもとにレシピ検索することもできます。

4. 使用例「みくさんのとある一日」

7:00 朝ごはんを作ります。使った材料をiMASATOに伝えます。



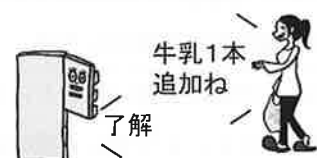
10:00 離れて暮らしているおばあちゃんをつぶやきを聞いて返事をします。



14:00 買い物に行くのでiMASATOから冷蔵庫内の食材データをもらいます。



15:00 買い物から帰ってきたので、買ったものをiMASATOに伝えます。



17:00 晩ごはんを作ります。食材データをもとにレシピを検索し、作り方をiMASATOに聞きながら調理します。



5. さいごに

iMASATOを使うことで食材の無駄を減らし、家族や友人との交流を持ち続けることにつながります。そして、食糧問題や無縁社会の問題を少しでも解決できると思い、開発しました。

謝辞

本システムでは以下のレシピサイトに協力いただきました。

E・レシピ（エキサイト㈱）、みんなのきょうの料理（㈱NHK エデュケーショナル）、オレンジページ net（㈱オレンジページ）、ホームクッキング（キッコーマン㈱）、3分クッキング（日本テレビ放送網㈱）

1. はじめに

みなさんは織物を織ったことはありますか？西陣織や緋(かすり)など今もなお多くの人々の手によって、色鮮やかな織物が作られています。しかし、いざ織物を始めようとしても難しい用語が多い、操作手順が複雑であるなどといった問題があります。

そこで私たちは、初めての人でも簡単に織物を織ることができるようにサポートする織物作成支援システムを開発しました。

2. システムの概要

本システムは、「誰でも、簡単に、楽しく」織物を楽しめることをコンセプトにしています。

織物を作成したいユーザに対して、布の模様、糸の材質、織り方などを設計できるデザイン機能、糸の材質などを考慮した作品の織り上がりを予想するシミュレーション機能、作品の織り方の手順を指示する機織り支援機能などを提供します。

これらの機能を用いることにより、ユーザの負担を大幅に減らし、より楽しく織物に親しんでいただけます。

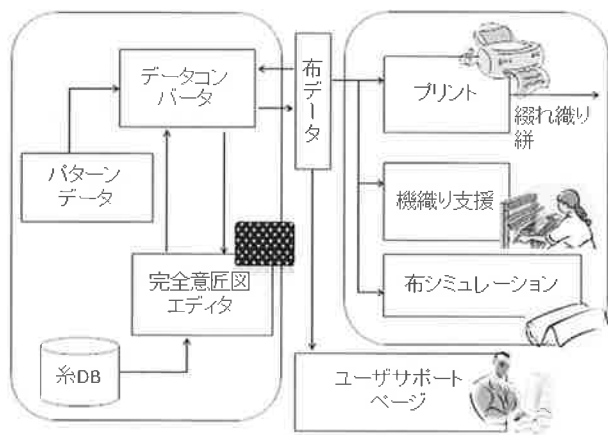


図1. システム構成図

3. 機能

3.1 デザイン機能

デザイン設計部分では、平織、綾織など代表的な織り方があらかじめパターンとして登録されており、ユ

ーザはマウスクリックの簡単操作で作品のデザインができます。選択されたパターンが実際に織れるかどうかの判断や、最適なタイアップ（綜統（そうこう）と踏み木の組み合わせ）の判断も行います。

一般的な画像から繻れ織りや緋（かすり）などの表現に変更することも可能です。

3.2 シミュレーション機能

布のテクスチャをよりリアルに再現するため、実際の糸の画像、太さ、材質などのデータベースを作成しました。これを用いて布のテクスチャを再現することで、完成イメージに近い布の表現を実現しています。

3.3 機織り支援機能

機織り支援部では、実際の機織り機に取り付けたセンサーによりユーザの操作を検出し、リアルタイムでの支援を実現しています。現在の作業状況や使用する糸の情報、完成予想図などをPCで確認することができるだけでなく、機織り機にLEDによる踏み木指示などの機能を付けています。

3.4 ユーザサポートのページ

機織りをより楽しむため、専用WWWページを作成しました。織物に関する知識や作品の投稿コーナーだけでなく、ビデオコンテンツによる機織り支援にも対応しています。

布の設計図を電子化することで、織物教室など様々な分野での活用が期待されます。

4. おわりに

奥深く楽しいものでありながら、その難しさ故に親しむ機会の少なくなった織物ですが、私たちの生活になくはならないものです。本システムをきっかけにして、織物を楽しんでみてはいかがでしょうか。

5. 参考文献

- ・はじめての手織り 小宮しげこ 日本ヴォーグ社
- ・手織り工房 彦根愛 グラフ社
- ・「織物」用具と使い方 水町真砂子 美術出版社

1. はじめに

水族館等の施設において、道に迷ったり、混雑していたりしたため時間を効率よく使えず、結局施設を楽しめなかったという方は多いと思います。そこで私たちは、施設利用者に施設内を効率よく回ってもらうためのシステム『CrowdFlip-群衆を取り払え!-』を提案します。

2. システム構成

本システムでは、各施設に設置された Web カメラの画像を定期的にサーバに送信し、混雑度を判定します。また、利用者はスマートフォンのカメラで、各施設に設置された AR マーカーを認識し、サーバに位置情報を送信します。サーバ側では、送信された位置情報と混雑度から混雑度の少ないルート进行計算し、スマートフォンのマップ上にルートを表示します。AR マーカーの使用により、GPS では精度が低い室内での位置情報の取得を可能にし、フロアの階数を含めた案内を可能にします。図 1 にシステムの構成図を示します。

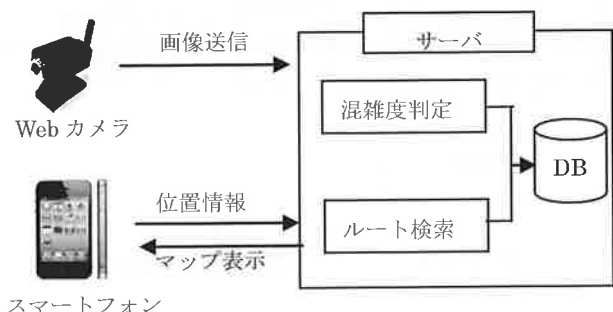


図1 システム構成図

3. 機能説明

本システムの主な機能を説明します。

3.1 混雑度測定機能

混雑度の測定は、足元が写るように設置されたカメラで行います。足元で混雑度を測定することにより、頭上からのカメラ撮影では完全に把握することができない荷物などを含めた混雑度の測定を可能にしています。また、背景の壁に複数色の線を用意することで、

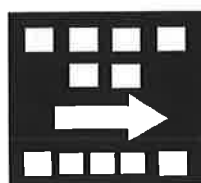
お客さんの着ている服が背景に同化してしまう場合でも混雑度の測定ができます。本システムでは、混雑度は五段階で表示されます。

3.2 屋内誘導機能

施設内に設置した AR マーカーを、スマートフォンのカメラで認識することで利用者がいる位置などを特定することにより、空いている施設への誘導が可能となります。また、利用者が施設毎に優先度をあらかじめ指定することにより、常に混んでいる人気施設にも比較的空いている時に案内することができます。利用者に対して、行先選択肢を2~3個示しますので、利用者は自由にルートを選択できます。また、利用者を各施設に分配することで、結果的に施設全体の混雑軽減につなげることができます。

4. AR マーカー

本システムでは図2に示すような独自のARマーカーを使用します。このマーカーを使用することにより、各施設の階数・方位・場所(番号)を習得することができます。そして、このマーカーは通常設置されている順路のマークと併用して使用することを目的として、基本の形を矢印のマークとしています。このことにより、本システムの利用者だけでなく、本システムを使用していない人に対してもマーカーを有効的に活用することができます。



矢印の上にある4つの正方形が番号を表します。
四隅にある正方形が方位を表します。
矢印の下にある3つの正方形が階数を表します。

図2 独自 AR マーカー

5. 終りに

本システムを使用することにより、施設を効率よく回ってもらい、最大限楽しんでもらえたら幸いです。

1. はじめに

地球ではボールを投げると、放物線を描きながら地面に落ちます。ですが、宇宙ではボールはまっすぐ一定の速度で飛んでいきます。宇宙では、今まで私たちが体験したことの無い感覚を味わうことができます。他にも直感的には解らない疑問はたくさんあるでしょう。

そこで私たちは、自分のジェスチャーで操作できるフリーソフトを使って宇宙空間を疑似体験することで、宇宙での運動力学について全身で楽しく学べる「space school」を開発しました。

2. システム概要

space school は Kinect を使った直感的な操作で、宇宙や無重力についての理解を深めるとともに物理的な思考を育てることを目的にした教育プログラムです。

3. システムの機能

「space school」には3つのモードがあります。

・ おもしろ実験モード

直感だと分かりづらい事柄を実験するとともに、分かりやすく解説してくれるモードです。このモードでは、実際に Kinect を使って無重力のシミュレーション実験を行うことで、物理法則の基本を学ぶことができます。

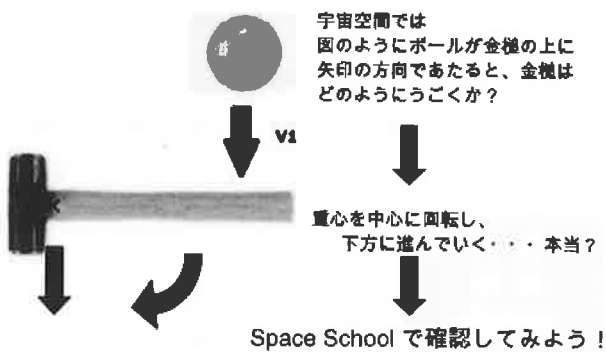


図1 シミュレーション実験例

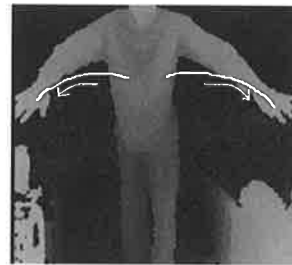
・ 問題提示モード(仮)

物理の運動法則の説明をし、提示された実験の結果がどうなるかクイズ形式で出題するモードです。このモードでは、クイズを解くことで宇宙での運動についてどれ

だけ理解しているかテストすることができます。

・ 宇宙遊泳モード

たくさんの物が置いてある仮想空間上を自由に移動できるモードです。



平泳ぎで前に進む



物体を掴む

図2 遊泳中の主な操作

4. システム構成

システムの構成を図3に示します。

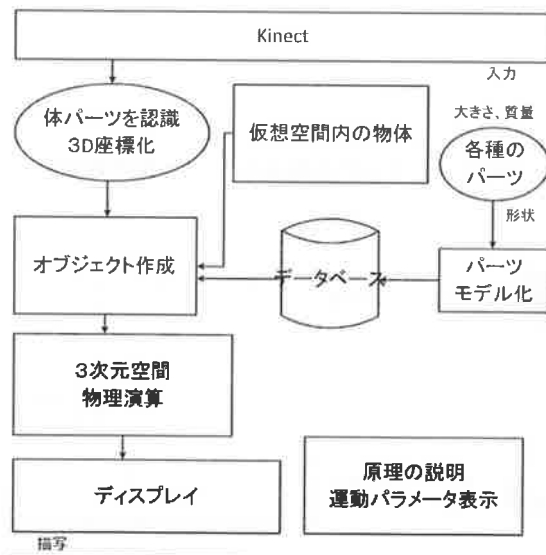


図3 システム構成

5. おわりに

今までのデバイスでは、無重力下での運動をシミュレーションすることはできても直感的に操作できないため、宇宙空間を体験するには向いていませんでした。しかし Kinect を使ったこのシステムでなら、初心者でも簡単に宇宙空間を体験することができます。

この「space school」で地球とは違った宇宙での物理法則について学びましょう。

1. はじめに

日本人の80%が楽器演奏に興味を持っている一方で、実際に楽器を保有している人は45%に留まっているといわれています。また、保有者のうち76%は自宅で一人で演奏しています⁽¹⁾。

「楽器を持っていない人や演奏経験がない人でも気軽に音楽を楽しむことができるとイネ!」「楽器演奏は複数人で行うことで本当の楽しさがわかるよネ!」と考えた私たちは、楽器演奏に興味を持った人が簡単に共同演奏(セッション)することができるコミュニケーションツール「Motion Session」を提案します。

2. システム概要

本システムは以下の3つの特徴的な機能により、楽器を持たず、演奏経験もないユーザー同士によるセッションをサポートします。

- ・ヴァーチャルプレイ機能
- ・メロディーアシスト機能
- ・タイムシフトセッション機能

システムはMicrosoft CorporationのKinectを利用しており、プレイヤーは図1に示すような構成のもと、Kinectの正面で演奏しているような動きをすることでセッションを楽しむことができます。

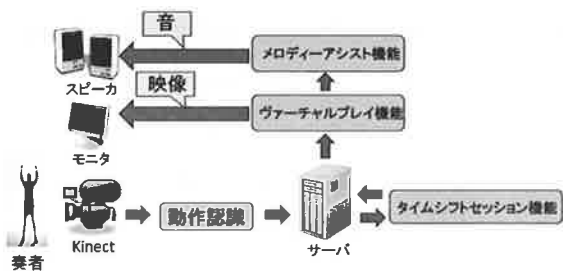


図1: システム概要図

3 機能説明

3.1 ヴァーチャルプレイ機能

Kinectによりプレイヤーのモーションを検出し、仮想音源の発音を行います。また、AR技術によって仮想

楽器画像のオーバーレイを表示します。

このことにより、楽器を買わなくても演奏している気分を十分に味わうことができます。

3.2 メロディーアシスト機能

音楽理論を用いて、発音される音程を自動算出するため、プレイヤーは練習をすることなくプレイを楽しむことができます。

3.3 タイムシフトセッション機能

サーバ上にある他者の演奏記録を読み出し、そのときの演奏と重ねての演奏を行うことができます。リアルタイムではないため、セッション相手とスケジュールを合わせる必要がなく、いつでも誰とでもセッションを行うことが可能です。

4. システム使用時のイメージ

本システムを利用している際のイメージを図2, 3に示します。図2のように何も持っていないユーザが、図3のようにディスプレイ上ではギターを保有し、他者とセッションをしている状態となります。



図2: 演奏者

図3: ディスプレイ上の表示

5. おわりに

本システムは「楽器演奏経験の有無にかかわらず、誰もが音楽そのものを楽しむことができる」ということを目的に開発しました。本システムを通して、多くの人が音楽と関わりを持つことが出来れば幸いです。

6. 参考文献

- (1)東京電力テポアレによる調査

1. はじめに

東日本大震災で電力が逼迫し、計画停電が実施されました。そのため節電に関する意識は高まってきていますが、今後も省電力を推進し、エネルギーリスクを下げていく必要があります。

そこで私たちは、オフィス、図書館など多人数が使用する部屋の照明について、インテリジェントな節電を効率よく実現するソリューション「All Lights!」を提案します。

2. システム概要

「All Lights!」は今後急速な普及が見込まれるLED照明を使い、周りの明るさに応じて適切に自動調光するシステムです(図1)。照明間での可視光通信ネットワークを構築し、調光コントローラで設定した節電率を部屋全体で達成できるように制御します。また、スマートフォンとの可視光通信により、必要なときにピンポイントで調光をすることもできます。さらに、節電状況の「見える化」も実現し、節電意欲を高め、効果的な節電の継続を目指します。

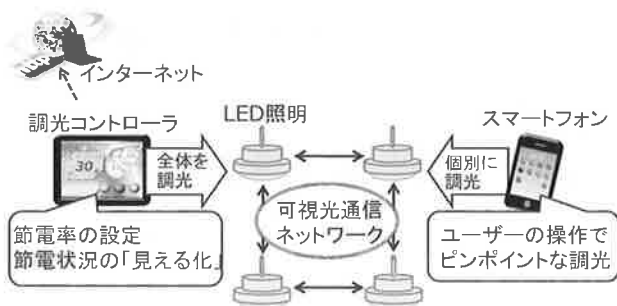


図1 「All Lights!」システム概要

3. システムの特長

3.1 可視光通信ネットワーク

「All Lights!」の通信には、照明等の可視光をそのまま利用します。照明を人間の目ではわからないほどの高周波数で点滅させ符号化を行い、データを伝送します。よって、通信ケーブルが不要で、施工コストが通常の照明設置工事と変わりません。

3.2 環境に応じた自動調光

各照明は照度センサーを持ち、窓からの採光やプロジェクタの明かりなどの「周辺の明るさ」を調光コントローラに送信します。調光コントローラでは、「周辺の明るさ」を考慮しながら、設定された節電率に合わせて照度を各照明に指示します。可視光通信ネットワークにより各照明へ指示が送られ、照度を自動調整します(図2)。これにより、窓際の明るいところでは照度を落とすなど、効率のよい節電が実現できます。

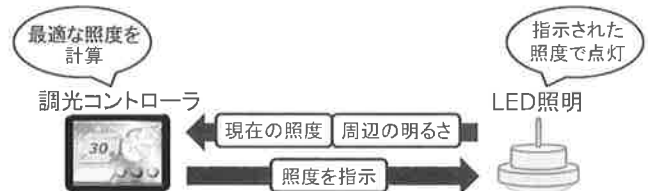


図2 調光コントローラ-照明間の通信

3.3 スマートフォンを使った調光

図書館などで、調光コントローラで節電率を設定した場合、人が読書をしている場所まで暗くしてしまう可能性があります。そこでその人がスマートフォンを使って上方の照明と可視光通信を行い、自分の周辺だけ調光できるようにします。

3.4 節電の見える化

調光コントローラより、節電率の設定、現在の節電状況の確認、過去の節電記録の確認をすることができます。この「見える化」により節電意識を高めることができます。また、企業内で調光コントローラをインターネット接続することで、企業全体で照明に対する節電が「見える化」され、節電意識を高めることに繋がり、経営管理に役立てられます。

4. おわりに

照明による節電は取り組みやすく、効果的です。現在の調光システムでは、節電は個人の努力に負うところが大きいのですが、「All Lights!」は、照明自身の光を使って照明間ネットワークを構築し、効率的な省電力を実現します。

1. はじめに

私たちは、音声放送の内容を文字情報として、スマートフォンで受信できるシステム「HOSO!」を開発しました。駅の構内放送や地域の防災放送では、周りの騒音や反響音などで聞き取れないという問題がありました。HOSO!は、音声放送を文字情報として配信することで、この問題を解決します。HOSO!は、情報を配信する駅やビルなどの施設向けの PC ソフトと、情報を受信するための Android アプリを提供します。図 1 に Android アプリの画面イメージを示します。

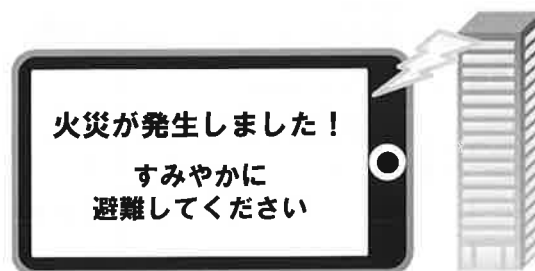


図 1 受信画面のイメージ

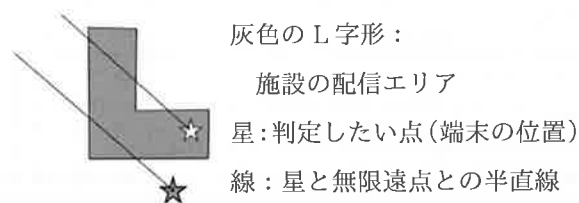


図 2 受信する端末の決定方法

2. HOSO! のメリット

ここでは、HOSO!により得られる 4 つのメリットについて説明します。

2.1 「わかりやすく」伝わる

HOSO!を用いると、音声だけで伝えられていた情報を文字で得られるようになります。文字情報は、自分の速度で読むことができ、繰り返し読むこともできるので、確実に情報を手にすることができます。画像を付加して配信することもできるため、避難経路図など、従来の音声放送では伝えづらい内容も伝えられます。

2.2 「どこにいても」使える

HOSO!は、各施設がそれぞれ配信を行うエリアの情報と、Android 端末の位置情報とを利用し、施設の配信エリア内にある端末だけに情報を配信します。

任意の多角形（施設の配信エリア）に対し、ある点（端末の位置）が多角形の領域内に含まれるかどうかは、判定したい点から無限遠点までの半直線を引き、半直線と多角形との交点の数が奇数か偶数かで判定できます。図 2 の例では、白い星での半直線とエリアの境界との交点は 3 つで奇数なのでエリアの内部、黒い星では 2 つで偶数なのでエリアの外部だとわかります。

配信者と受信者を中継するサーバが以上の判定を行うため、受信者が移動しても現在地の設定などをする必要はありません。近くにある施設が配信した情報だけを自動的に受信できます。

2.3 「かんたんに」使える

受信者は HOSO!を使う上で特別な操作をする必要はありません。アプリを一度起動すると自動で設定が行われ、システムを利用できるようになります。アプリが情報を受信すると、ユーザに音や振動で知らせて、画面に配信内容を表示してくれます。必要に応じて、文字のサイズや色を調節することもできます。

受信ユーザの管理はサーバがまとめて行なっているため、施設が情報を配信する際には、内容をサーバに送信するだけで、近くの端末に情報を伝えることができます。

2.4 「すぐに」届く

HOSO!は、サーバから Android 端末へ情報をプッシュ通知するための C2DM という技術を用いています。これにより Android アプリの消費電力が抑えつつ、リアルタイムな情報の配信を実現しています。また C2DM はインターネット回線を利用するため、非常時にはつながりづらい電話やメールを用いるより、安定して動作します。

3. おわりに

技術自体が直接人間を幸せにはできません。しかし技術によって「防げる不幸を防ぐ」ことはできます。HOSO! は、そのための一角となりえるシステムです。

応募全テーマ

■ 課題部門 「旅とコンピュータ」

タイトル	高専名	指導教員	参加学生
Minsyuku —簡単 宿借り、宿作り!—	有 明	松野 良信	羽瀬 太二、永見 友之、西 広規、後藤 隆文、山口 賢二
旅管 —“イマドキ”の団体旅行“超”管理システム—	沼 津	鈴木 康人	加藤 史也、庄司 直樹、湯山 孝雄、BALJINNYAM
SnapSwapTrimap —携帯端末による景観再現：創造—	米 子	河野 清尊	梶間 光法、亀井千佳良、鷺見 知洋、近藤 醇、田中 雄介
地域まとめしよん	鳥羽商船	江崎 修央	島影 瑞希、小林真佐大、NOROVERDENE BATZAYA、服部龍太郎、野村 昭好
旅標 —歩いた後に道はできる—	茨 城	吉成 偉久	齋藤 奎悟、川又 慎也、小泉 慧太
スマートアンブレラ	香川(高松)	重田 和弘	矢野 正人、助安 涼、米井 裕紀、田口 拓明、長谷川雄太
IRI-愛莉 —IntentRouteIntegrator—	津 山	宮下 卓也	妹尾 大地、寺尾 奈々、国定 凌太、奥村 直裕、八田 直也
旅行プランネット	有 明	松野 良信	野田 武司、川添浩太郎、都 嘉恋、國崎 恒成、熊手 圭
旅Navi —これ一つであなたの旅を快適サポート—	木 更 津	坂元 周作	秋葉 匡伸、七条 聖哉、富重 博之、則友 一磨
Attra-Collector	木 更 津	丸山 真佐夫	山田裕太郎、柳町 天使、石井 大貴、田口 博史、高田 凌平
PICT COMMY	舞 鶴	中川 重康	辻 春樹、陸山海一郎、芦田 陸、大利 晃司、中田 康貴
francDressing —おしゃれから始める旅行計画—	豊 田	神谷 直希	堀野亜里沙、前田 進吾、BILIGZAYA CHINBAT、岡本 亮輔、豊田 裕智
Image LynX	鈴 鹿	青山 俊弘	佐田 薫士、東口 里菜、宮崎 大輔、森上 諒也、油田 笑実
たびどあ!	石 川	越野 亮	林 真利奈、孫田 耕作、川端 優紀、山田 貴之、月田小百合
旅札	福 島	島村 浩	赤塚 篤、増子 宏大、高橋真奈加、齋藤 進成、小林 尚輝
方言支援なまライザー	仙台(名取)	北島 宏之	畠山 巧幹、安藤 稔、大泉 蓮、草野 慶太、本郷 卓也
卓上旅行	金 沢	館 宣伸	成清 敬太、中野 航実、作田聡二郎、村田 和馬、小町 凌司
旅の青春ポイント	和 歌 山	竹下 慎二	木下 拓也、夏見 勇矢、小林 祐介、梅崎 皓太、勝田 敏郎
MAPHIS —歩いて広がる歴史の世界—	松 江	福岡 久雄	荒川ひかり、難波 隼也、津田 達也、素手 一平、内田 謙大
旅案内人 —コミュニケーションのある旅を目指して—	明 石	濱田 幸弘	小林 誠、NGUYEN ANH TUAN、藤田 隆寛、江副 達也、荒木 日恵
とらりす —ノートを書き終わるまでが旅なのだ—	舞 鶴	中川 重康	新保 智喝、小林 賢太、金澤 朋寛、高山 凱久、梶井 良太
ひすとら!!	松 江	藤井 諭	森脇 正典、錦織 憲、伊輪 美嬉、楨原 宏志、松田 朗人
MUSIVEL —TPO JUKEBOX—	長 岡	竹部 啓輔	小林 樹矢、高橋 悟、武田 修平
ぐるりん —スマートフォンで観光情報を簡単発信—	熊本(熊本)	島川 学	白井 武史、加藤 孝史、坂本 譲二、大和 康平
旅ズバツ!	奈 良	岩田 大志	麻田 優真、上岡 真也、梶原 雄士、島岡 大遥、恒川 凌
ぶび☆づけ —京都でRPG—	阿 南	岡本 浩行	日下 晃佑、佐藤 栄一、前田 佑太、山本 優輝
BOLT —B0dy Language Translator—	新 居 浜	先山 卓朗	稲田 利亀、新宅 隆弘、飛鷹 智浩
たびたび動画 —旅の思い出をあなたも—	小 山	小堀 康功	長山 知司、戸井田 理、寶塚友希菜、菅沼 勇也、柴田 誠
進化を旅する —あなたのわたしのその指で—	小 山	平田 克己	青木 光、都井 大樹、寺田 圭佑、黒崎 大聖、日向野尚也
旅人 —あなたと私で二人旅—	熊本(八代)	村田 美友紀	濱崎 瞳、坂上 孝平、梅川 孝佑、岩本 優、関原 至音
散歩リングロード	久 留 米	黒木 祥光	井上 雄登、森 唯人、内田 聖也、田中大志郎、福永 隼也
パシャナビ! —現在位置確認システム—	金 沢	藤澤 武	堀岡 瞳、花谷 恵里、多田 朋生、長谷川一平、浅井 一真
偉人道 —先人に学ぶ自分探しの旅—	香川(詫間)	宮武 明義	白井 智也、三木 脩弘、宮武 順平、元木 祐介、白井 裕夏
カンファイン —簡単カンファチェックイン—	長 野	伊藤 祥一	奥本 隼、若林 哲宇、宮崎 大智、大和 優介、矢口 裕也

■ 課題部門 「旅とコンピュータ」

タイトル	高専名	指導教員	参加学生
TRGP —TravelerRoadPlayingGame—	福井	斉藤 徹	冨田 一貴, 青山 昂平, 玉木 義孝, 山腰 貴大
旅クエスト	宇部	武藤 義彦	松ヶ下浩司, 上田 悠希, 益永 鈴乃
TRIPPLANNER —理想の旅行プランを—	香川(説問)	宮武 明義	石村 大海, 川江 遼太, 吉田 圭佑, 岡田 祥太, 岡野 有里
コトまき —まこう コトバー	新居浜	占部 弘治	藤田 晴樹, 伊藤 大智, 吉田 晟佑, 今井 大樹, 高橋 俊亮
僕らのShareWalk —地元の良さを知る旅支援システム—	八戸	釜谷 博行	坂上 拓哉, 山下 和志, 菅 貴志, 釜谷 諒悟, 花牟礼茂人
ろ〜くまっぴ	岐阜	廣瀬 康之	石丸 翔吾, 白石 祐輝, 長尾 和暁, 澤 俊輔
ぼけつときんぎょ	鈴鹿	田添 丈博	市野 晴之, 岩田真次郎, 喜多 悠太, 渡辺 謙太
あやかし4U —妖怪と旅しよう—	米子	松本 正己	二岡 健道, 木下 和音
ShuWord —手話で旅するお手伝い—	沖繩	神里 志穂子	比嘉 早紀, 上江洲有希, 米須 樹, 大城 聖也, 比嘉健太郎
Floor→Flow Navigator	一関	小保方 幸次	吉田 琢朗, 館崎 優人, 大関 啓太, 佐々木拓也, 鈴木 拓弥
ぶらぶら旅ナビ	一関	小保方 幸次	及川 遥, 高橋 大智, 高野 啓哉, 高橋 源基

■ 自由部門

タイトル	高専名	指導教員	参加学生
放射能早期警戒表示システム (RWADS)	茨城	杉村 康	京谷 繁明, 池田 綾香, 西連地礼奈, 宮本 奈緒
contactone	沼津	長谷 賢治	山梨 航輝, 松陰 瞭, 石原 賢太, 小澤 和也, 増田 恭介
e-move	沼津	長谷 賢治	森 暁信, 内藤 広海, 小松 潤也, 竹下 航平, 小澤 洋樹
うえきもち	鳥羽商船	江崎 修央	木下 実優, 西岡 壮大, 西村 貴, 藤原 正希, 坂下 雄摩
Second Memory —大切な思い出を残したい君へ—	有明	松野 良信	木村 優一, 立石 潤矢, 河野 拓海, 乗松 聖矢
さんぷおと —looking for picture—	有明	松野 良信	西原 祐貴, 山口 翔平, 野口真菜美, 伊藤 綾奈, 伊藤実沙都
いとをかし —織人しらす—	弓削商船	長尾 和彦	長尾 詩織, 山下 弘晃, 桑原 裕也, 中本 真司, 肥田 琢弥
ひろイケ!キネクト交響楽団 —指揮者・楽器演奏体感システム—	米子	河野 清尊	花井 浩介, 山根 剛, 廣池 颯人, 貝田 祐輔, 下尾 賢二
TagFM2011	弓削商船	長尾 和彦	檜垣 俊希, 竹田 賢人, 瀬尾 敦生, 笹野 夏輝
Let's クツ王 —レッツ クッキング—	徳山	重村 哲至	小林 励志, 廣中 詩織, 有馬 諒, 近藤 佑樹, 小松 弘人
iFarm —次世代農業作業システム—	香川(高松)	村上 幸一	細野 敬太, KURISU, 井上 健太, 山根 佑樹, 清水 健作
iMASATO —おしゃべり冷蔵庫内管理システム—	広島商船	岩切 裕哉	當麻 美久, 奥津 尚子, 瀧畑 愛香, 土井 節美, 渡辺 康太
Newbe —ルービックキューブを用いた新しい入力デバイス—	豊田	安藤 浩哉	稲葉 祥, 大崎 裕貴, 神戸 大亮, 玉山 帆祐, 吉田 涼
アイビク	石川	越野 亮	藤江 拓哉, 石黒 健太, 木戸口里美, 前田有美子, 木村 優作
Spots	石川	越野 亮	山本 晃平, 武田 能里, 嶋津 芝織, 山元 翔太, 山村 大地
All Lights! —可視光通信による省電力照明システム—	東京	松林 勝志	大川 水緒, 田畑 愛実, 赤松 駿一, 榊原 裕章, 中川 理恵
Miraimi	長野	伊藤 祥一	吉野 翼, 小林 慎治, 桐山 裕平, 渋谷みさき, 久保田実咲
iMat —魔法の絨毯—	熊本(熊本)	孫 寧平	吉川 慧, 徳永 勇介, 山田 洋二, 長岡 翔, 長岡 博
Nocorupe! —のーは で あそび かそーくーかん!—	木更津	東 雄二	増田 慎平, 赤松 伸樹, 伊藤 隼斗, 白石 空
電腦指遊 —えれきぶいんがぁ—	金沢	田村 景明	中山 義崇, 澤田 友樹, 黒澤 清和, 小田 翔一, 山崎 寛人
ぶち・やさいぶあくとりー —ステレオカメラで野菜を育てよう—	鳥羽商船	白石 和章	杉田 敢, 木下 翔太, 浅井 成充, 柴原 基希, 江口 耀平
ざ・ひみつきち	松江	和田守 美穂	持田明加奈, 松本 真季, 堅固 潤也, 西川 仁将, 野津 佑太

自由部門

タイトル	高専名	指導教員	参加学生
アシスタントデザイン —直感的なwebページ作成支援ソフト—	豊田	岡部 直木	浦部 知成, 柴山 和哉, 鈴木 健文, 藤原 知紀, 青木 吾一
Book Information Share —書籍情報共有—	木更津	大枝 真一	長沼 孝弥, 幸田 大智, 宮坂 結花, 根本 明, 池田 正隆
習紙 —目指せ!!折り紙マスター!!—	長岡	竹部 啓輔	長谷川 侑太, 高橋 広樹, 竹内 優也
すまいる・りふれいん! —ボランティア情報共有システム—	八戸	久慈 憲夫	竹林 優樹, 島守 勇毅, 笠原 圭太, 柳 優人, 竹林 雅人
Mr. Johnny	鶴岡	内海 哲史	高橋 悠希, 後藤 拓人, 斉藤 飛翔, 柴田 歩
Kinectで織り成す影絵の世界	熊本(熊本)	島川 学	池田 幸平, 今坂 雅輝, 宮本 洋季, 原田 奈央, 中神 楓子
AirEditor —空間落書き帳—	阿南	小林 美緒	筒井甲子郎, 里谷 佳紀, 居石 峻寛, 清重 飛向, 宮崎 賢治
Travic!	新居浜	占部 弘治	菊池 佑, 三村 正樹, 岩本 悦子, 小池 豪人, 壺内 海
Qloud —生活に溶け込む新世代クラウドサービス—	奈良	松尾 賢一	小川 夏輝, 中村 匠, 矢鋪 知哉, 今吉翔太郎, 関 一至
Air2 Hockey	鈴鹿	箕浦 弘人	久留美祥平, 位田 祐基, 片山 勇次, 端無 祥真, 平野 将崇
LCL —Logic Circuit Learning—	旭川	森川 一	澁谷 賢一, 谷口 翔太, 篠村 堯史
Elementary SCHO2L —化学嫌いなあなたへ—	茨城	弥生 宗男	矢野倉伊織, 八重樫拓也, 高橋 雄太, 磯前 貴央, 佐久間夏輝
Let's多読! —英語の森への招待状—	徳山	高山 泰博	川部 純, 岡崎 香, 茂山丈太郎, 三浦 美奈, JOHN NG'ANG'A
prock(ブロック) —ブロックで学ぶプログラミング—	小山	平田 克己	杣木 芳成, 青木 大地, 鈴木 藍雅, 陣在 遼河, 押山 浩之
micro花vision	久留米	黒木 祥光	武田 聖平, 竹中 孝介, 井本 至, 伊瀬知洗平, 原 拓良
space school —宇宙遊泳を夢見るあなたへ—	小山	小堀 康功	飯島 悠介, 高平 寛之, 海老原俊輔, 福田 裕貴, 中内 宙志
idea Explorer —あのひと, 探します—	熊本(八代)	岩崎 洋平	岩本 舞, 桐 峰太郎, 池田 泰朗, 木村 匠, 石川 貞憲
EveryoneWorld —人間と生物との関係について—	熊本(八代)	岩崎 洋平	窪田 一平, 平岡 亨, 植柳 実, 井王 佐, 上田 尚人
べじたん —新・生きる野菜—	岐阜	山田 博文	河村 祐樹, 北川 淳嗣, 林 祐太, 吉田 智晴
Crowd Flip —群衆を取り払え!—	舞鶴	中川 重康	古林 俊祐, 稲垣 洗雄, 高橋 洗樹, 川崎 航平, 藤原健太郎
Motion Session	沖縄	正木 忠勝	兼城駿一郎, 宇江城貴仁, 世嘉良 紳, 藏屋沙那恵, 金城 楓太
TOUCH de DOYA!! —ふれる!新感覚ミュージック—	鈴鹿	浦尾 彰	亀井 優一, 坂上 浩太, 篠本 涼, 田中 聖真, 土田 佑慈
ReFinDer —無線通信を用いた物品管理システム—	和歌山	村田 充利	野村 隼人, 藤原 尚樹, 仲 泰正, 家柳 俊平, 吉田 海優
医師のみぞ知るセカイ	新居浜	平野 雅嗣	加賀城 充, MICHAEL, 真鍋 匠
CROSSCAN —クロス・スキャン—	香川(説間)	金澤 啓三	大倉 大樹, 正木 京, 森 健悟, 柳生 知志, 向山 虹祐
PONFF(ばんぷ)	津山	寺元 貴幸	梅原 奈穂, 田淵 皐, 末田 卓巳, 濱本 幸輝
CLATCH —CLAss x waTCH—	久留米	黒木 祥光	北川 巧, 宮原 康成, 中野 純利, 中尾 收, 岩本 海童
HOSO! —ホットなお知らせ・すばやくお伝え—	東京	小嶋 徹也	安藤 大輝, 菅原 政義, 松石 浩輔, 石橋 諒一, 後藤 健太
画伯 —こんな感じ?を伝えたい—	米子	松本 正己	阿部宗一郎, 山西晋太郎, 花倉 啓語, 永本 悠祐
見えるんです! —目視難所を観察するシステム—	苫小牧	三上 剛	藤田 光隆, 林 隆介, 下野 優作
まごトピッ!	松江	渡部 徹	加藤 涼子, 伊藤 千晶, 金津 穂, 岩田 翔星, 山本 千春
「天井文庫」	一関	千田 栄幸	北條 裕, 伊藤 宙夢, 小山 詩織, 佐藤 守任, 齋藤 真慶
Touch+Play=Study —触って遊んで勉強マスター—	一関	千田 栄幸	菊池 敬濟, 山谷 佳祐, 菅原 毅, 村上 隆明, 坂本 優作
Mind Indicator —ジェスチャーtoVoice!—	富山(射水)	篠川 敏行	北川 凌雅, 鶴山 北斗, 堀野 洋希
みんなでえいたんご漬け —英語が苦手な人のPCTレーニング—	サレジオ	内田 健	富 英一, 由比麻梨乃, 野部 愛, 西村 圭, 松本 拓也
eTotal —授業を楽しくする次世代双方向授業支援システム—	高知	今井 一雅	岡田 直也, 山田 嗣郎, 前田 峻, 笹岡 勇佑

競技部門本選参加テーマ

登録順	タイトル	高専名	指導教員	参加学生
1	ヴァッサーマンV2	秋 田	竹下 大樹	真壁 純矢, 佐々木貴晃, 渡辺 拓也
2	Mスタンプ	沼 津	出川 智啓	坂代 一弥, 斎藤 秀人, 金森 広樹
3	Watson On Stage	長 野	伊藤 祥一	渡辺 大瑛, 佐藤 俊之, 大平 哲也
4	きゅーかんぱー ー僕たちのきゅーりー	熊 本(熊 本)	孫 寧平	小中 俊樹, 木村 慎, VONGSOMXAI VILAYOUTH
5	ぶらり遺跡探検の旅	木 更 津	白木 厚司	平井 成海, 金巻 俊一, 東 遼平
6	Building Repairer	香 川(詫 閤)	宮武 明義	渡邊 裕輝, ANANDA ABDULLAH, 日野晋太郎
7	Φερβικη(ふあいらにさん) ー勝利を導く者ー	津 山	岡田 正	小山 竜矢, 神崎 拓人, 石谷 沙央
8	やげまぶもらと ー「ぶちりよんこ」ってなんだー	呉	藤井 敏則	平井 勇大, 山本 和空, 山根裕一郎
9	イイヨ!ゼンゼン!! ー壊しても修復やんぜ!ー	弓 削 商 船	長尾 和彦	中津 諒, 宇崎 裕太, 田内 混己
10	Stamp Stamp Stamp	鳥 羽 商 船	江崎 修央	AMARBAYASGALAN NYAM, 岩井 駿, 井上 修輔
11	Wish the vanishing ー消したい過ちー	近 畿 大 学	政清 史晃	林 健太, 中野 弘基, 尾上 貴子
12	ホテルナカヤマ ー304号室の中のリアルー	金 沢	元木 光雄	宅美 貴裕, 真田 純, 松永 隼季
13	Let's put stamp	岐 阜	廣瀬 康之	兼松 良輔, 世古 潤平
14	アドミタンスY ーインピーダンスの逆襲ー	石 川	越野 亮	本多 達也, 越田 和基, 寺本 大輝
15	どぞぞすこすこどぞぞすこすこ ー聖なる夜の、開・暮・だ!ー	舞 鶴	芦澤 恵太	竹野 峻輔, 岡井 晃一, 中川 慎哉
16	佃煮	広 島 商 船	佐久間 大	佃 将樹, 藤原 楓, HADINATA IGNATIUS STEVEN
17	そんなスタンプで大丈夫か? ー一番いいスタンプを頼むー	都 立(荒 川)	鈴木 弘	宮崎 亮輔, 平間 章裕, 三田 和広
18	ヨミ ー愛と勇気の修復マスターー	徳 山	三浦 靖一郎	中島 淳平, MUNKHJARGAL TUGS-ITGELT, CHINDEGSUREN CHINBAT
19	Restore ー失うことが出来るだろうか遺産のその景観をー	八 戸	細川 靖	大岩 知也, 越後谷龍之介, 木村 拓也
20	Revive	小 山	南斉 清巳	松本 恭平, 山上 宇宙, 小暮 裕樹
21	スタンプ・ザ・UD	大 島 商 船	神田 全啓	馬場 信彰, 重本 昌也, MAUREEN
22	うわっ…私の手数、多すぎ…?	和 歌 山	森 徹	櫻井 琢士, 村上 孝則, 高垣 翔平
23	とりあえず復刻的な事をしたいと思います。	北 九 州	白濱 成希	河本 和也, 田川 智也, 早川 玲央
24	Faster Pusher	明 石	濱田 幸弘	西村 友佑, 羽田 哲也, 柏木まもる
25	Kontravalenz ー1+1は10じゃないよー	大 阪 府 大	窪田 哲也	稲田 真吾, 浦 義裕, 三上 和馬
26	TRIAL AND ERROR	函 館	小山 慎哉	BATZORIG TUVSHINTUGS, 伊藤 明宏
27	えふえふしいしい ーフリップフロップサーキュラーサーキットー	仙 台(名 取)	北島 宏之	佐藤 高明, 藪田 樹, 山本 里香
28	TRIPPER	豊 田	岡部 直木	赤川 卓也, 杉田 篤紀, 藤井 直道
29	終わらないコーディング ーバグもエラーもあるんだよー	沖 縄	正木 忠勝	日熊 悠太, 呉屋 寛裕, 名渡山夏子
30	思い出のアルバム ーデジタルリマスター版ー	苫 小 牧	三上 剛	麻野 将平, NAIDANJAV ZOLBAYAR, 芳賀 雄基

登録順	タイトル	高専名	指導教員	参加学生
31	スタンプあ	奈良	山口 賢一	矢野 完人, 上田 健太, 秋末 真志
32	世界遺産は直せる!	長岡	竹部 啓輔	樋口 悠生, 渡辺 翔, 佐藤 健介
33	THE 修復	茨城	安細 勉	鬼沢 亮, 鈴木 一哉, 千葉 一貴
34	2値画像と復元機	阿南	一ノ瀬 元喜	NAMU, 天狗石 悠斗, 櫛田 佳那
35	世界遺産調査中。 —本日は晴天なり—	鶴岡	内海 哲史	阿部 勇次, 西川 顕正
36	カワイイはつくれる —モテる世界遺産を磨くための4つの心得—	松江	橋本 剛	土江 智明, 藤原 和太, 赤間 仁志
37	北のスタンプ2011	旭川	森川 一	高橋 勇紀, 田村 省太, 小野 敦夢
38	すたんぷ!	新居浜	占部 弘治	山内 康祐, 白木 俊成
39	Step By Step! —千回の修復作業も—スタンプから—	佐世保	嶋田 英樹	音丸 愛海, 前田 桃子, 谷川 友重
40	もうれつスタンプING (仮)	有明	石川 洋平	上原 耀, 松岡 禎明, 伊藤 綾奈
41	修復のスタンプ —未来のために—	香川(高松)	柿元 健	小笠原尚哉, 大前 彩, 河西 優樹
42	IRIS —Img Recover Imprint Sys—	福島	小泉 康一	松島 弘, 赤塚 篤, 赤津 亘
43	糸善 —”しぜん”に”つくろ”う—	都城	中村 博文	出水 慎也, 塚本 浩祐, 榎田 宗文
44	クリスマスの奇跡	熊本(八代)	小島 俊輔	窪田 一平, 植柳 実, 木村 匠
45	修復屋 —どんな傷でも直してみせる—	都立(品川)	福永 修一	池田 基樹, 手計 勝也, 阿久津卓土
46	Ikannol Mk-II —この前のうどん美味しかったね—	久留米	黒木 祥光	大城 泰平, 坂田 祐将, 徳永 誠
47	パズルそろったあああ〜! —i'm lovin' bit—	米子	倉田 久靖	澤下 陽, 木村 勇太, 松上 寿支
48	まったりスタンプ	群馬	大豆生田 利章	味曾野智礼, 味曾野雅史, 土屋 俊貴
49	あのか所の面影を求めて	福井	青山 義弘	和田はるか, 矢納 正浩
50	スタンプぺったん —キミも再生しよう!—	鹿児島	豊平 隆之	押川 直樹, 稲付 智昭, 阪元 亘
51	画像修正基地 —俺の画像が折れた—	大分	徳尾 健司	山崎 隆, 吉浪 遼, 河野 貴宏
52	GAIST	宇部	田辺 誠	石田 竹至, 大濱 勇平, 村長 将也
53	Mr. TangTangの反転大作戦 —ちゃが台は返された—	神戸市立	若林 茂	小田 悠介, 末廣 繁樹, 久保 圭太
54	スタンプボンボンポポポ〜ン押〜し〜	鈴鹿	田添 丈博	中山 大輔, 西脇慎太郎, 増井 隆治
55	世界遺産修復隊 —取り戻すのは僕らだ—	一関	管 隆寿	後藤 弘明, 五十嵐啓恵, 大門 雅尚
56	裏世界遺産	高知	谷澤 俊弘	高田 翔平, 敷地 辰也
57	Xオワコン —容疑者Yの探索—	富山(射水)	山口 晃史	毎原 雄介, 池原 光輝, 坂本 和哉
58	マルティネス —本当に修復って必要なのか?—	サレジオ	清水 哲也	大塚 達也, 川口 龍法, 小林 大悟
59	Mandarin Orange	東京	小嶋 徹也	荒川 智洋, 松田 和尙, 大塚 信吾
60	イマ・オシテマス	釧路	神谷 照基	大橋勘太郎, 荒木 建人, GOMBOLUDEV ENKHBAYAR

登録順	タイトル	高専名	指導教員	参加学生
VNU	TATO	ハノイ国家大学	PHAM Ngoc Hung	VU Thanh Tu, VU Viet Duc
NII	BEYOND DIFFICULTY	成都東軟学院	Zhou Peng	Zhong Zijian, Lu Yang
MUST	ONIS	モンゴル 科学技術大学	Batzolboo Bataa	Adiya Buyantogtokh, Batbaatar Burentogtokh
KMITL	Ladkrabang	キングモンクット 工科大学	Boontee Kruatrachu	Komkrit Anurit, Somthanat Wongsas

司会・解説者紹介

高田 信一

石川県在住。プロのナレーターとして活躍中。

第13回石川大会以降、現在に至るまで競技部門の司会・進行を担当。

伊原 充博

元・東京高専 電子情報工学科 教授。現在、特定非営利活動法人 高専プロコン交流育成協会 (NAPROCK) 理事 (事務局長)。

第7回北九州大会 (競技部門実施3回目) から競技部門の司会を担当、その明快な解説には定評がある。

競技部門のルール

はじめに1枚の画像がフィールドに示され、複数のスタンプが用意されます。スタンプをフィールドに適用するとスタンプと一致したフィールドが反転します。うまくスタンプを適用して、フィールドを最終画像にします。できるだけ早く初期画像を最終画像に「修復」したチームが勝利する競技です。

●用語など

セル

- ・画像の1ドットを「セル」と呼びます。
- ・セルは0または1の値をとります。

フィールド

- ・フィールドは最大で横640個、縦480個のセルから構成されます。
- ・フィールドの原点(0,0)は左上とし、右および下を正の方向とします。
- ・フィールドの大きさは問題ごとに定めます。

スタンプ

- ・スタンプは最大で横128個、縦128個のセルから構成されます。
- ・スタンプの大きさは原則異なりますが、同じものが含まれる場合もあります。
- ・スタンプの最大枚数は128枚です。
- ・スタンプの枚数は問題ごとに定めます。

修復

- ・フィールドの、ある矩形領域に対してスタンプを適用することを「修復」といい、1回の修復を「手数」としてカウントします。
- ・修復を行うとセルの値はフィールドとスタンプの対応したセルのxorの値となります。
- ・スタンプはフィールドの任意の位置に対して修復可能です。
- ・スタンプの一部がフィールドに重なっていれば修復可能です(はみだしてもよい)。

- ・修復場所はスタンプの左上セルを適用するフィールドの座標で表します。
- ・修復の例を図1に示します。これは、スタンプを(1, -1)に適用した例です。

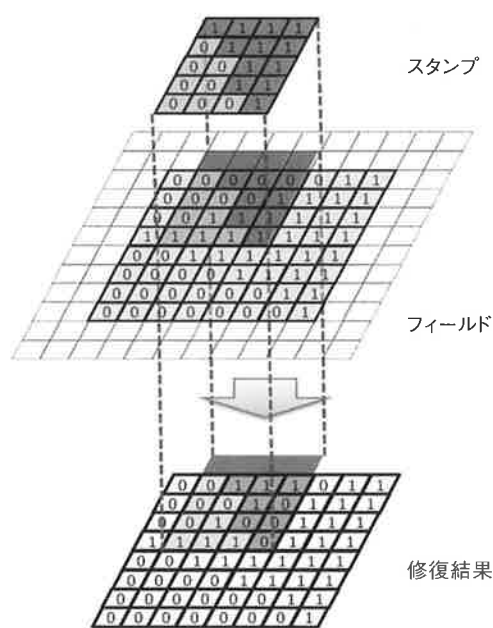


図1 修復例

問題

- ・1つの問題は指定画像2枚(初期画像と最終画像)と複数のスタンプで構成されます。
- ・初期画像はフィールドの初期状態を表します。
- ・すべての問題に、サイズが1x1でセルの値が1のスタンプが必ず含まれます。
- ・問題は1つのファイルとして提供されます(問題のフォーマットを参照)。
- ・問題のサンプルを次に示します。これは初期・最終画像の2つの画像と3つのスタンプで構成されている例です。

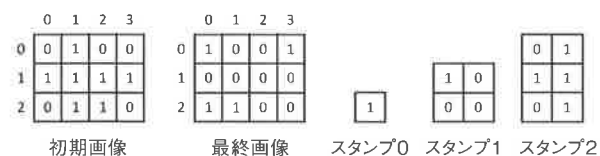


図2 問題例

●制限時間

- ・問題ごとに回答の制限時間を定めます。
- ・制限時間は1問あたり1～10分です。
- ・制限時間内に回答が送信し終わっている必要があります。回答の送信中に制限時間となった場合は、その回答は無効となります。
- ・制限時間は試合開始前に連絡します。

●試合の進行手順

- ・1試合は最大12チーム対戦で行います。
- ・1試合は2問で行います。
- ・試合開始とともに最初の問題がネットワーク経由で提供されます。
- ・各チームは制限時間内に問題を解き、回答をネットワーク経由で提出します。
- ・各チームから回答が提出されると、回答の状況(下記)についてスクリーンに表示します。
 - ☆回答が受理された
 - ☆回答画像(初期画像に回答手順を適用した画像)が最終画像と一致
 - ☆回答画像が最終画像と不一致
 - ☆回答がフォーマットエラー
- ・制限時間内であれば、10回以内の提出が可能です。
- ・最後に提出された回答が有効となります。
- ・1問目の制限時間経過直後に、1問目の結果(手数と時間と暫定順位)をスクリーンに表示します。また修復手順を表示することがあります。
- ・1問目の制限時間経過直後に2問目の問題がネットワーク経由で提供されます。
- ・2問目を1問目と同様の手順で行います。
- ・2問目の制限時間が経過したら、集計を行い、順位を決定します。

●順位決定方法

勝敗判定は以下の優先順位で決定します。

- 1.最終画像との一致度が高い方(修復画像と最終画像との相違セルの総数が少ない方)が勝ち。
- 2.修復手数が少ない方が勝ち。
- 3.最後の回答を早く提出した方が勝ち。
- 4.じゃんけんで勝った方が勝ち。

1試合の最終結果は上記の相違セルの総数・手数・時間を2問分合計して最終的な順位を決定します。

●その他のルール等

- ・競技に持ち込んで利用できるコンピュータ類は携帯可能なものを2台以内とします。そのうち1台は100BASE-TXが使用できるEthernetポートとUSBポートを持つ必要があります。コンピュータは用意されたテーブルに置くものとします。
- ・テーブルには各チームに合計150W程度の電源コンセント2口を用意する予定です。
- ・競技ネットワークに接続するためにLANケーブルを各チーム1本用意する予定です。2台のコンピュータを接続する必要がある場合は、スイッチングHUB等の機器を各チームで用意してください。
- ・競技中は、チーム内での情報のやり取りは構いませんが、チーム以外と情報交換することは認めません。
- ・コンピュータ間の無線による通信は認めません。
- ・サーバや他チームの試合進行を妨害する行為は認めません。
- ・試合の進行の妨害や審判または他チーム等への妨害、その他禁止行為があったと判断された場合等には、失格とすることもあります。失格とした場合は、試合の順位は最下位となります。
- ・システムに不具合が発生した場合はオフラインでの問題の配布と回答の提出になることがあります。この場合、試合時間や提出回数等が変更される可能性があります。また、別の問題を用意して再試合を実施する可能性があります。
- ・競技中、競技に参加している参加者および机の上を審査委員が審査のため閲覧したり、ビデオカメラ等で撮影・録画し、同時にスクリーン等に表示したりする事があります。
- ・競技で使用したデータおよび、各チームがサーバに送信したデータは、競技終了後Web等で公開する事があります。

競技部門の組合せ

1回戦 各試合上位3チームが準決勝へ進出する。4位以下は敗者復活戦へ。(海外チームはオープン参加)

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
1	富山(射水)	熊本(八代)	和歌山	新居浜	熊本(熊本)	沖縄
2	群馬	長野	津山	神戸市立	大阪府大	米子
3	鶴岡	苫小牧	仙台(名取)	北九州	八戸	弓削商船
4	鹿児島	阿南	大島商船	奈良	金沢	香川(高松)
5	旭川	明石	小山	都城	茨城	高知
6	東京	豊田	釧路	鳥羽商船	徳山	一関
7	鈴鹿	舞鶴	函館	木更津	有明	久留米
8	呉	佐世保	秋田	都立(荒川)	長岡	広島商船
9	都立(品川)	大分	岐阜	沼津	福井	石川
10	福島	近畿大学	香川(詫間)	サレジオ	宇部	松江
11			ハノイ国家大学	成都東軟学院	モンゴル科学技術大学	キングモックト工科大学
12						

敗者復活戦 各試合上位2チームが準決勝へ進出する。

ブース	第1試合	第2試合	第3試合	第4試合
1	1-1-4	1-2-4	1-3-4	1-4-4
2	1-5-4	1-6-4	1-1-5	1-2-5
3	1-3-5	1-4-5	1-5-5	1-6-5
4	1-1-6	1-2-6	1-3-6	1-4-6
5	1-5-6	1-6-6	1-1-7	1-2-7
6	1-3-7	1-4-7	1-5-7	1-6-7
7	1-1-8	1-2-8	1-3-8	1-4-8
8	1-5-8	1-6-8	1-1-9	1-2-9
9	1-3-9	1-4-9	1-5-9	1-6-9
10	1-1-10	1-2-10	1-3-10	1-4-10
11	1-5-10	1-6-10		
12				

準決勝 各試合上位4チームが決勝へ進出する。

ブース	第1試合	第2試合	第3試合
1	1-1-1	1-2-1	1-3-1
2	1-4-1	1-5-1	1-6-1
3	1-1-2	1-2-2	1-3-2
4	1-4-2	1-5-2	1-6-2
5	1-1-3	1-2-3	1-3-3
6	1-4-3	1-5-3	1-6-3
7	C-1-1	C-2-1	C-3-1
8	C-4-1	C-1-2	C-2-2
9	C-3-2	C-4-2	成都東軟学院
10	モンゴル科学技術大学	ハノイ国家大学	キングモックト工科大学
11			
12			

決勝戦

ブース	第1試合
1	S-1-1
2	S-2-1
3	S-3-1
4	S-1-2
5	S-2-2
6	S-3-2
7	S-1-3
8	S-2-3
9	S-3-3
10	S-1-4
11	S-2-4
12	S-3-4

※ X-Y-Z は X回戦-第Y試合-第Z位 を表す。

※ ただしX部分のCは敗者復活戦、Sは準決勝を表す。

※ 1回戦の第3試合目、第4試合目、第5試合目、第6試合目の11番ブースには、海外チームがオープン参加します。

オープン参加のチームの成績は、日本チームの順位には関係ありません。

※ 準決勝・決勝はNAPROCK国際プロコン(国際大会)を兼ねて実施されます。準決勝・決勝の海外チームは国際大会の公式エントリで、日本チーム・海外チームの区別なく、試合の成績により決勝進出や準決勝・決勝での国際大会の順位が決まります。

※ 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。

1

ヴァッサーマンV2

秋田

真壁 純矢(5年) 佐々木貴晃(5年)
渡辺 拓也(4年) 竹下 大樹(教員)

1. プログラムについて

修正箇所を明確に把握するために、初期画像と最終画像の各要素で排他的論理和を行ったものを要素とするフィールド XOR を使用して画像の修正を行う。

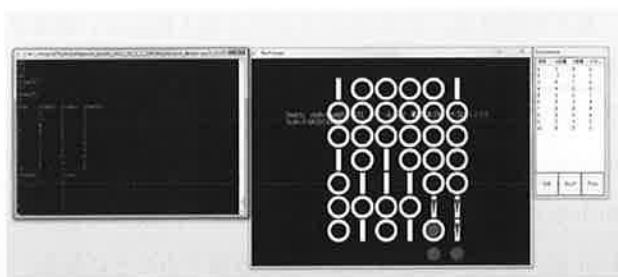
すべてのスタンプをフィールドの隅々まで当てはめるような総当たりの探索では時間が掛かり過ぎるため、フィールド XOR やスタンプの1となっている部分をリスト化して、それぞれのリストを探索することで処理速度の向上を図っている。また、探索に使用するスタンプは、最大の修正が望まれるスタンプから使用していき、それ以外の最大の修正に満たないスタンプは、探索の途中で探索を破棄する処理を行っている。

2. GUI について

開発支援システムを用いることにより、アルゴリズムの動作を分かりやすくするとともに、本大会の際には、送る

データの検討をしやすいことを重点に置いている。これらのシステムを用いることによって、より良いプログラムの実現を可能とさせる。

どのような機能を備えているかについては、棋譜の生成や使用したスタンプの枚数等の表示などであり、その他にも開発環境をより向上させることを目的とした機能を備えている。



3. 開発環境

Microsoft Visual Studio C++ 2008

2

Mスタンプ

沼津

坂代 一弥(専1年) 斎藤 秀人(5年)
金森 広樹(4年) 出川 智啓(教員)

1. 競技開始から終了まで

競技開始して、問題が提供されてからプログラムの処理が終了して結果を提出するまでの流れを示す。

- (1) サーバーから問題をテキストデータとして取得
- (2) 引数にテキストへのパスを入れてプログラムを実行
- (3) 実行結果をサーバーへ転送

ここで、処理結果は解答ファイルとしてテキストデータで出力し、サーバーとのやりとりを行うクライアントプログラムはサンプルで提供されたものを使用する予定である。

2. 競技進行中

まず、全てのスタンプを面積の大きい順にソートし、1番目のスタンプを、そのスタンプを適用可能な全ての座標にそれぞれ適用する。その結果、最も多くの画素を修正できた座標を採用し、原画像に反映させる。同じス

タンプで修正画素数が一定値未満になるまで繰り返し、一定値を下回った場合は次のスタンプに移り、同様の処理を行う。これを最後のスタンプ(サイズが1×1のスタンプ)になるまで行う。その後状況に応じてメタヒューリスティック・重み優先探索・局所探索などを用いてさらなるスタンプの適用を行う。この一連の処理を制限時間をなるべく限界まで使って行う。

3. UI について

今回の競技内容は処理速度が特に重要になってくると考えられるため、GUIによる進行状況の表示などは行わず、CUIを用いて各種パラメータを引数で指定して実行する形式を採用する予定である。

4. 開発環境

Microsoft Visual C++ 2008 Express Edition
NVIDIA Computing SDK (CUDA 4.0)

3

Watson On Stage

長 野

渡辺 大瑛(3年) 佐藤 俊之(3年)
大平 哲也(3年) 伊藤 祥一(教員)

1. 前提

本競技における修復作業では1×1のスタンプが用意されているため、どのような問題であっても確実に修復手順を導くことができる。従って、中途半端な修復では確実に敗北する。

2. アルゴリズム

我々の指針は、1×1のスタンプを極力使わない修復手順を、制限時間ぎりぎりまで粘って探し続けることである。アルゴリズムには、遺伝的アルゴリズムや幅優先探索の総当たりなどを用いている。しかし、いずれも実行に時間がかかるため、適当なところで打ち切り、残った領域は1×1のスタンプも動員して手早く修復することになる。

「最後の回答を早く提出した方が勝ち」という判定基準があるが、アルゴリズムの特性上、時間をかけるほど良い

結果が出るため、制限時間を大幅に残すようなことが無いようにしている。

3. 実行環境

1つのコンソールプログラムで、問題ファイルの読み出しから回答ファイルの書き出しまでを行う。制限時間を最大限に活用したいので、人による操作を極力減らし、人為的なミスやタイムロスが発生しにくいようにする。

ノートPCを2台使用可能なので、それぞれで打ち切り条件などを変えて実行し、成績の良い回答を提出する。

4. 開発環境

Microsoft Visual Studio 2010

C++

4

きゅーかんぱー —僕たちのきゅうりー

熊本(熊本)

小中 俊樹(4年) 木村 慎(4年)
VONGSOMXAI VILAYOUTH(4年)
孫 寧平(教員)

1. はじめに

本プログラムはJAVA言語を用いて実装した。計算速度を重視するために、オブジェクト指向ハッシュ法とビット演算を導入した。さらに、実行時操作できるインターフェイスを作成した。

2. 基本構造とプログラムの概要

初期画像をフィールド1、完成画像をフィールド2とする。(1)式より、必要なスタンプパターンを導出し、そのパターンをフィールド3とする。フィールドとスタンプのデータ構造をBitSetクラス(並べているビット)で管理する。

フィールド1 XOR フィールド2 = フィールド3 (1)
スタンプの最初に1が入っている行の事を Focus と、今探している行の事を Now と呼ぶ。フィールド3のパターンを探るとき、上から順番に探していく。この各スタンプの Focus 行をハッシュングの key として HashMap オブジェクトに保存し、ハッシュの Value(結果)をそのスタンプにする。検索する際に、まずフィー

ルド3の上からみて、Focus とそっくりのパターンがあるかどうかを探していく。もしなければ、スタンプ0番にして、スタンプ0番のFocusとフィールド3のNowとスタンプする。これを繰り返すと、正しい答えを得られる。

3. オブジェクト指向ハッシュ法とインターフェイス

探索に使われたハッシュオブジェクトの中には、KeyとValueがある。Keyは検索するためのキーワードで、Valueは検索したあとの結果である。このアルゴリズムの特徴はKeyとValueのデータ形式を依存しないことである。1回ずつプログラミング検索必要な回数は1回しか行われぬ。よって計算速度を向上させることができる。

GUIを用いたインターフェイスを作成し実行時に利用する。画像データ・スタンプデータを読み込む際に、ファイル名の入力ミス等による時間の遅延を防ぐために、直観的にドラッグとドロップでデータファイルの入力を行う。

5

ぶらり遺跡探検の旅

木更津

平井 成海(5年) 金巻 俊一(5年)
東 遼平(5年) 白木 厚司(教員)

1. 概要

並列計算により、将来的に最適と考えられる解の探索を行う。

2. 戦略

多段階の先読みを行い、中長期にわたって多く修正がされる手を採用する。

3. ユーザーインターフェース

コマンドライン版のフロントエンドとして動作する。状況を視覚的に確認することを可能にし、操作を行う。

開発環境

Microsoft Visual Studio 2010
NVIDIA CUDA SDK
Boost C++ Libraries

```

デコーダー4終了(.%decode4.cpp: 85)
デコーダー1スタート(.%decode1.cpp: 11)
デコーダー1終了(.%decode1.cpp: 50)
デコード終了(.%ProgramingContest.cpp:216)
現在の時間: 0(.%ProgramingContest.cpp:217)
デコード後のステージ表示(.%ProgramingContest.cpp:252)
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
0 #####
1 #####
2 #####
3 #####
4 #####
5 #####
6 #####
7 #####
8 #####
9 #####
10 #####
11 #####
12 #####
13 #####
14 #####
15 #####
16 #####
17 #####
18 #####
19 #####
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

```

図：実行の様子

競技部門

6

Building Repairer

香川(詔間)

渡邊 裕輝(5年)
ANANDA ABDULLAH(5年)
日野晋太郎(3年) 宮武 明義(教員)

1. システム

問題の難易度、大きさ、初期画像と最終画像との相違セルの割合などに応じて、それぞれに合ったアルゴリズムを複数用意しておき、それらのアルゴリズムの中から最適なものを選んで実行する。

2. 探索アルゴリズム

基本的には幅優先探索を行う。評価関数によって修復画像を評価し、評価値が閾値より高いもののみを残し、最終画像に近づけていく。評価値の閾値は探索の進行状況や残り時間にに応じて変更する。

3. GUI

初期画像、最終画像、スタンプ、現在探索中の画像などをGUIに表示させる(図1)。これを見て、パラメータの調整や不要と思われるスタンプのカットを手動で行う。



図1 開発中のGUI

4. 開発環境

Microsoft Visual Studio 2010
思考ルーチン部: C++ GUI部: C#

7

Φερενικη (ふあらにきん)
——勝利を導く者——

津 山

小山 竜矢(3年) 神崎 拓人(3年)
石谷 沙央(3年) 岡田 正(教員)

1. アルゴリズム概要

この問題は完全情報ゲームなので、ゲーム木を作り予め先読みしておいて、合計のコストが最小になる手を選択する。

今回は計算量が多く、深さが定まってないので、最小コストの探索には IDDFS (反復深化深さ優先探索) を使用する。

また、予め定め石データベースを作っておき最適解を蓄えておくことにより解探索の効率化を図る。

2. ゲーム中の運用方法

予め重み付けパラメータの違う AI を複数用意しておき、最悪でも単純解は求められるようにしておく。

また思わぬ自体にも対応するため解答ファイルをチェックするプログラムの実装や、電源の接続状態を競技開始前に確認しておく。

3. 開発環境

Borland C++ 5.1.1 for Win32

Microsoft Visual Studio 2010

DX ライブラリ 3.06c (ビジュアライザ)

Active Perl 5.10.1 (問題ダウンロード部)



ビジュアライザ(※画像はイメージです)

8

やぜまぶもらと
——「ぷちりょんこ」ってなんだ——

呉

平井 勇大(3年) 山本 和空(3年)
山根裕一郎(3年) 藤井 敏則(教員)

1. システム概要

1.1 XOR スタンプデータ計算

- (1) はじめは、 1×1 の1つめスタンプで確実に解答する。
- (2) 次に、2つ目の小さなスタンプの最も一致したところから順に組み合わせて行き、1つより一致度が小さくなったら、 1×1 の1つめスタンプで、解答する。
- (3) 3つめのスタンプでも同様に行い。128個のスタンプまで、大きいスタンプから順に一致度をチェックして解答していく。
- (4) (3)の作業が終了したら、今度は全スタンプデータを最大16分割し、各分割箇所、最も適するスタンプを大きな順に決め、最適解に近づけていく。このとき、はみ出して、変化したデータに応じて、次の分割した箇所の最適化を行う。
- (5)(4)での操作後、分割数を減らし、(4)の操作を行う。

解答時間が迫ってきたら、最も手数の少ない解答(9回目)を行い。最終的な最適化結果(10回目)を時間ぎりぎりまで計算する。

2. 開発中のスクリーンショット

計算量が多いので、openMP などの並列計算を使用することも考えています。



1. システム概要

本システムは以下のアルゴリズムを使用して、問題の最適解を求める。作戦として、PC2台を使用しそれぞれ別の手段で解をもとめる。また A*アルゴリズムを使用することで、効率よく画像の修復をする。

2. 修復について

2.1 修復の基本方針

① 修正範囲の特定

まず、修正すべき部分を確認するために初期画像と最終画像の異なる部分を AND 処理によって比較する。そして最も変化の多い領域を修正範囲の候補とする。

② スタンプの適用

修正する場所を特定したらスタンプを適用し、画像を修正していく。このとき最も効果的なスタンプを優先して適用することとする。

③ ①、②の処理を繰り返す

スタンプの適用において、1つの処理を行うと、局所的な解に陥る場合があるため、先読みの範囲を限定した A*アルゴリズムを用いた効率的な探索を導入する。

2.2 実践における作戦

実践においてはアルゴリズムの性質を変えた PC2 台で同時に問題を解く。1台は速度を重視し、もう1台ではコスト重視で問題を解く。最終的に最小手数で画像を修復できた方の解をその問題の解として提出する。

3. 開発環境

Microsoft Visual Studio 2010

1. アルゴリズム

本課題の解決アルゴリズムは下記の通りである。

画面に対して大きいスタンプをランダムな位置に押しつけた場合の画像を仮画像とする。

仮画像と最終画像で比較を行い、画像間の距離が近づけば実際にスタンプを押すことにする。

次に中程度のサイズのスタンプを用いて、大きいスタンプと同様に、最終画像との距離を判定しながら押していく。

中程度のスタンプを押しても画像間の距離が縮まらなくなったら次の工程に移る。

最後に最終画像との差画像を用意し、あらかじめ用意しておいた小さいスタンプと一致する場所があればスタンプを押し、最終画像と一致すればスタンプを押す作業を終了する。

上記の判定を繰り返し、最小の手数で最終画像になった

場合について終了処理を行う。

2. 開発環境

Visual C++ 2010 Express Edition

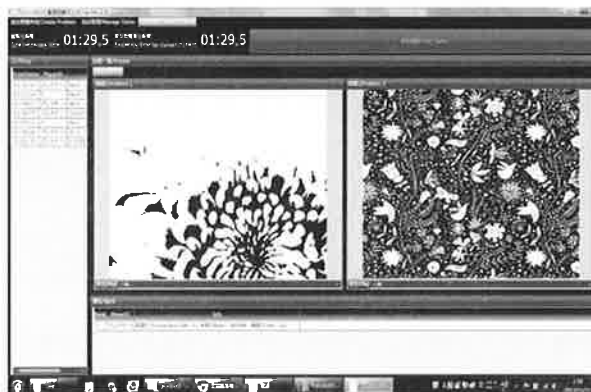


図 開発中の画像

11

Wish the vanishing
—消したい過ち—

近畿大学

林 健太(5年) 中野 弘基(5年)
尾上 貴子(5年) 政清 史晃(教員)

1. プログラムの概要

ネットワーク経由で画像データを取得する。目標画像と現在の画像との差分を調べ、修正箇所を表示する。ユーザーインターフェースにより、修正箇所が集中している範囲をいくつか指定し、その範囲内を修正してゆく。その範囲の修正が終わると次の修正範囲を修正する。差分が無くなるとプログラムを終了し、手順をファイルに出力する。

2. アルゴリズム

修正範囲内の修正方法は、修正範囲の中心座標を求め、中心から外側へ広がっていくように修正してゆく。また、修正箇所をより正しく修正できるスタンプを選び、手順を少なくする。

3. ユーザーインターフェース

修正が必要な箇所を表示し、それを人間が見て判断する。修正箇所が集中している範囲をマウス等で選択し、プログラムに座標を渡す。

4. 開発環境

Microsoft Visual C++

12

ホテルナカヤマ
—304号室の中のリアル—

金 沢

宅美 貴裕(4年) 真田 純(4年)
松永 隼季(4年) 元木 光雄(教員)

1. 問題をどのようにして解くか

幅優先探索を用いて解く。当然、スタンプ数や画像のサイズが大きくなれば、計算時間が大幅に増える。計算時間を短縮するために、修復対象の画像と完成画像の距離を必要に応じて計算し、ある位置にスタンプを適用すると距離が遠くなる場合は、その手を捨てるようにしている。

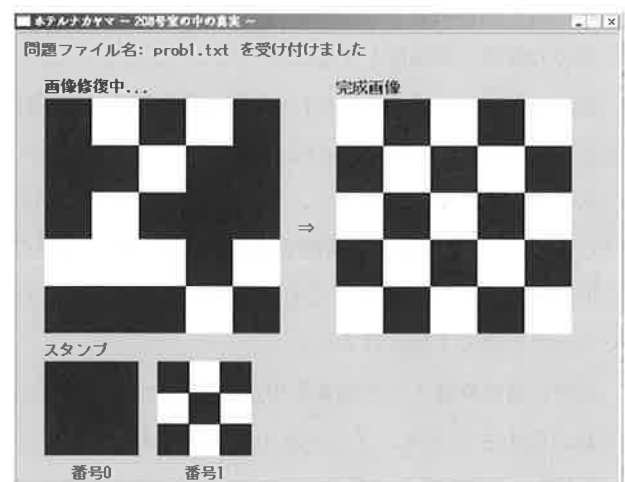
距離の計算は、修復対象画像と完成画像のそれぞれのセルについて、同じかそうでないかを比較し、違うセルの数が少ない場合、距離が近いとするようにしている。

2. 入出力について

問題ファイルをウィンドウにドラッグ&ドロップすることで読み込む。出力は「問題ファイル名+_ans.txt」というファイル名で問題ファイルがあるディレクトリに出力する。

3. 開発環境

Microsoft Visual C++ 2008



開発中の画面です

1. 競技進行中

プログラムを起動し、修復方法を選択、修復が終わり、回答ファイルが作成されたら回答提出クライアントを起動し、回答を提出する。残り時間に余裕があれば他の修復方法を試し、修復手数が少なくなれば提出する。

2. 修復方法の選択について

今回の競技で我々は問題の内容によって有効な修復方法が変わってくると考えた。また、時間がかかる修復方法を用いた場合、設定された制限時間によっては修復が間に合わない可能性も考えられるため問題によって修復方法を変更できるようにした。

3. 相違フィールドについて

我々のプログラムでは初期画像と最終画像のビットデータを比較し、異なっている座標のビットデータを1とした相違フィールドを作成します。これによって修復時のス

タンプを押すか否かの判断及び修復が成功したかどうかの確認が容易になります。

4. 修復方法

4.1 パターン1

スタンプを相違フィールドの左端から順に比較していき、スタンプの1と相違フィールドの1が重なる割合が一定以上ならばスタンプを押す方法。修復手数が少なくなるが時間がかかる。

4.2 パターン2

相違フィールドの左端から修復する基準点を決め、その基準点で最も修復が進むスタンプを押す方法。時間があまりかからない。

5. 開発環境

Microsoft Visual C++ 2008 Express Edition

1. はじめに

今回の競技内容は、問題の回答を導くアルゴリズムによってのみ結果が決定する。他チームの影響などを考える必要がないため、独立したプログラムにより計算を行う。

2. 前処理

前処理として、初期状態と完成状態の排他的論理和をとる。これによって反転しなくてはいけない部分だけのフィールドを得ることができる。また、このフィールドにおいて反転させるべき場所を反転した数を評価値とする。

3. 処理の流れ

- [1]反転させる面積が大きいスタンプから処理を行う。
- [2]ある座標に仮想的にスタンプを適用した場合に、得られる評価値を取得していく。
- [3]評価値が高かったスタンプおよび、適用した座標のものをいくつか候補とする。

[4]候補となったスタンプと座標セットにおいて、次の手を計算する。

[5]数手先まで計算したものの中で一番評価値が高かったものを確定とし、それを適用した状態を初期状態とみなし、再び上記の計算を行う。

4. 枝刈り

本プログラムでは、以下の点において枝刈りを行うことで高速化を図る。

- ・[2]でのスタンプ適用時に、そのスタンプが持つ最大評価値よりも、すでに算出された評価値が上回る場合、そのスタンプの各座標における評価計算を行わない。
- ・一定数先の評価計算ごとに、確定とすることで競技時間に間に合うように計算量を減らす。

5. 開発環境

Visual Studio 2010 (C#)

1. 今回の競技について

今大会は、先の東日本大震災の影響により 12 月開催となり、例年よりもかなり遅めの開催です。開催日程が遅くなったこともあり、今年はよりプログラムの作りこみが求められることになりました。特に本競技は、多人数対戦のように不確定要素が少ない内容で、チーム毎のアルゴリズムの性能差が顕著に出ることになります。そのため、どのような手段が有効か考えていく必要がありました。以下私たちの約 8 ヶ月間の取り組みについて簡単な説明させていただきます。

2. 探索アルゴリズム

競技に臨むにあたって、問題の難易度によって幾つかのアルゴリズムを切り替える形にしました。

使用アルゴリズムとしては幅限定探索と広域探索に有効である GA, SA をベースとしています。

3. 探索範囲の縮小

最大で画像は 640×480、スタンプは 128×128、と探索範囲は非常に広いものになります。そのため、そのまま探索したのでは、計算量が膨大になるため制限時間内に解を求めるのは難しいと判断しました。そこで、私たちは与えられた画像を幾つかに分割するなどをして探索範囲の縮小を図ります。

4. UI について

処理速度と開発コストを考え、プログラムの入出力はコンソールアプリを用いています。

5. 開発環境

Microsoft VisualStudio2010 Professional C/C++

16 佃煮

1. システムの概要

ネットワーク経由で配信された問題について、独自に開発したアルゴリズムを用いて最適解の導出を目指す。今回、解の探索回数をできるだけ抑え、制限時間内に最適解に近い解を導出するアルゴリズムを構築した。

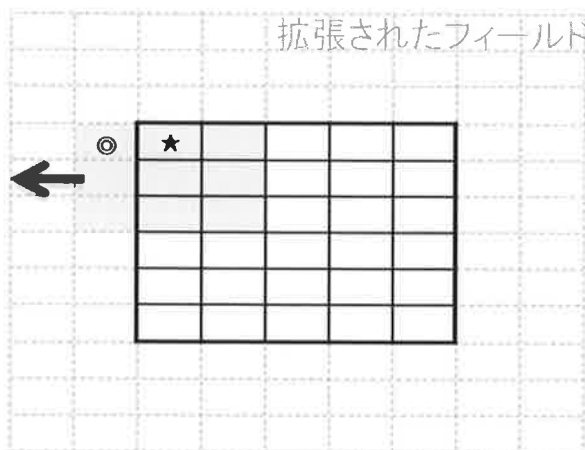
2. アルゴリズム

フィールド上のあるセル(例えば、図のマーク★)にスタンプの左上(図のマーク◎)を合わせスタンプした場合の効果、スタンプの種類および位置を候補に追加する。スタンプの領域が★に重なる範囲で左および上にずらし候補を追加する。ここまでの操作を他のスタンプについても同様に行い、得られた候補全ての中から最も効果的な位置にスタンプを押す。次にフィールド内で★をずらし、以上の操作を繰り返す。初期画像が最終画像に変換された後、得られた解答を送信する。

3. 開発環境

OS : Windows7

言語 : Java



17

そんなスタンプで大丈夫か？ — 一番いいスタンプを頼む —

都立(荒川)

宮崎 亮輔(4年) 平間 章裕(4年)
三田 和広(4年) 鈴木 弘(教員)

1. 競技開始まで

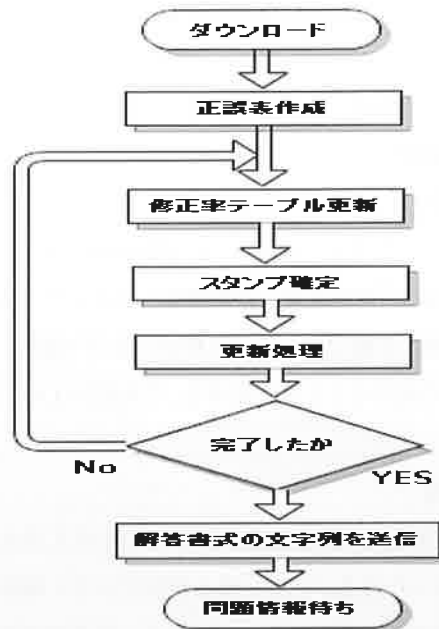
競技開始1分前に、ソフトウェアを起動して以下の処理を行う。

- ・サーバー側にプレイヤー情報を送信し、プレイヤーとして参加できるようにする。
- ・競技開始までは問題情報の取得待ち状態としておき、常に問題情報を取得するためにサーバー側に要求をする。
- ・競技開始直後に問題情報をダウンロードする。

2. アルゴリズム

元の画像と現状の画像の EXOR を取った正誤表を作成し、それに基づいて各セル(フィールド外も含む)での修正率が一番良い(元の画像に一番近づく)スタンプを修正率テーブルに格納する。修正率テーブルの中から最も修正率が良いスタンプを押していくことで元の画像への修正を行う。

3. フローチャート



18

ヨミ — 愛と勇気の修復マスター —

徳山

中島 淳平(4年)
MUNKHJARGAL TUGS-ITGELT(5年)
CHINDEGSUREN CHINBAT(4年)
三浦靖一郎(教員)

1. システム概要

2台のノートパソコンを用い、それぞれ別のプログラムを使って同じ問題を解き、修復手数が少ない方を回答として提出する。

2. 問題解決の手法

最良優先探索に基づいた方法で解を探索する。

始めに初期画像をフィールドの初期状態として探索候補に追加し、次の手順を探索候補がなくなるか時間制限に達するまで繰り返す。

探索候補から評価値が最も高いものを取り出し、スタンプを適用する。その結果が最終画像と一致しているならば、今までの修復手順を解として保存する。そうでなければ、スタンプを適用した結果を新しいフィールドの状態として探索候補に追加する。ただし、すべての状態を保存することはコンピュータのメモリの関係から不可能なため、評

価値の低い探索候補は適宜除外する。

評価値を計算する方法は複数準備し、問題の特徴や現在のフィールドに応じて用いる計算方法を変更する。

3. 開発環境

Microsoft Visual Studio 2010 (C#)



図. システム画面

1. システム概要

本システムは、通信部、思考部、表示部に分かれている。これらは図 1 に示すようなユーザーインターフェイスから操作可能である。

2. 各部の処理

2.1 表示部

サーバーからダウンロードした問題の表示を行う。表示される内容は、初期画像、最終画像、スタンプである。また初期画像に手動でスタンプを押すことが可能で、それを回答として出力することも出来る。これは図 1 の①に相当する。

2.2 思考部

パズルの解法は探索を基本としたアルゴリズムを利用した。作成した思考パターンは 1 種類ではなく数種類用意し、その中から人間が選択し実行する。思考部の起動は図

1 の②から起動する。

2.3 通信部

サーバーに接続し、問題のダウンロード、回答のアップロードを行う。回答データは、思考部が回答を終了した順に自動選択する。図 1 の③の部分に相当する。

3. 開発環境

Microsoft Visual C# 2010

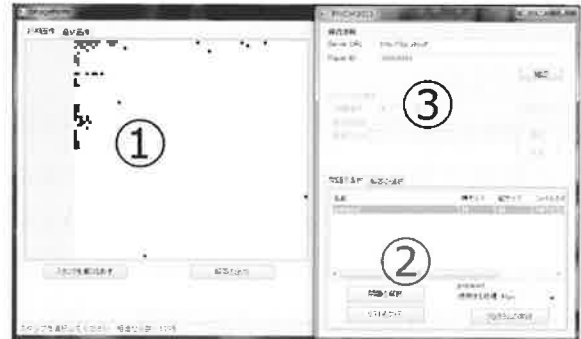


図 1 ユーザーインターフェイス例

1. 作戦

本システムは、時間内で確実に提出できることを重視し設計している。そのため最初に簡単なアルゴリズムによるプログラムを実行してその結果を提出し、その後、手数を減らせる複雑なアルゴリズムを適用し、実行結果を提出する。基本的な方針は以下である。

- ・大きいスタンプをなるべく使う
- ・ただし正解のセルが多いところでは小さいスタンプを用いて小回りを利かせる
- ・フィールドを分割したら早く正解にたどり着けたため 2×2 や 3×3 に分割してスタンプを適用していく
- ・スタンプの数が多いときには、似通ったスタンプを探し一方のみを使用するようにする

2. プログラム

2.1 基本形

スタンプ左上のブロック 1 マスのみを使用したプログ

ラムである。左上が 1 の場合に反転したいときにスタンプを適用させる。

2.2 スタンプ選考形

左上から右側もしくは下側に連続する 2 マスが (01、10、11) の組み合わせで並んでいるスタンプを選び出し、スタンプ 1~3 として登録する。これを問題の左上から走査していき、登録したスタンプの中から適切なものを選んで適用していく。

2.3 遺伝的アルゴリズム

遺伝的アルゴリズムによる解答を試みる。上記の基本形やスタンプ選考形の結果を種としより良い結果を得る。時間がかかるアルゴリズムのため、並列化による高速化を図り制限時間内での提出を達成する。

3. 開発環境

Microsoft Visual Studio 2008 C++

・システム

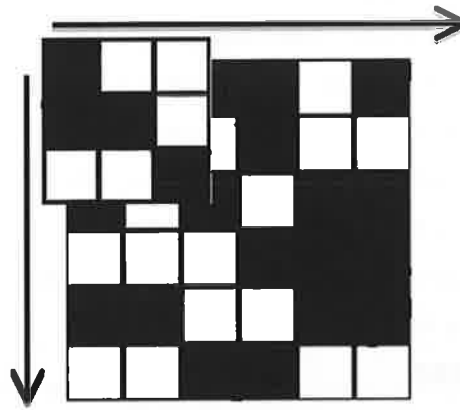
初期画像と最終画像を比較し、誤差部分を1、それ以外を0とした誤差画像を作成。誤差画像を全て0にするようにスタンプを置いていく。

各セルにスタンプを適用した時の改善セル数を計算し、最適改善セルにスタンプを適用する。

・戦略

- ・ 使用するスタンプを5～10に絞り、処理回数を減らす。
- ・ 大きなスタンプに対しては1の割合を目安となる閾値の設定し、スタンプを選別し処理から外す。
- ・ スタンプの1の割合と適応させる画像の1の割合の関係から改善セル数の目安となる閾値の設定し、スタンプを選別し処理から外す。
- ・ スタンプが枠からはみ出るセルは除外する。

などの処理を組み込み、処理効率を上げる。



左上から完成後に一番近づく
場所を探す

・開発環境

OS windows7

言語 Microsoft Visual Studio C++

1. 概要

制限時間 10 分で最適なスタンプの修復位置や手数が分かる「第 22 回プログラミングコンテスト」

参加チームは 60 チームを突破!

結果もすぐ分ると大人気だ。

2. アルゴリズム

画像の修復に用いるスタンプと修復セルの位置、修復される画像のセルの状態等を評価関数を用いて評価を計算し、その値に応じて適切な修復を行っていく。

さらに、比較回数が膨大になるため適切なアルゴリズムやデータ構造を用いて処理を高速化を行う。

```

bash
[*] Hello!
[*] confirmed data!
    image: 144x284
    stamp: 7 count
    1x1 , 2x2 , 3x3 , 4x4
[*] start algorithm 3 seconds ago...
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value
[*]intermediate image has cell value

```

1. プログラムのコンセプト

この競技では、より少ない手数で画像を修復したチームが勝ちなのだが、かなり単純、もしくは画像サイズの小さな問題でない限りはどう頑張ってもそれ以上短くすることができないような最少手数の回答は出ないだろうと考えた。そこで、自分達のプログラム製作の方針として、「制限時間をできる限りいっぱいまで使って最善手を導き出す」というものを考えた。

2. 実戦における動作

2.1 前準備

修復前の画像を修復後のものに修復することが目的なので、それらの差異が重要になる。問題をダウンロードした時点で、差異の認識を行い、その後問題に含まれているスタンプの確認、実際に修復に用いるかの取捨選択、優先度の設定を行う。

2.2 手順模索

現在の状態で実行できる一手の中で、先程確認した2枚の画像の差異をより多く埋められるものを優先して実行していく。この時、大きなスタンプはある程度修復が進むと修復に利用できなくなると考えられるので、適宜修復から除外していく。

2.3 制限時間が無くなってきた場合について

ぎりぎりまで時間を使うことをコンセプトとしているので、随時設定される制限時間に柔軟に対応する必要がある。そこで、制限時間が無くなってきた場合にある程度の手数増加の代わりに修復速度が向上するアルゴリズムを別に用意する、途中提出を可能にするなどの対応をしている。

3. 開発環境について

Microsoft Visual C# 2010 Express Edition

1. 概略

初期画像と最終画像の XOR をとった画像を目的画像とする。目的画像のすべてのマスが 0 になるようなスタンプの組合せのうち、最も手数の少ないものを求める。

2. データ構造

解を、「x, y, スタンプの番号」の三つの情報をひとまとめにしたものの配列で表現する。ここで x は、スタンプの左上の x 座標、y は左上の y 座標を表わす。

3. 解の評価

目的画像に解のすべてのスタンプを押したとき、残っている 1 のマス数とスタンプを押した回数の和を解の評価値とする。

4. アルゴリズム

ランダム性を持たせた貪欲法を用いて初期解を生成し、これを改善する。解の改善には局所探索法を始めとするメタヒューリスティックアルゴリズムを用いる。

初期解の生成、改善を複数回繰り返すことにより、より良い解を求める。

5. 開発環境

Microsoft Visual C++

Microsoft Windows7 Professional

1. はじめに

今年の競技部門は昨年度の課題と大きく異なり、運による揺らぎが無く、プログラムの出来がそのまま勝敗を左右する問題でした。また、愚直に計算すると全く解が得られないほど十分に奥が深い問題で、良い解が得られるようになるまで非常に苦労しました。

2. プログラムについて

2.1 競技開始

初期画像と最終画像のすべてのドットに対し xor 演算を行い、差分画像を生成する。今回の問題では差分画像をすべて0にするスタンプの適用を求めればよい。

2.2 山のぼり法

評価関数を用いて盤面を評価し、最も良い評価となるスタンプの押し方を調べ適用することを繰り返す。

2.3 解の改善

一度求めた解から新たに差分画像を生成し、小さい問題を生成することにより解を改善していく。

2.4 Viewer

求めた解を視覚化するプログラムを作成し、評価関数の調整を行った。

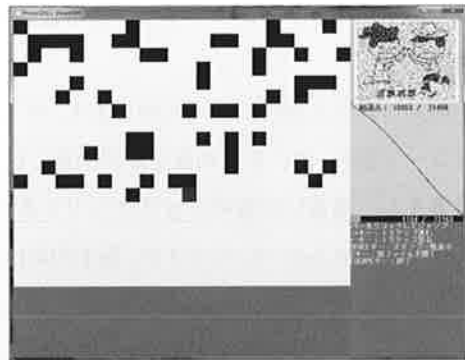


図 1 Viewer

1. アルゴリズムについて

問題を受信したら、まず初期画像と最終画像の XOR を取った画像を求める。この画像を処理用画像と呼ぶこととする。この画像にある"1"をすべて"0"にすることは、初期画像を最終画像に変換することと等価なので、問題の単純化のために行なう。

その上で、処理用画像を適切な大きさに分割し、分割した領域それぞれについて大きい順に 2、3 個のスタンプを適用し、領域内の"1"を最も少なくするスタンプおよび押下位置を探索し、適用する。これを繰り返し、最終的には 1×1 のスタンプですべて"0"にするようにする。

2. 全探索アルゴリズムとの併用

PC を 2 台用意し、1 台では上記アルゴリズムによる処理で解を求めるが、もう 1 台ではすべてのスタンプをすべての位置で適用して、処理用画像内の"1"の数を最小にす

るスタンプおよび押下位置を求める、いわば全探索での処理を行なう。

もちろん、問題が複雑であればあるほど、全探索で求めることは非効率となるが、初期段階でのスタンプ押下位置を全探索で求めることができれば、考案したアルゴリズムと組み合わせて、より効率的に解を求めることができると考えた。

3. 開発環境

Microsoft VisualStudio 2010 Express Edition

1. プログラムについて

1.1 競技開始前

プログラムを起動し、URL を指定します。一度入力するとプログラム終了時に保存され次回起動時に読み込まれます。

必要であれば、接続テストをします。接続テストでは指定した URL のデータをダウンロードして表示します。

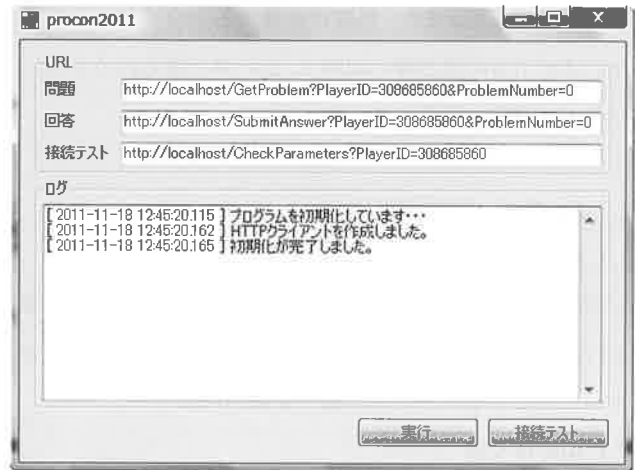
1.2 競技開始後

競技が開始されたら実行ボタンを押します。すると問題のダウンロードされ、次に初期画像を最終画像に復元するための計算をし、最後に回答のアップロードします。

復元のアルゴリズムについてですが、基本的には複数の座標にスタンプを押してそれぞれの有効度を評価し、最も有効なものを適応していきます。ただし、画像が小さい場合総当たり検索など、状況によって変わります。

2. UI について

今回の競技内容は計算コストがかなりかかるものなので、UI もできるだけシンプルにすることにしました。



3. 開発環境

Microsoft Visual Studio 2010

1. システム概要

プログラムは以下の流れで処理を行う

1. 問題の取得
2. 最適な手順の探索
3. 解答の提出

2. 最適な手順の探索

2.1 フィールドと最終画像との一致度の判定

フィールドと最終画像との一致度を計算するために、初期画像と最終画像を比較し、相違セルを1、一致セルを0として、作業を行うフィールド(作業フィールド)を作成する。この作業フィールドの値が1のセルを数えることにより、初期画像と最終画像との一致度を計算する。作業フィールドのセルの値がすべて0になったら、最終画像と一致したとみなし、探索を終了する。

2.2 グラフの登録

まず、作業フィールドにスタンプの仮適用を行う。次に、仮適用後の作業フィールドを評価関数により、評価を行う。評価値の高かった作業フィールドの評価値、セルの情報、適用したスタンプ、適用位置をグラフに登録する。

2.3 探索

グラフの探索を行い、最良な手順を導き出す。幅優先探索を用いれば、最適解が求まると考えられるが、単純な幅優先探索では探索量が膨大になり、時間内に問題が解けない。そのため探索は、制限付きの深さ優先探索や最良優先探索などを用いて行う。

3. 開発環境

VisualStudio 2010(C++)

1. はじめに

今回の競技部門は最適解を求めるためには問題によって異なるアルゴリズムによって解を求める必要があると考え、複数の解法アルゴリズムを選択可能にした。

2. アルゴリズム

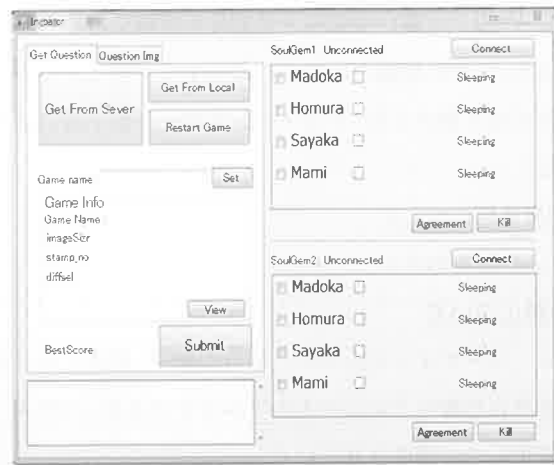
問題の画像の大きさやスタンプの種類によって次のいずれかのアルゴリズムを適用する。

- ①【Zero アタック】スタンプ 0 を差異セルに適用する。
- ②【5Split アタック】5*5のスタンプを主に適用し、分割統治により計算量を削減する
- ③【元気玉アタック】mix-max 法を用いた深さ優先探索により画像全体の最適解を求める
- ④【BS アタック】大きいスタンプを優先的に適用する

3. 実装

2 台の PC を用いて、複数アルゴリズムの分散処理を可能と

するシステムを実装した。アルゴリズムは自動的に選択されるが、プレイヤーで変更できるよう GUI を実装した。



4. 開発環境

Microsoft Visual C++

1. システム概要

計算部と表示部を分けてプログラムを作成する。

2. 計算部のアルゴリズム

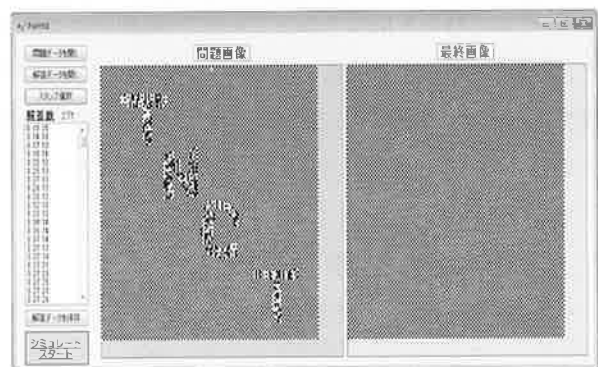
競技開始と同時にプログラムを実行する。その際に、画像の大きさによって大きく二つの戦略を取り入れる。

- ・ 小さな画像の場合
経路探索アルゴリズムを応用し、最適解を求める。
- ・ 大きな画像の場合
時間効率の良い近似アルゴリズムによって、近似解を求める。

3. 表示部 (GUI)

表示部は画像の読み込み、解答中の動作や詳細なデバッグデータの出力に対応している。ただし、計算部単体でも CUI ベースで動作するように設計し、表示との結合度を

下げる。



4 開発環境

Microsoft Visual Studio 2010

Microsoft Visual Studio 2008

NetBeans

1. システム概要

アルゴリズムモジュールの独立性を高めたシステム設計を行っており、高い生産性と並列性を実現することが可能である。また問題の取得から提出までを全て1つのシステムで行う事ができ、ヒューマンエラーを最小限に抑えている。

2. 開発言語について

Visual C#を利用している。

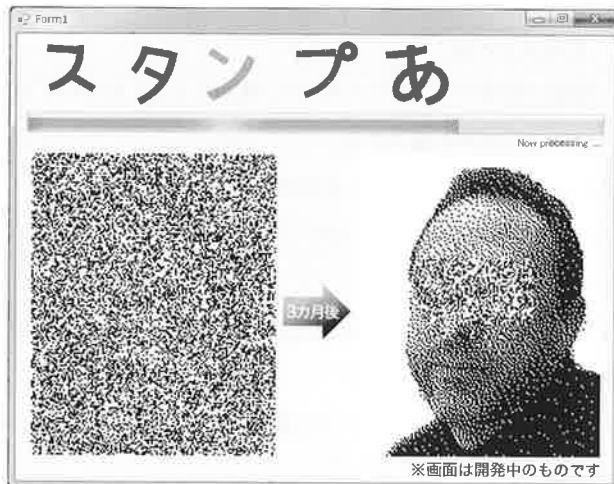
3. 戦略について

ヒューリスティックを用いた独自の物や、遺伝アルゴリズムを用いた物などいくつものアルゴリズムを、パラメータを変更しつつ適応させていく。

また1x1のスタンプが存在するため、最低点が自明である。その最低点より最適な演算の終了タイミングを割り出し、それよりパラメータを調整する。

4. UIについて

本システムの操作は全てGUIから行う。主に2種類のGUIがある。アルゴリズムの検証を行うために最適化されたGUIと、本戦にて用いる事を想定した対戦用GUIである。



5. 開発環境

Microsoft Visual Studio 2010 Professional

1. 概要

このシステムは大きく分けて問題の受け取りから修復処理、そして解答を出力するまでを行う『処理プログラム』と、出題者が提供する問題ファイルや処理プログラムが修復処理中に途中経過を出力したファイルなどを読み取り、人が見てわかりやすいように情報を可視化させる『ビューアー』の2つからなります。

2. 処理プログラムの機能

処理プログラムは、自作のアルゴリズムを用いて問題処理を行い、最終的に解答出力を行うものです。このアルゴリズムは初期画像と最終画像のXORをとった『相違図』中の"1"のセル、つまり『相違セル』に注目して修復作業を行います。また、手数を少なくしてセルの一致度を高めるために相違セルの群集を捉えたり、大きなサイズのスタンプを優先して使用する仕様としています。

この他特徴として、相違セルが散在しアルゴリズムによる処理が効果的でなくなった場合や修復処理中に解答提出の制限時間が迫った場合に、相違セルは必ず0にするという当初からの方針より、残りの相違セルをサイズ1x1のスタンプで修正していき、解答を半ば強制的に出力する機能があります。

3. ユーザーインターフェース

処理プログラムは、コマンドプロンプト上で動作するCUIアプリケーションです。処理プログラムの動作を決める程度であればGUIを実装する必要は無いと考えこの仕様としました。

ビューアーは、ウィンドウ内での図の表示やクリック操作によるファイル読取が可能なGUIアプリケーションです。ただし、ビューアーの機能が簡素なものであることから、画面構成もシンプルにしています。

1. 全体の流れ

競技用サーバから問題データをテキスト形式で受け取り、その情報を2台のコンピュータでWindowsファイル共有を用いて共有する。そして、2台のコンピュータで異なるアルゴリズムを用いて回答を作成し、より良い解を採用する。

2. 戦略

複数のアルゴリズムを用意して、2台のコンピュータを用いて並列計算させる。大小のスタンプを用いて修復率の高くなるように演算を行う。複数のアルゴリズムの中での最適解を提出する解答とする。

3. UI

ユーザーインターフェースは非対話CUIとする。閾値については引数で任意の値をユーザが入力できる状態にし、バッチやスクリプト等で連続実行を可能とする。

4. 開発環境

Microsoft Visual Studio 2010

Microsoft Visual C++ 2010

1. 問題解決の考え方

サイズの大きな問題について適切な解法を導き出すのは大変なので与えられた問題をあらかじめ適当に細分化しておく。そして細分化したそれぞれについて解法の探索を行った後、問題全体に修正をかけていく。

2. 問題へのスタンプの適用

問題サイズとスタンプサイズとの比較から適切なスタンプ及びその適用位置を決定していく。

3. 開発環境

Java Development Kit 1.60

```

15 2041 10 601 213 655
15 2042 10 501 296 652
15 2043 117 631 214 649
15 2044 117 531 218 617
15 2045 13 532 325 644
15 2046 13 535 293 643
15 2047 58 541 335 648
15 2048 66 555 311 638
15 2049 39 580 313 630
15 2050 32 558 315 637
15 2051 39 517 378 631
15 2052 34 610 368 629
15 2053 35 584 278 621
15 2054 58 539 395 624
15 2055 89 629 333 622
15 2056 38 589 375 619
15 2057 58 573 310 617
15 2058 10 535 213 616

```

図 解法の探索

1. 競技の進行について

競技開始直後、修復作業の準備として以下の処理を行う。

- ・問題のデータを読み込み、それぞれのデータをプログラム内に格納する。
- ・スタンプを大きさの順に並べかえる。
- ・初期画像、最終画像を照らし合わせ、差異の情報を作成する。

この差異の情報は修復する位置の決定に使用する。

次に修復作業を行い、修復が完了したら回答を提出する。

2. 修復作業について

スタンプと修復位置を乱数によって決め、その位置に対してそのスタンプが有効であるかを判定する。

有効であるかどうかの判断は、そのスタンプとフィールドとの相違を測定し、一定値以上の一致があれば有効であるとするとする。

有効であった場合はその位置を修復し、次の位置の決定に移るが、有効でない場合は、もう一度スタンプを選択しなおす。この時、位置の変更は行わない。

前半では大きなスタンプを使用し、後半は小さなスタンプで細かな修復を行う。

3. 回答の提出について

回答は回答用ソフトウェアを使用する。2台のノートパソコンを使用するため、結果の良いほうを最終的に提出する。

4. 開発環境

Microsoft Visual C++ 2010 Express Edition

1. 主な戦略

主な戦略として IDA*と焼き鈍し法を採用し、その他のメタヒューリスティックアルゴリズムも実装している。状況に応じて最適なアルゴリズムを選ぶ機構を取り入れているので、多くの問題に対して効率的に計算することができる。また、より効率よくスタンプの操作を実行するために、フィールドおよびスタンプを Bit 列で表現し、それらに対して Bit 演算を適用することで 20 倍の高速化に成功した。

2. 問題対策

今年度は自由度が高い問題のため、様々な性格の出題が予想される。そのため、予想問題の作成に力を入れ、多くの性質の問題に対応できるようアルゴリズムを強化した。

3. 開発について

今回は開発環境がメンバーそれぞれ全く異なっているため、マルチプラットフォームなプログラムを開発することとなり、下記のような状況となった。

4. 開発環境

言語: C++

OS: Windows7

Mac OS X

Ubuntu11

IDE: MicrosoftVisualC++2010

QtCreator

Xcode

Emacs

1. アルゴリズム

私たちのアルゴリズムは、とても単純です。初期画像と最終画像とを比較して、最終画像に最も近づきやすいと想定されるスタンプを決定し、それを押す座標を見つけてスタンプを押す(探索+修復)工程を繰り返します。この繰り返し工程により、最終画像になった場合に、「探索+修復」の工程を終了し、ファイルをサーバに提出します。

2. 特長

2.1 並列処理

アルゴリズム自体は、非常に単純なので、問題画像やスタンプが大きくなるにつれて処理時間を要します。このため、どれだけ手数の短い解を出せたとしても、時間内に提出できない可能性が想定されました。そのため、PC性能をフル活用できるCPUの『並列処理』が可能なようにプログラム・コードを工夫して、実装しました。これにより、デュアルコア以上のCPUを搭載したPCを利用することで、比較的高速の処理が可能になります(図1)。

2.2 データ送受信と処理のほぼ自動化

問題サイズが小さい場合には手動送受信において時間ロスが大きく、また大規模問題では手動操作では時間を要します。このため、開発したプログラムは、CUIをベースとして、ほぼワンクリックでデータ送受信・処理を可能とし、手動操作と途中経過表示をほぼ排除しました(図2)。

3. 開発環境

・Microsoft Visual Studio 2010 (C++及びC#)

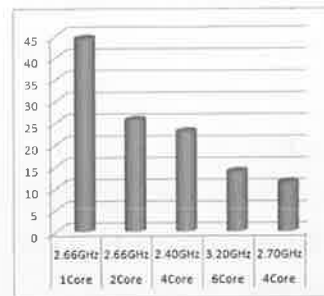


図1 CPUコア数による処理時間



図2 画面表示例

1. はじめに

最も少ない手数で完全に修復できる解を、すばやく求めることは難しいと思われる。総当たりで全てのパターンを試していけばいずれは求まるが時間がかかってしまうため、制限時間内に終わらない可能性がある。そこで、次のようなアルゴリズムで解くことにした。

2. アルゴリズム

本アルゴリズムではフィールドをいくつかに分割して考える。この分割されたエリアを一つ一つ確実に修復していくことで全体を一致させる。

どのように分割するかはスタンプのサイズや不一致なセル等の要素から決定する。このアルゴリズムの場合分割の仕方次第で解が変わるため、乱数を加える、分割前にいくつかスタンプを適用する、手動で分割するなどして変化

を持たせる。また、修復に用いたスタンプの組み合わせは記憶しておき、少しでも効率が良くなるようにする。

このようにしてさまざまなパターンを試し、手数の少ない解を求めていく。

3. ユーザーインターフェイス

ユーザーによる操作は基本的に開始や停止程度の簡単なものだけにして、面倒な操作は行わなくてもいいようにする。

4. 開発環境

Microsoft Visual Studio 2010

1. システム概要

本システム (Step By Step!) は、サーバから送られてくる情報を読み取る画像作成モジュール、画像修復を進めるマッチング・適用モジュール、提出用データを作成する回答作成モジュールによって構成される。

2. 構成モジュール

2.1 画像作成モジュール

送られてきたテキストデータから、初期画像、最終画像、及びスタンプを作成する。また、初期画像と最終画像との不一致部分の把握のため、これら2つの画像から XOR 画像を作成する。

2.2 マッチング・適用モジュール

スタンプと画像のマッチングを行い、マッチング率を計算し、その結果からスタンプ適用場所を決定する。これを繰り返すことで初期画像を最終画像に近づけていく。

2.3 回答作成モジュール

スタンプを適用した場所・スタンプの種類を保持し、回答用テキストファイルを作成する。

3. GUI 部

画像作成モジュールによって作成された初期画像、最終画像、スタンプならびに XOR 画像を表示する。また、回答データを読み込み、スタンプ適用後の画像を確認することができる。

4. 開発環境

Microsoft Visual C# 2010 Express



※画像は開発中のものです

1. 基本戦略

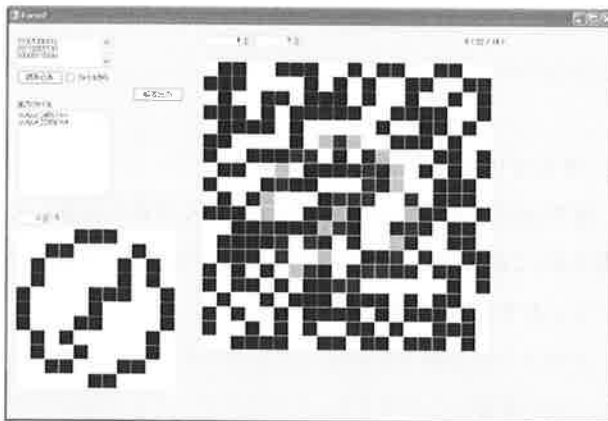
問題を読み込んだ後、初期画像と最終画像の差分を取る。この差分にスタンプを順次適用していくが、この戦略については現在のところ以下の3つを考えている。

- ① 適用した際に最も最終画像に近づくスタンプおよびその位置を求め、貪欲に修復を行なっていく方法。
但し、この方法を厳密に行うと時間が足りないので、少し簡略化したものを用いる。
- ② 現在の画像と最終画像の差が大きい領域を計算し、その付近について重点的に調べていく方法。
- ③ y座標(あるいはx座標)の小さな部分から、少しずつ順に修復していく方法。

これらのいずれか、あるいはこれらを組み合わせた方法でいくつか解を出力し、最も手数が小さくなったものを解答として送信する。

2. UI について

先に挙げた方法に加えて、手動での解答も可能にする予定である。そのため、分かりやすいUIを作成し、キーボード等で操作できるようにする。



3. 開発環境

Microsoft Visual Studio 2008

41

修復のスタンプ

—未来のために—

香川 (高松)

小笠原尚哉(4年) 大前 彩(4年)
河西 優樹(2年) 柿元 健(教員)

1. はじめに

今年度は日本各地が多くの自然災害に見舞われた。それらの災害で被害を受けた世界遺産や重要文化財などが修復されることを祈りつつ、我々はプロコンにおいて画像の修復のために修復のスタンプを未来に向けて押していく。

2. システム概要

今年度の競技部門は、最終画像の一致度、修復手数、回答時間を順位決定の要素としている。問題のフィールドの大きさによって勝敗の要素は変わると考え、フィールドの大きさに適したアルゴリズムを使用する。

3. 戦略概要

まず、配布される元のデータと最終画像のデータを用いて、XORにより修復すべき箇所を求める。次に、その修復

すべき箇所を大きいスタンプを用いて大まかに修復し、残りの修復箇所を小さいスタンプを用いて完成させる。

4. 開発環境

Microsoft Visual Studio 2008(C++)

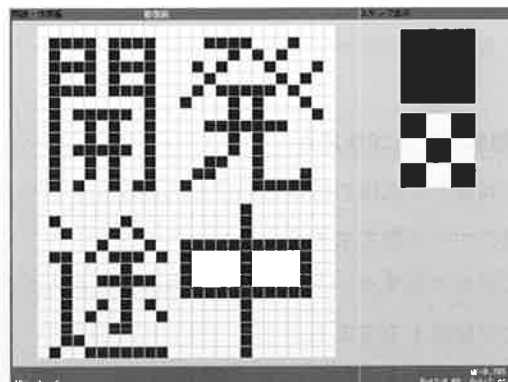


図1 開発中のシステムUI

競技部門

42

IRIS

—Img Recover Imprint Sys—

福 島

松島 弘(5年) 赤塚 篤(5年)
赤津 亘(3年) 小泉 康一(教員)

1. 問題状況の認識

問題を読み込むのにあたって、本プログラムではまず始めに初期画像と最終画像の XOR をとった画像を生成している。理由はこの画像の要素がすべて0になるようにスタンプを押す手順が本来求めるべき手順と同じ事であり、また状況の確認が容易である為である。

2. 解の導出方法

今回の競技は「修復」の選択肢が非常に多くなる為、厳密な解を求める方法では競技時間内に回答することが出来ないと判断した。

そこで、厳密な解ではなく、ある程度の近似解(最終画像にある程度近づくまでの手順)を求める方法で解を求めるアルゴリズムを実装した。(ルール上、1x1 のスタンプが必ず含まれている為、近似解でも問題ないと考えた。)

3. ユーザーインターフェース

ユーザーインターフェースは、サンプルクライアントを利用し、さらに下段に現在の探索状況を表示するようにした。

4. 開発環境

Microsoft Visual C# 2010



図1 プログラムのインターフェース

1. クライアントソフト

問題のファイルのダウンロード後、問題解決アルゴリズムを呼び出し、回答が終了するまで待機する。

アルゴリズムは複数用意し、2台のコンピュータでそれぞれ別のアルゴリズムを使用し回答を作成する。

出来る限り簡素、簡単なヒューマンインターフェースにすることで、ヒューマンエラーを防ぐ。

2. 問題解決アルゴリズム

元画像と完成後の画像との排他的論理和を取り、その画像について解を求める。

スタンプとスタンプを適用する座標との組み合わせ全てで探索木を考える。

全ての根の走査を行い、その結果適切だと思われる根から伸びている、全ての枝の走査を行う。各手において

必要ならスタンプを仮置きし、フィールド内に残った1の総数を計測する。その結果、適切だと思われる枝から伸びている全ての枝について再び走査を行う。

再帰的な探索がある深さまで達したら走査を終了し、現時点で最もよいスタンプの組み合わせを適用する。

以上を修復が完了するまで繰り返す。

走査方法と枝切りの条件が異なる複数のプログラムを用意する。

3. 開発環境

Microsoft Visual C++ 2008 Express

Microsoft Visual C++ 2010 Express

Microsoft Visual Studio 2010 Professional

1. はじめに

本プログラムでは、与えられた問題に対して複数のアルゴリズムを使用して複数の解答データを得る。その後、複数の解答データの中で最良のものを選択し、選択された解答データを提出する。

2. つぎに

問題画像データと完成画像データの排他的論理和をマップと名付け、このマップを使用して問題を解く。マップの左上から順に使用するスタンプの大きさなどから、マップと合致する可能性があるビット数を期待値として定め、期待値以上の結果が得られるスタンプを選択する。

他にも複数のアルゴリズムを用意し、問題を解かせる。

3. さいごに

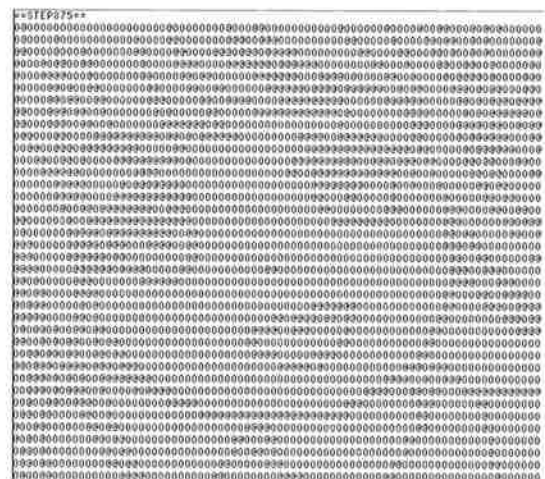
複数のアルゴリズムから出力された解答データから、最良の解答データを選択し、提出する。

4. おまけ

・ 開発環境

gcc 3.4.6

Linux 2.6.16



※画像は我が部のマスコット「やつよちゃん」です

45

修復屋

—どんな傷でも直してみせる—

都立(品川)

池田 基樹(2年) 手計 勝也(2年)
阿久津卓土(2年) 福永 修一(教員)1. はじめに

この問題では、ネットワーク経由で問題を取得し、与えられた処理して得られた解答データを送信する必要がある。そのため、我々は次のような構成でプログラムを作成した。

2. 構成2.1 システム制御部

全体のシステムを制御し、適切なタイミングで各部分を呼び出す。

2.2 ネットワーク制御部

問題データをサーバから取得し、解答データをサーバへと送信する。

2.3 処理部

問題データを処理し、解答データを作成する。

3. アルゴリズム

より良い解を求めるために、現在の解から探索をして最適解により近い解を新しく現在の解として選択していくことを繰り返す。

また、時間内に複数回解答を提出することが可能なので、解答の提出のタイミングを自動で制御させる。

4. ユーザーインターフェースについて

プログラムはコンソールアプリケーションとして作成した。また、思わぬ不具合を発生させることを防ぐため、なるべく人間が操作する部分を減らした。

4. 開発環境

Microsoft Visual Studio 2010

GNU Compiler Collection

46

Ikannol Mk-II

—この前のうどん美味しかったね—

久留米

大城 泰平(4年) 坂田 祐将(3年)
徳永 誠(4年) 黒木 祥光(教員)1. 基本的なアプローチ

押せるスタンプの種類と位置の中で、押した時に画像の評価値が最も高くなるものを貪欲的に採用していく。しかしスタンプのサイズが大きい場合、押せる場所全てについて評価値を調べると非常に時間がかかるため、このようなスタンプに関してはあらかじめ枝刈りを行う。この動作を初期画像が完成画像と同じ画像に変わるまで繰り返す。

2. 評価関数

評価関数は「スタンプを押すことで完成画像と一致するマスの数の増減」「完成画像と一致しないマスの集積度」「画像全体の完成画像と一致しないマスの密度」の三点について、実験的に得た最適な重みを乗じて足しあわせたものを計算し評価値とする。

3. 枝刈り

スタンプのサイズがある閾値以上の場合、スタンプの黒マスの密度と画像の密度を比較し、確率的にスタンプを押しても状態が改善しない可能性が高いと判断したら評価値を計算せずにそのスタンプは採用しないとする。

4. ユーザインタフェース

プログラムの起動等はプロンプト上で行うが、画像の状態を逐次画面に表示するアプリケーションを作成した。これにより問題の内容や計算状態を視覚的に確認することが可能となっている。

5. 開発環境

Vim - vi Improved 7.2

Gnu Compiler Collection 4.7.0

1. 競技の流れ

1.1 初期画像の加工

試合開始とともにデータをダウンロードする。その後、初期画像を複数領域へ分割し、符号化して圧縮する。

1.2 アルゴリズムの適用

今回の競技内容では、あらゆる場面で有効であるというアルゴリズムを考えるのは困難である。そこで私たちは、複数のアルゴリズムを組み合わせて使用して、最も手数が少ないものを送信する方針とした。

アルゴリズムの組み合わせ方は、初期画像のサイズやスタンプの数、サイズ、試合時間などによって変化させる。

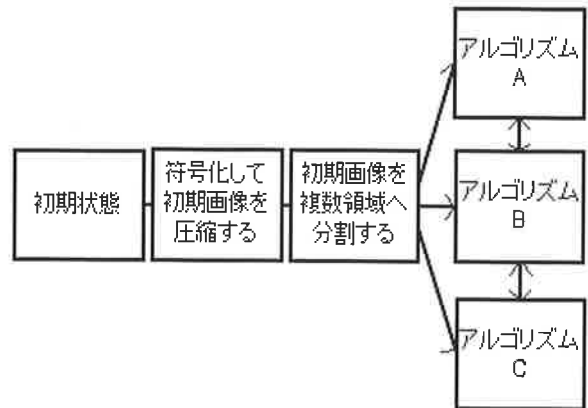
2. ユーザインタフェース

基本的に人間の手による操作は行わないが、アルゴリズムの切り替えなどを手動で行えるようにする。

また、試合状況を把握できるようなビジュアルライザを用意する。

3. 開発環境

Visual C# 2010 Express/Professional Edition



1. 競技の流れ

ネットワーク経由で問題を取得した後、最適解を求めていく。回答提出は、最適解が見つかった場合か、もしくは、(制限時間÷3)分ごと、および制限時間終了間際に前回提出したものよりも優れた手順が見つかった場合提出する。

最適解を求めるアプローチは次の通りである。まず最も大きなスタンプを用いて、スタンプの面積の半分以上が反転しなくなるまで画像を修正する。それ以上の修正ができなくなった時点で一回り小さなスタンプに持ち帰る。勝敗の判定は画像との一致が高いほうが価値となる点に注意して、開始時間からある程度時間が経ったら1x1スタンプを用いて仮の回答を求めておく。

2. 使用言語

C++

3. 開発環境

Windows 7 Home Premium

Microsoft Visual Studio 2010

Mac OS X 10.6

GNU Compiler Collection

Vim7.3

1. 競技の重点

- ・ より少ない手数で修復を終えること
- ・ その所要時間が短いこと

2. 修復の評価

プログラムでは、初期画像と最終画像から、反転が必要なセルを1、それ以外のセルを0とした「反転フィールド」を形成し、そこにスタンプを適用していく。反転フィールド内の1のセル数を評価基準とし、これが0になったとき修復完了と考える。

3. 修復

3.1 フィールドの分割

いくつかの矩形領域に区切り、領域を狭くすることでスタンプ適用の試行回数を減らすことを目的とする。

3.2 スタンプの選択

あるセルを左上原点とすると、そこにスタンプを適用した場

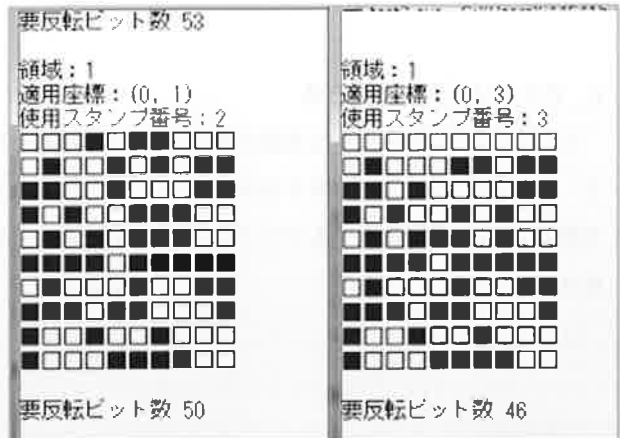
合、もっとも要反転セル数が減少するスタンプを使用する。

3.3 所要時間の考慮

反転フィールド内の要反転セル数が一定数以下になったとき、ダイレクトに1*1スタンプを適用する。

4. 開発言語

C言語 (Borland C++ Compiler)



1. 概要

今回の競技ではいかにスタンプの適用回数を少なくするかが重要となってくる。本プログラムでは、より大きなスタンプを適切な位置に適用することで、少ない手順で最終画像に導いていくことを狙いとし、時間制限についても考慮している。また、確実に最終画像に一致する解を最初に作っておき、徐々に最適解を求めていく。

2. プログラムの主な動き

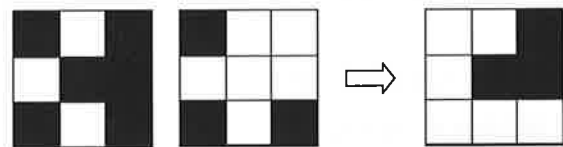
まず、初期画像と最終画像の差分を求める。そして、差分の大きさに応じて画像を複数の領域に分ける。それぞれの領域について、以下の処理を並列で行う。

1. スタンプを大きい順に選択して以下の2から4を繰り返す。
2. 現在の画像に選択したスタンプを適用して差分画像との一致度を計算し、保存する。

3. 2. の操作を縦横に1セルずつずらして行う。

4. 一致度の最も高い位置すべてにスタンプを使用する。その際、スタンプを使用する前の一致度と比べて、より低ければ使用しないこととする。

5. 以上の操作でサイズ1のスタンプまで使用すると差分画像のビットが全て0になり、初期画像と最終画像が一致する。



初期画像

最終画像

差分画像

図. 差分画像生成

3. 開発環境

Eclipse3.7/GCC4.5.3/boost1.46.1/doxygen1.7.4/git1.7.3.4

1. 概要

今回の競技部門のルールは、フィールドとスタンプの大きさや数、制限時間といった初期条件の取りうる組み合わせが多すぎるため、どのような場合でもある程度対処でき、短時間で探索を終了できるアルゴリズムを作成した。

2. アルゴリズムの選択方法

あらかじめ以下のような複数の探索アルゴリズムを用意しておき、スタンプの数や制限時間などの初期条件を判断材料に、使用するアルゴリズムをプログラム上で判断する。

3. アルゴリズム

まず、使用するスタンプの順番を決定し、それぞれのスタンプについて、フィールドを網羅的に探索し、一定以上のセルを修復できる場合は修復を行う。

このときスタンプを、そのスタンプを構成する値が1のセルが多い順に選択する。そのスタンプをフィールドに対して押した場合に、どの程度正解に近づくかどうかで修復を行うか判断する。この一連の動作を基本とする。

さらに、与えられた初期条件を基に、精密さよりも速度を重視した探索や、速度よりも精密さを重視した探索など、複数のアルゴリズムを用意しておく。

1. 概要

今年の競技部門は、与えられた課題を、なるべく少ない手数でクリアするというものでした。さらに、時間制限や一回の競技に複数の課題を解かなければならないなど、アルゴリズムの高速化も要求されます。このような問題の中、私たちは以下のような方法で解決するプログラムを開発しましたのでそれを解説します。

2. 問題解決へのアプローチ

2.1 遺伝的アルゴリズム

基本的な方針は、遺伝的アルゴリズムを用いた最短手の追求です。今回の問題形式ではスタンプを適応することによって画像の値が反転します。そのため、適応の順列によって結果が変わることがありません。これは個体の交配によって不必要な遺伝子の並びが発生しづらいことにつな

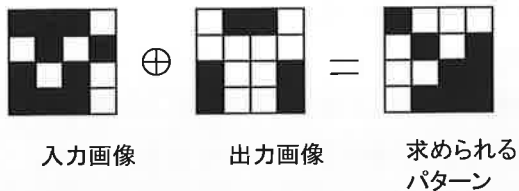
がります。また、順列に関係ないことで、スキーマ定理による短く有効な遺伝子が生き残っていくことが予想されます。この特徴から、致死遺伝子の発生による計算のロスに弱い遺伝的アルゴリズムと、今回の課題が比較的相性がいいことが分かりました。

2.2 インターフェース

今回は人間による操作が少なく、UIにはそこまで機能を必要としませんでした。しかし、計算状況を把握しておくことや、アルゴリズムだけでは完璧に行えない細かな修正など、人間の手で行ったほうが効率的な場面はいくつかあります。それを考慮したうえで、ユーザが理解しやすく、かつ必要な作業がすぐに行えるようなユーザインターフェースを目指しました。

1. はじめに

入力画像からスタンプ画像を複数回用いて出力画像を生成するというのが題意だが、これは入出力間の差分画像をスタンプの合成により求めることに等しい。このため、まず求めるべき差分画像を入出力の排他的論理和によって算出する。



2. 解法

スタンプをはみ出して押ししてもよいというルールより、割り算アルゴリズムを用いれば必ず解が求まることは保障できる。しかし得られる解は、ふつう最適なものでは

ない。また、条件によっていくつかの異なる解が得られることになる。このため、これらの複数の解から GA による最適化を行い、最終的な解を導く。

3. 高速化

上記の計算には大量の数値計算が必要となる。画像の合成に関しては画素ごとに完全に独立した計算であるため、プログラムの並列化が可能である。OpenMP ならびに GPGPU などによる高速化を検討する。

1. 問題を解く下準備

考えやすいようにフィールドの初期画像と最終画像の XOR を取ったものを用意しておく。

2. 問題を解く方法

XOR を取ったものにスタンプを適用し、全てが 0 になった時にフィールドが最終画像と同じになっている。だがスタンプを適用する場所がかなりの数あり、考える手が沢山あるので、スタンプと一致する部分がより多い場所を探せば手数が少なくなると考え、2つの方法を考えた。この2つの方法を使って、より良い結果のほうを選択する。

方法①

XOR を取ったものと、スタンプの 1 の一致度が高いものからどんどん適用していくという方法。ローラー作戦ではあるが、手堅い方法なので解を導くといった点においては信頼性が高いと考える。

方法②

XOR を取ったもの、スタンプ、それぞれを 2 値ヒストグラム化して類似値が高い最適な場所を探し、スタンプを適用していく方法。2 値ヒストグラムを使うことで全体をぼかして大雑把に見ることができるので、その一帯が大まかに 1 なのか 0 なのか、またスタンプも大まかに 1 なのか 0 なのか判断し、スタンプを適用することができる。判断材料としてヒストグラムを使うため実際に見てみるとスタンプと全然違ったフィールドの可能性があるので、それは方法①のように適用する前に一致度を確認する。探索効率が方法①よりは良く、探索時間の短縮を狙うのであればこの方法が良いと考える。

3. 開発環境

Microsoft Visual Studio 2010

MinGW(Minimalist GNU for Windows)

1. はじめに

今回の競技部門の課題は比較的小さな問題でも解の全探索を行おうとすると計算量が爆発してしまうため、最適解を求めるというアプローチは事実上不可能である。しかしながら、競技中は10回までなら解答の提出が認められている。

したがって、初期解から漸次解答を改善していくAIを作るのが最も効率が良いと考えた。

また、上記のようなAIを効率良く運用するためのUIを設計・実装することにした。

2. AIについて

上記のとおり、愚直な方法で解に辿り着くことは困難である。そこで解を次第に改善することができるヒューリスティックアルゴリズムを考え実装した。

高速化・効率化のため以下に挙げるような工夫を施した。

2.1 AIのマルチスレッド化

私たちが考えたアルゴリズムはマルチスレッド化が比較的容易だったため、これにより高速化を実現した。

2.2 初期解生成方法の工夫

初期解によってAIの能力は大幅に変化することが実験結果から分かっていたので、さまざまな手法の初期解生成アルゴリズムを用意した。

3 UIについて

初期解が重要なことは先に述べたが、この初期解生成と解の改善を個々に動作させるためのUIを開発した。

これにより、人間が問題に合った効率のよい初期解生成AIを実行することができるようになった。

また、必要に応じて人間が解に手を加えることができるようにすることで、万が一AIで解くことが難しい問題が出題されても対応出来るようにした。

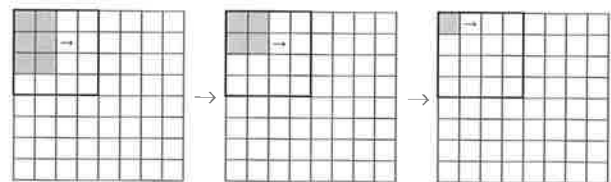
1. 概要

スタンプとフィールドの各セルを比較し最終画像に近づくかどうかを判定してスタンプを押すか判断する。

2. 評価方法

まず、一番大きいスタンプを押した場合と何もしない場合でどちらが最終画像に近いかを判定する。もし、押した方が最終画像に近い場合はスタンプを押し、残りの小さいスタンプで一番大きいスタンプの判定時の領域を完成図に近づける。

スタンプはサイズが大きい順に判定し使っていくようにする。



例として図のように初めに最大サイズの4×4のスタンプの判定後、4×4の領域において、スタンプのサイズが大きい順に左上から右下に向かうように順に判定をする。判定が終われば、右の4×4の領域の判定に移る。

3. 開発環境

Microsoft Visual C# 2010 Express Edition

1. システム概要

サーバーから問題データを取得し、データの解析を行う。全てのステップが求まった段階で解答データをサーバーへ出力する。処理軽減のため、各ステップが決まった際に修復画像を描画するかを選択できる。

2. アルゴリズム

全てのセルとスタンプに対して評価を行う。その中で最も修復できたものを選択する。次に評価する際はセルの変更があった箇所のみを評価する。しかし、フィールドの大きさとスタンプの数により処理が重くなるためフィールドを適度に分割して処理を行う。



図 1. 開発中の実行画面

3. 開発環境

Microsoft Visual C# 2010

1. はじめに

我がチームは、“これでかつる！”という信念のもとに次の3つのことを考えた。問題に対する柔軟化・短時間で少しでもよい解を得る・人の閃きの活用である。これらを実現するために、それぞれ異なった強みをもつメンバーを集め、各人の要望に応えられる開発環境で開発を行った。

2. システム環境

- OS:GNU/Linux システム
- 開発言語:Java, JRuby, Groovy
- 開発環境:Eclipse, NetBeans
- 実行環境:JVM(Jrockit)
- バージョン管理システム:Git

3. これでかつるために

(1) 問題に応じた柔軟な対応！

アルゴリズムは汎用的ではなく、特定の問題傾向に特化したアルゴリズムを複数用意した。ゲーム開始時に問題傾向を解析して最適アルゴリズムを選択することで、出題される問題に柔軟に対応して解を導くことができる。

(2) 処理の高速化！

マルチコア CPU の性能を最大限に使用するため、解法は並列で実行される。JVM もマルチコアに最適化されたものを用いている。これにより倍程度の速度向上が認められる。

(3) 閃きは優勝への近道！

問題傾向はアルゴリズムだけではなく人間による選択を可能にしている。実行アルゴリズム候補をプログラムから得た際にも、人間によって除外や追加も可能としている。これらによって柔軟に問題に対応することができる。

1. 競技開始まで

競技が開始する前に、事前に配られる解答者識別トークンを登録しておく。競技開始時の情報取得と進行中の解答はこのトークンを用いて通信を行う。

2. 競技進行中

競技開始後、まず初期画像、最終画像およびスタンプを取得する。この初期画像と最終画像を比較し、2つの画像で相違があるセル（以下、相違セルという）によって新たな画像を作成し、アルゴリズム実行の準備をする。

相違セルによって新しく作成された画像に、取得済みのスタンプを適用し、すべてのセルを修正する。以下に実行の順序を示す。

1. フィールドを複数のブロックに分割する
2. セルごとに、その周辺の濃度を求める

3. 濃度が高いセルほど優先的に修復を行う

4. 修正しきれなかったセルを 1x1 のスタンプで修正する

また上記の手順を基本とし、複数のアルゴリズムが組み上げられている。

3. 解答

複数のアルゴリズムのうち、そのいずれかで解答が用意できたら、他のアルゴリズムについても実行する。これを制限時間内に適切な回数行った上で、複数の解答のうち最も手数少ないものを選択し、これを送信する。

4. 開発環境

- 使用言語：Java
- 統合開発環境：Eclipse

1. システムの概要

本システムは GPU による計算部、CPU による優劣判断部、ネットワーク部によって構成される。

2. GPU による計算部について

本プログラムでは並列演算を用いて、スタンプを押した場合の結果をすべての場所で計算し、その中から解を選択する。その際、GPU を用いて、並列演算部を実現する。

GPU 計算部では、スタンプ押下後の反転ビット数と分散を求める。

3. CPU による優劣判断部について

GPU による計算部で求められた反転

ビット数と分散から、優劣を判断する。この2つのパラメーターから、反転ビット数が多く、マップ全体のビットの分散がより小さいものを解とする。

時間に余裕があれば、2つのパラメーターの重み付けのバランスを変更し、別の解を探して、再計算をする。

4. ネットワーク部

ネットワーク部では、2台のパソコンの間で解のやり取りを行い、両方で得られた結果のうち、最適な解を選ぶ。

5. 開発環境

Visual Studio 2010

CUDA toolkit

We work on a new image, it is called “*intermediate image*”. This image is created by XORing the value of the corresponding cells on the fields of the initial image and the final image. Therefore, we only need to find a technique to put the stamps on the intermediate field, changing the cells’ value until the intermediate field becomes a zero matrix.

We use a divide and conquer algorithm in order to solve the described problem. We divide the intermediate field into multiple small fields, they are called “*regions*” and their sizes are not fixed. For each problem, based on the sizes of the initial image, the final image and

the restricted time, we try to choose the sizes of the intermediate field’s regions so that they are the most suitable sizes.

We also construct a function that has some arguments and use some other techniques to appreciate the suitability of the stamps. For each region, we scan the stamps sequence and select several stamps of it so that they are the most optimal, then change the region to solved and label it with the candidate stamps. In solving the different regions, we are skilled so that do not affect the solved regions.

We used the C/C++ language and deployed in Windows platform.

Algorithms

First, filter the stamps of equal area, then reserve those which has the most "1" in the same area. Assuming that there are T stamps whose areas are different. Firstly sort the stamps by the area (N*M) from large to small, then compute a weight by using the stamps according to coordinates X: (1...n), Y (1...m). Computation rule is that if stamp can repair a point, weight will be added 1 to it; if stamp damages the picture which needs repairing, weight should be subtracted 1 from it. Then find a maximum weight, use the current stamp T_i to cover and to mark the repaired area which cannot be covered by the stamp T_i twice. Repeat this operation on T_i stamp for K times until all the stamps have been executed. After that, if there is any point which has not been repaired, we use 1x1 stamp to repair it until the image is fully restored.

Note:

N and M refers to the length and width of an image, $N * M$ refers to the image area which needs to be repaired.

T_i is the No. i stamp (i less than T) after filter.

K is the image area which needs to be repaired.

Programming Language:

Algorithms are implemented by using C++ and deployed in Windows platform.

1. Introduction

The task requires us to have small amount of stamping process. This requirement led us to use search algorithms. The optimal solution can be found using search algorithm. However, the time limit on each task is making it impossible to get the optimal solution. Therefore we decided to use search algorithms based on the input values. For the small cases, we are using optimal search algorithm and for the medium cases, we are using efficient search algorithm but not necessarily optimal. And last, for the big cases, we are using randomized algorithm.

2. Approach

- Small cases: Breadth-First search
- Medium cases: A* heuristic search
- Big cases: Monte-Carlo

3. Optimization

In order to utilize modern technologies, we decided to use capabilities such as multiprocessing and 64bit operating system. Thus we are dividing the algorithm into pieces and distributing the pieces to processors. Our search algorithm uses much memory. It led us to use 64 bit OS to avoid memory limit on 32 bit OS.

4. Environment

Microsoft .NET 2008 – Team edition

Visual C++

1. Problem

The problem of NAPROCK International Procon 2011 is to optimize number of stamps applied to the original image to achieve the final image. The image and stamp are an arrays[i,j] of boolean cells. Stamp is used to transform image by XOR with the image.

2. Algorithms

Rotating and positioning stamps on the initial image to achieve the final image is a hard problem. We try to use heuristic guide searching to achieve near optimal solution in a limited time constrain. Particularly, we apply greedy heuristic, where the stamp with transform the image the most is applied first. The solution obtained is then improved using other learning method such as genetic algorithm.

3. Programming language

Algorithms are implemented using C++ and deployed in Windows platform.

協賛企業広告一覧

第22回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。)

【特別協賛】

さくらインターネット(株)	98-99	(株)トヨタコミュニケーションシステム	110-111
東芝ソリューション(株)	100-101	日本マイクロソフト(株)	112-113
(株)インテリジェントウェイブ	102-103	ネクストウェア(株)	114-115
(株)サプライ	104-105	富士通(株)	116-117
チームラボ(株)	106-107	(株)ブロードリーフ	118-119
東京エレクトロンソフトウェア・テクノロジーズ(株)	108-109		

【一般協賛】

アイフォーコム(株)	120	日本電気(株)	126
(株)NTTデータ	121	(株)BCN	127
(株)オブティム	122	メガソフト(株)	128
(株)サミーネットワークス	123	弥生(株)	129
セイコーエプソン(株)	124	(株)ワコム	130
トヨタテクニカルディベロップメント(株)	125		

【広告協賛】

アラクサラネットワークス(株)	131	(株)エイビット	141
(株)エヌ・ティ・ティ ネオメイト	132	木村情報技術(株)	141
金沢電子出版(株)	133	(株)コムニック	142
京セラコミュニケーションシステム(株)	134	(株)ソルコム	142
(株)jig.jp	135	(株)ネットスプリング	143
(株)デザイン・クリエイション	136	三菱レイヨン(株)	143
(株)ドワンゴ	137	NEC フィールディング(株)	144
ネットワンシステムズ(株)	138	NPO法人インターネット・ラーニングアカデミー (ILA)	144
パナソニック電工インフォメーションシステムズ(株)	139		
(株)インフォクラフト	140	蛇の目ミシン(株)	144
(株)ウェブクラス	140	(株)ファインディックス	144

MEMO

A series of horizontal dashed lines for writing, spanning the width of the page.

大会役員・プロコン委員・事務局員

大会役員

大会会長	高等専門学校連合会会長	赤坂 裕	鹿児島工業高等専門学校長
副会長	高等専門学校連合会副会長	長澤 啓行	大阪府立大学工業高等専門学校長
副会長	高等専門学校連合会副会長	神野 稔	近畿大学工業高等専門学校長
副会長	高等専門学校連合会副会長	京兼 純	明石工業高等専門学校長
副会長	高等専門学校連合会理事	古屋 一仁	東京工業高等専門学校長
副会長	高等専門学校連合会監事	岸浪 建史	釧路工業高等専門学校長
副会長	第22回大会主管校校長	太田 泰雄	舞鶴工業高等専門学校長
副会長	高専プロコン交流育成協会理事	堀内 征治	

プログラミングコンテスト委員会

委員長	太田 泰雄	舞鶴工業高等専門学校	校長
副委員長	金寺 登	石川工業高等専門学校	電子情報工学科 教授
副委員長	三輪 浩	舞鶴工業高等専門学校	建設システム工学科 教授
ブロック委員	中村 庸郎	苫小牧工業高等専門学校	情報工学科 教授
ブロック委員	千田 栄幸	一関工業高等専門学校	電気情報工学科 准教授
ブロック委員	吉成 偉久	茨城工業高等専門学校	電気電子システム工学科 准教授
ブロック委員	柴田 博司	富山(射水)高等専門学校	電気制御システム工学科 教授
ブロック委員	片山 英昭	舞鶴工業高等専門学校	電気情報工学科 教授
ブロック委員	河野 清尊	米子工業高等専門学校	電子制御工学科 教授
ブロック委員	長尾 和彦	弓削商船高等専門学校	情報工学科 教授
ブロック委員	松野 良信	有明工業高等専門学校	電子情報工学科 准教授
企業委員	久保 慎一	ネクストウェア株式会社	
専門委員	田辺 正実	熊本(熊本)高等専門学校	人間情報システム工学科 教授
専門委員	桑原 裕史	鈴鹿工業高等専門学校	電子情報工学科 教授
専門委員	宮下 卓也	津山工業高等専門学校	情報工学科 准教授
専門委員	寺元 貴幸	津山工業高等専門学校	情報工学科 准教授
専門委員	佐藤 秀一	長岡工業高等専門学校	電子制御工学科 教授
専門委員	中道 義之	沼津工業高等専門学校	総合情報センター 講師
専門委員	小保方幸次	一関工業高等専門学校	制御情報工学科 准教授
専門委員	吉村 晉	サレジオ工業高等専門学校	情報工学科 特任教授
専門委員	鈴木 宏	長野工業高等専門学校	電気電子工学科 教授
専門委員	松山 勝志	東京工業高等専門学校	情報工学科 教授
専門委員	山崎 誠	長岡工業高等専門学校	電気電子システム工学科 教授
専門委員	小嶋 徹也	東京工業高等専門学校	情報工学科 准教授
有識者委員	松澤 照男	北陸先端科学技術大学院大学	情報社会基盤研究センター 教授
有識者委員	伊原 充博	高専プロコン交流育成協会	理事(事務局長)
主管校委員	片山 英昭	舞鶴工業高等専門学校	電気情報工学科 教授[兼任]
主管校委員	船木 英岳	舞鶴工業高等専門学校	電気情報工学科 准教授
主管校委員	井上 泰仁	舞鶴工業高等専門学校	電気情報工学科 講師
次年度主管校委員	松野 良信	有明工業高等専門学校	電子情報工学科 准教授[兼任]
前年度主管校委員	今井 一雅	高知工業高等専門学校	電気情報工学科 教授

主管校実行委員会(舞鶴工業高等専門学校)

委員長	太田泰雄(校長)
副委員長	金山光一(副校長)、背戸柳実(学生主事)、深津一也(事務部長)
事務局長	片山英昭(電気情報工学科) 事務局長補佐 三輪浩(建設システム工学科)
【総務・企画】	金山光一、金森満、村上美登志、宮野敏男、宮崎昭仁、奥村昌司、山田耕一郎、西山等、町田秀和、宮元健次、小野栄津夫、中高賢也、嵯峨信子、由良耕士、芦田康弘、大谷康祐、久保聡志、池澤拓史、半田健、川西祐子、荒賀美枝子、岡本幸子
【救護・学生交流】	背戸柳実、小野伸一郎、荒川吉孝、中島賢也、嵯峨信子、清水智苗美、山口絵美
【受付】	川田昌克、奥村幸彦、亀谷陸、小松幸恵、仲川 力、松宮正明、南部佳紀、吉田容子、萩野弘貴、増田智仁
【応接・案内】	三川譲二、深津一也、上杉智子、宮崎操、西海彰二、住友靖代、矢敷沙也加
【式典】	新池一弘、亀谷陸、上杉智子、三田達夫、南部佳紀
【輸送/弁当/駐車】	高谷富也、尾上亮介、石川一平、坂本英輔、山中啓太郎、橋本豊、渡邊彰宏、山崎基史、北代浩次、溝畑仁
【広報/記録】	野間正泰、谷川博哉、垂谷茂弘、岡田浩嗣、豊田香、福井繁雄、森田義己、能勢嘉朗、榊田勲
【国際交流】	荒川吉孝、松井信義、ジョナサン・ケイ、竹内規裕
【会場設備】	船木英岳、玉田和也、野毛宏文、児玉圭司、眞柄賢一、能勢嘉朗、西村良平、福山慎太郎、大和辰徳
【課題/自由部門】	平地克也、清原修二、徳永泰伸、伊藤稔、四蔵茂雄、釣健孝、古林達哉、西川隼人、櫻井一樹
【競技部門】	井上泰仁、篠原正浩、内海淳志、加登文学、小林洋平、南裕樹、丹下裕、村上信太郎、石井貴弘
【学生指導】	中川重康、芦澤恵太
【中学生対応】	竹澤智樹、田村修一、吉永進一、清原修二、四蔵茂雄、釣健孝、西川隼人、芦田康弘、川西祐子

大会事務局

〒101-0003 東京都千代田区一ツ橋2-1-2 学術総合センター 10階
 高等専門学校連合会(独立行政法人 国立高等専門学校機構内)
 TEL: 03(4212)6815(代表) FAX: 03(4212)6810
 担当 宮倉尚之 独立行政法人国立高等専門学校機構企画課長
 西山隆宏 独立行政法人国立高等専門学校機構企画課課長補佐
 黒田美穂 独立行政法人国立高等専門学校機構企画第二係長

委員会事務局

〒625-8511 京都府舞鶴市宇白屋234番地
 舞鶴工業高等専門学校 学生課学生支援係
 TEL: 0773(62)8882 FAX: 0773(62)8889

NAPROCK事務局

〒541-0057 大阪府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 伊原充博 NAPROCK事務局長
 田中佐和子 NAPROCK事務局次長

第21回プログラミングコンテスト

2010年10月16日(土)・17日(日)

主管校：高知工業高等専門学校 会場：高知市文化プラザかるぽーと

課題部門



自由部門



競技部門



表彰式



第22回プログラミングコンテスト

2011年12月22日(木)・23日(金)

主管校：舞鶴工業高等専門学校

会場：舞鶴市総合文化会館



舞鶴市総合文化会館
 京都府舞鶴市浜2021番地
 TEL:0773-64-0880