

# ドライブレコーダのデータから旅行の思い出動画を自動生成する web サービスの実装

尾頭 花奈\* 石黒 祥生\*† 椎尾 一郎‡ 武田 一哉\*†

**概要.** 本研究では、ドライブ旅行の思い出を楽しく振り返ることを目的に、ドライブレコーダのデータを活用した思い出動画自動作成システムを提案する。最近のドライブレコーダは、車内外の様子を高解像度動画として記録するため、紀行ムービーの素材として最適である。ドライブレコーダで撮影された映像データは、SD メモリカードから簡単に入手できるが、膨大なデータを編集するには多くの時間と労力が必要なため、ほとんど利用されていない。そこで、思い出動画を自動作成するシステムを開発した。提案システムは名所、会話の盛り上がりなど、旅行において重要だと思われるシーンを判定して動画を要約し、文字テロップ・地図・合成音声の追加を行う。本稿では、自動編集システムの設計に加え、広く一般に使用してもらうために開発した web サービスの実装と評価について述べる。本研究は、今まであまり注目されてこなかった自動車乗車後の UX 向上に貢献する。

## 1 はじめに

スマートフォンやウェアラブルカメラの登場により、個人の体験を映像として簡単に記録できるようになった [9, 14]。また、SNS の普及により、体験を他者に共有し、コミュニケーションをとる機会が増えている。自動車にも多数のカメラやセンサ類がついており、自動車乗車中の体験が細かく記録されている。しかし、これらのセンサ情報は自律走行や交通安全以外の目的にはあまり利用されていない。

ここ数年、車両から得られる映像データやセンサ情報をエンターテインメントシステムに利用しようという試みがなされている [12, 13, 21]。しかし、これらは乗車中の UX 向上を目的としており、乗車後の UX 向上に焦点を当てた研究は極めて少ない。そこで本研究では、旅行の振り返りに焦点を当て、乗車後の UX 向上を目的とした。

本研究では、近年急速に普及しているドライブレコーダに着目した [16]。ドライブレコーダの主な用途は交通事故や煽り運転車との遭遇のような非常事態への対策である。そのため、ドライブレコーダに記録されたデータは通常はほとんど活用されず、記録媒体の容量を超えた内容から廃棄される。最近のドライブレコーダは GPS 機能を搭載したモデルや複数台の高品位カメラを搭載したモデルも多く [10]、ホームビデオや SNS で共有する動画の素材として最適である。

しかしながら、長時間の走行動画を単に再生するだけでは冗長なシーンが多いため、楽しい動画にするには適切な編集が必要である。ドライブレコーダ

から得られる大量の動画ファイルから印象的なイベントが記録された箇所を選択し、結合し、エフェクトを追加するには多くの時間と労力が必要であるため、これら全てを毎回手作業で行うのは困難である。

そこで本研究では、これらの課題を解決するため、ドライブレコーダのデータからドライブ時に特化したイベントを検出し、車旅行の思い出動画を自動生成するシステムを開発した。今回作成したシステムは、自動車から得られる映像データの新しい活用アイデアをサポートし、これまであまり検討されてこなかった自動車乗車後の UX 向上に貢献する。

本システムのプロトタイプについては、第 69 回 UBI 研究発表会 [20] と AutomotiveUI '21[1] で発表した。本稿ではこれを発展させ、一般公開用に開発した web サービス全体の設計について述べる。

## 2 関連研究

### 2.1 ドライブレコーダの活用

ドライブレコーダのデータを交通安全に活用する試みは多数存在する。例えば、ドライブレコーダを交通事故発生予測に活用する研究や [3, 19]、路上駐車の実タイム検知に活用する研究 [15]、注意散漫なドライバーを自動検出する研究 [4] などがされている。本研究では、これらの取り組みとは違い、ドライブレコーダのデータをエンターテインメントに活用することを目的とする。

### 2.2 動画の自動要約技術

動画の自動要約に関する研究は多数されてきた。例えば特異値分解 (SVD) とクラスタリングによる映像要約技術や [7]、シーンの変化をグラフのモデル化により検出する動画要約技術 [17]、ビデオフレー

\* 名古屋大学

† 株式会社ティアフォー

‡ お茶の水女子大学

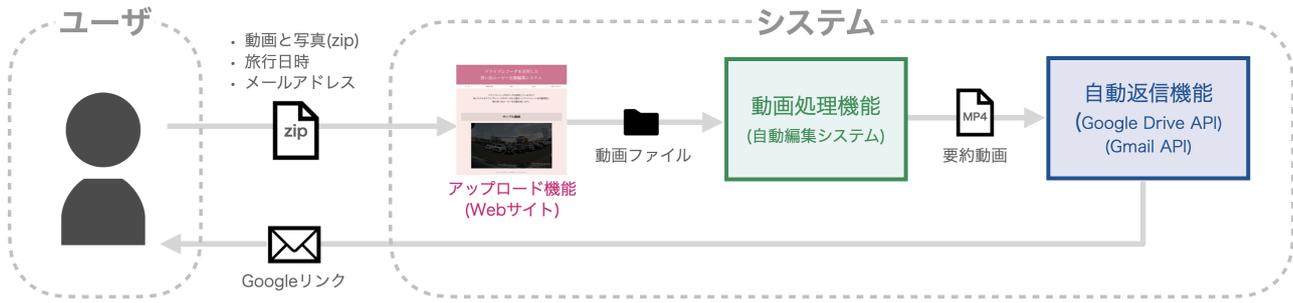


図 1. 構築した web システムの概要図.

ムからの色特徴抽出と k-means クラスタリングアルゴリズムに基づいた動画要約技術 [5] などがある。これらの研究では主にシーンの変化に着目し、要約を行っている。しかし、車内映像に関しては、自動車に乗車中は人の動きは少ないため、視覚的なシーンの変化はほとんどない。また、旅行においては、全てのシーンを網羅的に切り取る必要はなく、例えば、「目的地や観光名所付近の映像は必要だが高速道路や住宅街の映像は不要である」など、走行位置や周囲の状況によってシーンの重要度が大きく異なる。そのため、これらの手法はドライブ旅行の映像の要約には適さない。

### 2.3 身近な製品を利用した体験要約

身近な製品を使用することで、生活シーンの切り取りや要約、記憶支援を実現する研究も多い。Comic-Diary は、携帯情報端末 (PDA) やキオスク端末を利用し、個人のプロフィールや行動履歴、他のユーザーとの交流記録などから、学術会議参加や博物館見学の日記を漫画形式で自動生成する [18]。Video-Recording Your Life では、ウェアラブルカメラである GoPro と、スマートフォンの加速度センサを利用し、日常生活を記録した映像から利用者が興味を引くと思われるシーンを自動抽出する [2]。マイクロソフトのプロジェクト「MyLifeBits」で使用されたウェアラブルカメラ SenseCam には多数のセンサが組み込まれており、身につけているだけで光量の著しい変化やカメラの前の体温などを自動で検出して写真を撮影する [11, 6]。身近な製品を利用することは、多くの人に使ってもらおう一つのきっかけになると考えられる。本研究では、近年急速に普及しており、手軽に購入可能なドライブレコーダの活用に着目し、要約の対象を自動車乗車中とした。

また、世に広く普及している旅行の自動記録システムとして、Google Maps のタイムライン機能がある。このサービスは、移動経路や滞在場所を地図上にプロットし、滞在時間などの情報と合わせた旅行の記録を自動でまとめる。また、Google photo に保存されている写真を合わせて表示することもできる [8]。本研究でも、移動経路、滞在場所、時刻な

どの情報や、ユーザが旅行中に撮影した写真を自動で動画に組み込むように設計した。

## 3 web システム概要

提案する自動編集システムは、ドライブレコーダから得られた動画ファイル、GPS のログファイル、ユーザが旅先で撮影した写真をもとにハイライトシーンを抽出し、サウンドエフェクトやトランジション効果を加えながら旅の要約動画を自動生成する。また、自動編集システムを広く使ってもらうための web システムを構築した。本 web システムの概要を図 1 に示す。提案システムは <https://bitokana.com> から誰でも利用可能である (図 2)。対応ブラウザは Safari, Chrome, Firefox であり、PC から閲覧することを前提としている。ユーザは本サイトに必要事項を入力し、ドライブレコーダのデータをアップロードするだけで思い出動画を手に入れることができる。web サイトの構築のためにさくら VPS を使用し、SSL 証明書には Let's Encrypt を用いた。VPS の CPU は仮想 4Core、メモリは 4GB、SSD は 400GB、OS は CentOS を選択した。本システムは Python の Web フレームワークである FastAPI を利用している。

### 3.1 システム使用手順

アップロード機能は web サイトのトップページに配置した。ユーザは本システムを利用するにあたって以下の操作をする必要がある。

#### 3.1.1 対応機種

ドライブレコーダは機種によってカメラの撮影範囲、保存されるデータの拡張子、1 ファイルあたりの撮影時間などが異なるため、使用する機種によって自動編集システムの処理が異なる。本稿では、水平方向に 360° 撮影可能であり、GPS 測位機能が搭載されている Yupiteru 社 Q20-P を使用した際の処理について述べる。このドライブレコーダは動画データと位置情報を、それぞれ MP4 ファイルおよび NMEA (National Marine Electronics Association) ファ



図 2. 作成した web サイトを通してデータをアップロードする様子. データ送信前の画面 (左), アップロード中の画面 (中央), データ送信後の画面 (右).

イルとして SD カードに記録する. いずれも約 1 分間のデータを 1 ファイルとして (以下, 1 分録画ファイル), 複数のファイルが生成される. NMEA ファイルには, GPS により検出された位置情報が 1 秒毎に書き込まれる. MP4 ファイルの映像の記録形式は魚眼と 2 分割 (上半分が車外, 下半分が車内の様子) が選べるが, 本システムでは 2 分割の形式で記録していることを前提とした. なお, 現段階では Yupiteru 社 Q20, Q20-P 以外の機種では検証していないため未対応となっているが, 今後は他の機種にも対応する予定である.

### 3.1.2 事前準備

ユーザは以下の手順によりアップロード用のファイルを事前準備する.

- (1) “movie” という名前の空のフォルダを作成する
- (2) microSD カードに入っている全てのフォルダを movie フォルダの中にコピーする
- (3) movie フォルダを圧縮する

また, 旅先で撮影した写真 (但し Exif 情報があるものに限る) を要約動画に組み込みたい場合は上記の手順 3 の前に追加で以下の処理を行う必要がある.

- (a) “photo” という名前の空のフォルダを作成する
- (b) photo フォルダの中に要約動画に組み込みたい写真を入れる
- (c) photo フォルダを movie フォルダの中に移動する

### 3.1.3 データの送信

ユーザは上記の手順で作成した zip ファイルに加え, 以下の 3 つの情報をシステムに送る必要がある. 一つ目は, 完成した動画をユーザに送信するためのメールアドレスである. 本システムが生成する要約動画はデータサイズが大きいため, メールに動画をそのまま添付することはできない. そこで, 本システムは動画の共有に GoogleDrive のリンクを利用

する. そのため, 本システムの利用には Google アカウントが必要であり, G メールアドレスの入力を必須とした. 二つ目はドライブレコーダの機種である. 前述のように, ドライブレコーダは機種によって映像の記録形式や保存データのディレクトリ構造が異なる. そこで, ドライブレコーダの機種をユーザに選択してもらうように設計した. 三つ目は旅行期間である. 通常ドライブレコーダから取り出した microSD カードの中には要約したい旅行期間以外のデータが含まれている. そこで, アップロードしたデータ全てを要約するのか, それとも一部を要約動画の素材として使用するのかを選択できるようにし, 要約動画に使用したい動画データの期間を設定できるようにした.

以上の 3 項目を入力し送信ボタンを押すと, データがサーバに送信される. データの送信が完了するとユーザに受け付け完了のメールが自動送信される.

### 3.2 動画処理

サーバにアップロードされた zip ファイルは, 以下の手順で要約動画に変換される. まず, アップロードされた zip ファイルを解凍する. そして, システムがデータの管理をしやすいように, 解凍したファイルのファイル名をユーザのメールアドレスに変更する. ここで, アップロードされたデータのうち一部の期間のデータのみを要約動画の素材として利用する場合は, NMEA ファイルから取得した日時の情報を用いて, ユーザが指定した時間外のデータを削除する. こうして出来上がったファイルを自動編集システムに読み込むことで, 要約動画が生成される.

自動編集システムは, Python3 によって記述され, コマンドラインツールである FFmpeg を利用して動画処理を行なう. 自動編集システムの詳細な設計については 4 章で説明する.



図 3. 自動編集システムによる明るさの調整. 調整前 (左) と調整後 (右).

### 3.3 自動返信機能

作成された要約動画は以下の手順で Google ドライブのリンクとしてユーザに共有される。まず、自動編集システムが作成した動画を Google Drive API を用いて Google ドライブにアップロードし、動画の共有リンクを取得する。この時、ユーザがリンク先の動画を閲覧できるように、ユーザが入力した G メールアドレスを利用してユーザに動画の共有権を付与する。そして、取得した共有リンクをメールの本文に貼り付け、Gmail API の自動送信機能により、ユーザのメールアドレス宛に自動送信する。

以上の処理により、ユーザは要約動画を取得する。また、全ての処理を終えたら、アップロードされた全データを自動で消去し、サーバ上に個人情報を残さないようにした。

## 4 自動編集システム

本システムは以下の 7 つの処理を行う。

### 4.1 動画のサイズの調整

Q20-P ドライブレコーダから得られた動画データは、横縦が 2048×1536 画素、28fps である。1 フレームは上下に 2 分割され、それぞれに車内外の映像が記録されている。車外の映像を使用するときは上半分中央部の 1360×765 画素を、車内の映像を使用するときは下半分中央部の 1360×765 画素を切り出し、1280×720 にリサイズして使用する。なお、本システムが最終的に出力するムービーの画素数も 1280×720 画素である。

### 4.2 動画の明るさの調整

ドライブレコーダの主な目的は車外の様子の撮影であるため、360° 全天球カメラで撮影された映像は窓から見える景色の光量に合わせた露出となっている。そのため、昼間でも車内の映像は暗く、夜間に至っては暗くてほとんど何も見えない状態であった。そこで、車内の映像に対しガンマ補正、彩度調整、コントラスト調整を行った (図 3)。

### 4.3 ハイライトカットの自動判定と切り取り

本システムは、ドライブ中に注目すべき事象が発生した際に撮影されるカット (以下、ハイライトカッ

ト) を自動抽出する。現在実装している事象は以下に示す 6 種類である。

(1) **出発時**: 車両の ACC (アクセサリ) 電源が ON になったタイミングを出発時と判定し、そのときの車外の映像を 7 秒、車内の映像を 10 秒切り取った。車外の映像には NMEA ファイルから取得した時刻の情報を用いて、文字テロップと合成音声の挿入を行った。なお、ドライブレコーダは ACC 電源の供給に連動して自動的に記録を開始/終了するため、運転中だけが記録対象である。本研究ではこれを利用し、録画時間が 10 分以上開いた場合、休憩地点もしくは目的地に到着したと判断する。

(2) **到着時**: ACC 電源が OFF になったタイミングを到着時とし、車内の映像を 10 秒、車外の映像を 7 秒切り取った。車外の映像には、NMEA ファイルから取得した時刻と座標の情報、場所情報 API を用いて、到着時の時刻と住所を取得し、文字テロップと合成音声の挿入を行った (図 4)。

(3) **県境を通過したとき**: NMEA ファイルから取得した座標情報と場所情報 API を用いて都道府県名を取得した。そして、取得した都道府県名の変化から県境を判定し、県を跨いだ瞬間の車外の映像を 3 秒切り取った。また、都道府県名を文字テロップの挿入と合成音声の挿入により示した (図 4)。

(4) **名所付近を通過したとき**: NMEA ファイルから取得した座標情報と場所情報 API を用いて周辺の大規模施設や地名などの地域情報を取得した。そして、取得した情報から名所付近を通過したことを判定し、そのときの車外の映像を 3 秒切り取った。また、地域情報を文字テロップの挿入と合成音声の挿入により示した。このとき、情報を表示するのは場所情報 API の Score (取得した地域情報の確信度) が 99.9 以上の場合に限定し、名所のすぐ近くを通過するときだけ表示を行った (図 4)。

(5) **高速道路の出入り口**: NMEA ファイルから速度を取得し、1 分間の平均速度を計算した。本研究では、1 分間の平均速度が 60km/h 以上なら高速道路と判断し、60km/h 未満なら一般道であると判断する。また、ETC (Electronic Toll Collection System) ゲートは 20km/h 以下での徐行が求められていることから、高速道路と一般道の切り替わる箇所のうち、10km/h~20km/h の箇所を高速道路の出入り口であるとし、その部分を 4 秒切り取った。

(6) **会話が盛り上がっているとき**: MP4 ファイルから取得した音声データにバンドパスフィルターを適用し、日本人の音声周波数帯域とされる 300~3400Hz の周波数帯域を抽出した。そして、抽出した音声データの振幅が一定の閾値を超えた箇所を会話が盛り上がっているシーンと判定し、そのときの車内の映像を 6 秒切り取った。また、Python の SpeechRecognition ライブラリを利用して会話に字幕を表示した (図 4)。



図 4. 自動編集システムが切り出すハイライトカット. 左から順に到着時, 県境を通過したとき, 名所付近を通過したとき, 会話が盛り上がっているとき.

#### 4.4 旅先で撮影した写真の追加

スマートフォンやデジタルカメラで撮影された写真には, 撮影日時や位置情報を含む Exif (Exchangeable image file format) 情報が記録されている. そこで, 旅先で撮影した写真に Exif 情報が埋め込まれていた場合, そこから撮影日時を取得し, 作成動画の適切な位置に表示させる処理を行なった. 画像の表示時間は 1.2 秒に設定し, 縦長の写真の場合は左右に余白をつけることで 16:9 の比率に合わせた.

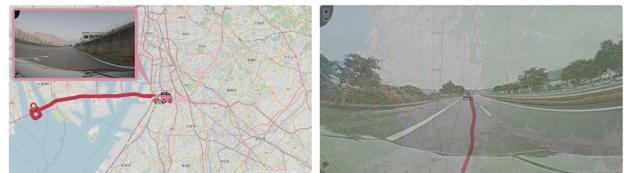


図 5. ハイライトシーンがなかった 1 分録画ファイルに対する 2 種類の処理. 景色の変化が大きいときの処理 (左) と景色の変化が小さいときの処理 (右).

#### 4.5 地図の生成

車旅行では, 移動経路や滞在場所の地理情報も重要である. そこで, ハイライトカットの対象にならなかった 1 分録画ファイルに対しては, 地図を伴った表示を行なった. このために, Python の Folium ライブラリを用いて取得座標を地図上に可視化する処理を行った. 出発してから目的地に到着するまでに通過した道を赤色の線で示し, 目的地に到着したときには赤色のマーカーをプロットした. そして, 再出発時には, これまでに通過した道を赤色の線からピンク色の線に変更し, 再出発してから次の目的地に到着するまでに通過した道を新たに赤色の線で示した. この地図を車外映像と組み合わせるために, 2 種類の表現方法を採用した. まず, それぞれの 1 分録画ファイルの 1 フレーム目を取り出し, 一つ前の 1 分録画ファイル第 1 フレームの対応画素の RGB 値の差分の絶対値の総和を計算する. この総和を, 設定した閾値と比較して以下の処理を行なう.

**差分が閾値以上だった場合:** 周囲の景色の変化が大きいと判断する. 地図上に, 移動経路, 車外の映像から切り取った静止画, 車のイラストを載せた静止画を, 1 分録画に対して 1 枚作成しこれを 0.3 秒程度表示する. 車外映像領域は 1 秒のコマ落としにした (図 5. 以下, 地図主体表示).

**差分が閾値未満だった場合:** 景色の変化が少ないと判断する. 1 分録画から切り出した車外映像を 100 倍速にして, 移動経路を表示した地図をアルファブレンディングする (図 5. 以下, 100 倍速動画).

#### 4.6 処理の優先順位の調整

**切り取り箇所重複の回避:** ここまでの手順により作成された動画ファイルを時系列順に繋げることで,

思い出動画を作成することができる. しかし, このままでは同じシーンが, 複数のハイライトカットの対象となり, 重複して切り取られてしまう可能性がある. そこで, ハイライトカットに優先順位を設定し, 同一箇所が複数回切り取られることがないように調整した.

**地図表示手法の調整:** 作成したハイライトカットや地図表示をそのまま繋げると, 画面が過剰に切り替わり, 見辛い動画になってしまう. そこで, 1 分録画ファイルに対する処理を指定する配列に対して以下の処理を行い, 変化を抑えた.

1. 地図主体表示が 100 倍速動画に挟まれている場合, 地図主体表示の代わりに 100 倍速動画を表示する.
2. 100 倍速動画が 3 連続未満なら, それら全てを地図主体表示に変更する.
3. 地図主体表示が続かない場合, その地図情報は使用しない.

#### 4.7 動画の連結とエフェクトの追加

以上の処理によって生成された素材に対して, 以下の視覚音響効果とファイルの結合処理を行うことで思い出動画が完成する.

**吹き出しの拡大・縮小:** 地図主体表示とその他の動画ファイルの切り替えをスムーズに行うため, 吹き出しのエフェクトを使用した (図 6).

**フェードイン・フェードアウト:** 出発時にはフェードインの処理を, 到着時にはフェードアウトの処理を加えることで一旦画面を暗くし, 目的地に到着したことを示した.

表 1. 使用したデータの情報と自動編集処理にかかる時間、手動編集の作業時間。

旅行期間	乗車時間	ファイルサイズ	ファイル圧縮時間	アップロード時間	自動編集時間	手動編集時間 (P1)	手動編集時間 (P2)
5時間 18分	98分	12.42GB	4分	35分	36分	4時間 47分	3時間 48分
8時間 24分	108分	15.61GB	5分	45分	36分	4時間 37分	3時間 08分
8時間 54分	159分	20.1GB	6分	58分	50分	3時間 50分	3時間 45分



図 6. 吹き出しの拡大エフェクトによるトランジション。



図 7. 放射状に広がるワイプエフェクト。

**放射状に広がるワイプエフェクト：**通常速度の動画と 100 倍速動画の切り替えをスムーズに行うために、放射状のワイプエフェクトを加えた (図 7)。

**ジングル：**県境と名所付近を通過したとき、吹き出しのエフェクトによるトランジション、放射状に広がるワイプエフェクトにはそれぞれ効果音をつけた。100 倍速動画にはアップテンポのジングル、地図主体表示にはミディアムテンポのジングルをつけた。

**動画ファイルの結合：**以上の処理で作成された動画はすべて、切り取り箇所の該当時間に対応した名前がつけられ、完成動画用フォルダに保存される。たとえば、80 ファイル目の 30 秒地点から 4 秒間の映像を切り取った場合、“00000080\_30.mp4” という名前がつけられる。そこで、全ての処理完了後に、完成動画用フォルダにある処理済みの動画を名前順にソートし、全てのファイルを連結して一つの動画を生成した。

## 5 処理時間の評価

著者の 1 人が母親と 2 人で実施した 3 回のドライブ旅行のデータを本システムに適用した。また、自動編集と手動編集の違いを検証するため、双方とも動画編集初心者である、母親 (P1) と兄弟 (P2) に手動で動画編集をしてもらった。

P1：50 代女性、旅行に参加

P2：20 代男性、旅行に不参加

編集には iMovie を利用した。本システムに適用した際にかかった時間と 2 人の被験者が手動編集にかかった時間を表 1 にまとめる。ドライブレコーダの

データは、1 時間半程度のドライブで約 10GB と大きいので、アップロードと自動編集システムの処理に時間がかかる。加えて、ユーザはデータのアップロード中はブラウザを閉じることはできない。しかし、今回の実験では、送信ボタンを押してから要約動画の受信までにかかった時間は、実際の乗車時間よりも短く、手動編集する際にかかる時間に比べて大幅に短かった。また、送信ボタン押下後はユーザの作業は一切不要であるため、実用的であると言える。

## 6 議論

多くのドライブレコーダは車外のみを撮影し、車内の様子は記録できないため、今回は旅行ルートの要約と景色の切り取りに重点を置いたシステム設計にした。本システムは、手動編集では難しい周辺地域情報の取得やルートの可視化が可能である。一方、会話の内容に基づく動画の切り取りは難しく、現在は実装されていない。会話の内容による自動要約を実現するには、システムが会話の内容を適切に理解する必要があるのに加え、人によって面白いと思う会話や思い出に残したいシーンが大きく異なるといった課題がある。今後、会話の内容に基づくシーンの抽出を行う際には、例えば、システムが表情認識や音声感情認識を用いてハイライトカットの候補を絞り、その後、ユーザが動画に組み込みたいシーンを選択するなどの半自動化処理を検討したい。なお、自動編集システムと手動編集の詳細な比較評価、及び被験者アンケートについては [1] を参照されたい。

## 7 おわりに

本研究では、ドライブレコーダのデータを活用し、旅のハイライトを自動検出して思い出動画を自動生成するシステムの提案と実装を行った。ユーザは web サイトにデータをアップロードするだけで、ドライブ旅行の思い出を紀行ムービーとしてまとめることができる。今後はより多くのユーザが利用できるよう、対応機種を増やす予定である。

一方、自動運転の技術が成熟し、急速に普及しつつある。自動運転車に搭載されている多数の高画質カメラ、センサ類を利用して紀行ムービーを作成することで、現在の手法に比べてはるかに高いユーザエクスペリエンスを提供できると考えている。

## 参考文献

- [1] K. Bitto, I. Siio, Y. Ishiguro, K. Takeda. Automatic Generation of Road Trip Summary Video for Reminiscence and Entertainment using Dashcam Video. In *13th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '21, New York, NY, USA, 2021. ACM.
- [2] D. Buschek, M. Spitzer, and F. Alt. Video-Recording Your Life: User Perception and Experiences. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, p. 2223–2228, New York, NY, USA, 2015. ACM.
- [3] F.-H. Chan, Y.-T. Chen, Y. Xiang, and M. Sun. Anticipating Accidents in Dashcam Videos. In S.-H. Lai, V. Lepetit, K. Nishino, and Y. Sato eds., *Computer Vision – ACCV 2016*, ACCV 2016, pp. 10114:136–153, Cham, 2017. Springer.
- [4] J. Cronje and A. P. Engelbrecht. Training Convolutional Neural Networks with Class Based Data Augmentation for Detecting Distracted Drivers. In *Proceedings of the 9th International Conference on Computer and Automation Engineering*, ICCAE '17, p. 126–130, New York, NY, USA, 2017. ACM.
- [5] S. E. F. de Avila, A. P. B. Lopes, A. da Luz, and A. de Albuquerque Araújo. VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognition Letters*, pp. 32(1):56–68, 2011.
- [6] J. Gemmell, G. Bell, R. Lueder, S. Drucker, and C. Wong. MyLifeBits: Fulfilling the Memex Vision. In *Proceedings of the Tenth ACM International Conference on Multimedia*, MULTIMEDIA '02, p. 235–238, New York, NY, USA, 2002. ACM.
- [7] Y. Gong and X. Liu. Video summarization using singular value decomposition. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 2000, pp. 2:174–180. IEEE, 2000.
- [8] Google. Google Maps Timeline. <https://www.google.com/maps/timeline>. (accessed 2021-09-18).
- [9] GoPro. HER10 Black. <https://gopro.com/en/us/shop/cameras>, 2021. (accessed 2021-09-18).
- [10] Grand View Research. Dashboard Camera Market Size, Share & Trends Analysis Report By Technology (Basic, Advanced, Smart), By Product, By Video Quality, By Application, By Distribution Channel, By Region, And Segment Forecasts, 2020 - 2027, 2020.
- [11] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur, and K. Wood. SenseCam: A Retrospective Memory Aid. In *Proceedings of the 8th International Conference on Ubiquitous Computing*, UbiComp '06, p. 177–193, Berlin, Heidelberg, 2006. Springer-Verlag.
- [12] Holoride. Holoride. <https://www.holoride.com>, 2019. (accessed 2021-09-18).
- [13] Honda. Honda Dream Drive. <https://www.honda.co.jp/CES/2017/detail/007/>, 2017. (accessed 2021-09-18).
- [14] Insta360. Insta360 Go 2. <https://www.insta360.com>, 2021. (accessed 2021-09-18).
- [15] A. Matsuda, T. Matsui, Y. Matsuda, H. Suwa, and K. Yasumoto. A Method for Detecting Street Parking Using Dashboard Camera Videos. *Sensors and Materials*, pp. 33(1):17–34, 2021.
- [16] Mordor Intelligence. DASHBOARD CAMERA MARKET - GROWTH, TRENDS, COVID-19 IMPACT, AND FORECASTS (2021 - 2026). <https://www.mordorintelligence.com/industry-reports/dashboard-camera-market>. (accessed 2021-09-18).
- [17] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang. Automatic video summarization by graph modeling. In *Proceedings Ninth IEEE International Conference on Computer Vision*, ICCV 2003, pp. 1:104–109. IEEE, 2003.
- [18] Y. Sumi, R. Sakamoto, K. Nakao, and K. Mase. ComicDiary: Representing Individual Experiences in a Comics Style. In *Proceedings of the 4th International Conference on Ubiquitous Computing*, UbiComp '02, p. 16–32, Berlin, Heidelberg, 2002. Springer-Verlag.
- [19] Y. Takimoto, Y. Tanaka, T. Kurashima, S. Yamamoto, M. Okawa, and H. Toda. Predicting Traffic Accidents with Event Recorder Data. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Prediction of Human Mobility*, PredictGIS '19, p. 11–14, New York, NY, USA, 2019. ACM.
- [20] 尾頭 花奈, 椎尾 一郎. ドライブレコーダを活用した思い出ムービー自動編集システム. 研究報告 ユビキタスコンピューティングシステム (UBI) , 2021-UBI-69(19), p. 1–8, 2021.
- [21] 石黒 祥生, 山田献二郎. 自動運転車両内インフォテインメントのための VR ゲーム. エンタテインメントコンピューティングシンポジウム 2017 論文集, pp. 210–216, 2017.