

# 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

尾頭 花奈<sup>1,a)</sup> 椎尾 一郎<sup>1</sup>

**概要:** 近年、ドライブレコーダが急速に普及している。しかし、ドライブレコーダを搭載する目的は、交通事故や危険運転車に遭遇した場合への備えである場合が多く、得られたデータが利用される機会は極めて少ない。高性能な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、ドライブ紀行ムービーの素材としても使用可能である。しかし、何時間にも及ぶ膨大なデータを手作業で編集するには、多くの時間と手間が必要である。本研究では、旅行時におけるドライブレコーダのデータを活用し、全自動で旅の思い出動画を作成するシステムを提案・実装した。本システムは名所、会話の盛り上がりなどを判定して動画を要約し、文字・地図・テロップ・合成音声の追加などを行う。

**キーワード:** ドライブレコーダ, 動画編集, 動画自動作成

## Computer Automated Edit of a Driving Movie from Dashcam Video Data

KANA BITO<sup>1,a)</sup> ITIRO SHIO<sup>1</sup>

**Abstract:** The use of dashcams has been rapidly increasing. They are mainly used to take measures against a traffic accident or encounter with a dangerous driver, and the recorded video data are usually not utilized but just deleted when the storage is full. Many of the recent dashcams feature HD video recording, which can be used to create travel video. Although the HD video data captured by a dashcam is easily obtained by getting the SD memory card, it takes a lot of time and effort to edit the huge amount of the data. In this study, we proposed and implemented a system that can automatically create a driving movie summarizing dashcam data recorded during a trip. This allows users to easily create a driving movie from the dashcam video.

**Keywords:** dashcam, dashboard camera, video editing, automatic video creation

### 1. はじめに

近年、ドライブレコーダは急速に普及している。国土交通省によると、年代・地域を問わずほぼ100%の人がドライブレコーダを認知しており、その搭載率も約46%である[1]。ドライブレコーダの主な用途は、交通事故やお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

んど活用されず、記憶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、360度カメラや複数台の高品位カメラを搭載したモデルも多く、車内外の映像を高画質で記録する。記録された動画データは、家族・友人と共有する思い出の振り返り動画、SNSや動画投稿サイトで共有・公開する旅動画の素材として十分な画質である。このような紀行ムービーをドライブレコーダで自動記録されたデータから作成することには、多くのメリットがある。まず、ムービー作成のために新たに撮影機材を用意する必要がなく、手動で撮影する手間も不要である。また、主体的

<sup>1</sup> お茶の水女子大学  
Ochanomizu University  
<sup>a)</sup> g1720534@is.ocha.ac.jp

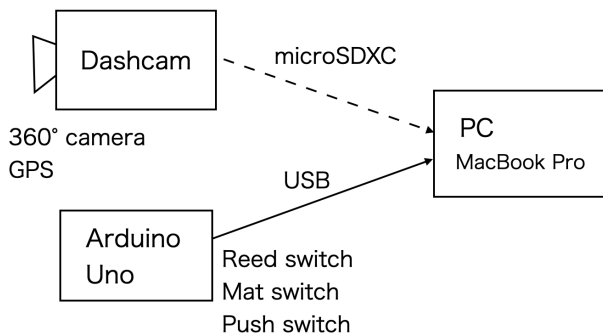


図 1 車内に設置するハードウェアの概略。

Fig. 1 Schematic of the hardware to be installed in the car.

に記録した画像よりも無意識的（自動的）に記録された画像の方が過去の記憶を想起しやすい [2] ことから、ドライブレコーダで自動撮影した動画から作成された紀行ムービーを閲覧することで、当時の思い出がより蘇りやすいと考えられる。加えて、車内の会話トピックの 50% 以上は何らかの形で車両周辺の場所と関連している [3] ことから、車両周辺の映像に車内会話を付加することで、場所に関連したナレーション付きのカットを生成できる可能性が高い。

ドライブレコーダの動画データは紀行ムービーの素材として有用であるものの、記録された長時間に渡るデータ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などの作業を手作業で行うことは困難である。GoPro アプリ<sup>\*1</sup>や Magisto<sup>\*2</sup>などのソフトウェアを使用することで、アクションカメラなどで撮影された動画の切り取り、結合、エフェクト追加などの自動編集が可能である。しかし、これらのソフトウェアを利用した場合も、作品に使用する動画の選択、文字テロップの入力などはユーザが手作業で行う必要がある。

そこで本研究では、ドライブレコーダにより記録した旅行中の車内外の動画データを全自動で編集する、思い出ムービー自動編集システムの提案および実装を行った。

## 2. システム概要

本システムは、ドライブ中の動画と状況を記録する車載システムと、取得したデータからドライブ紀行ムービーを自動作成する自動編集システムから構成される。これらを用いて、ユーザは以下のステップでドライブ旅行を記録したムービーを作成する。撮影はドライブレコーダが行い、本システムが自動編集するので、ユーザがすべき作業は少ない。

- (1) ドライブレコーダを起動しドライブする。必要に応じて、手動撮影を指示するスイッチを押す。
- (2) 帰宅後、ドライブレコーダに内蔵されている SD カードと、旅先で撮影した写真を、本システムが稼働する

<sup>\*1</sup> <https://gopro.com/ja/jp/shop/softwareandapp>

<sup>\*2</sup> <https://www.magisto.com>

PC に取り込む。

- (3) 本システムの自動編集機能を起動する。

現在の試作機段階では、ドライブ開始の際に PC の起動や Arduino からのログ取得開始などの操作が必要であるが、最終的には上記の 3 ステップで、編集作業を一切することなく思い出ムービーを作成することを目指している。

## 3. 車載システム

車内に設置するハードウェアを図 1 に示す。動画を撮影するためにドライブレコーダを設置し、ドア開閉検出、搭乗者検出、手動撮影指示の目的で、PC と Arduino を設置する。

### 3.1 ドライブレコーダ

本研究では、Yupiteru 社のドライブレコーダ Q-20P を使用した。この製品は、水平方向に 360 度/垂直方向に 240 度撮影可能で、GPS 測位機能を搭載する。取得した動画データと位置情報は、それぞれ MP4 ファイルおよび NMEA (National Marine Electronics Association) ファイルとして SD カードに記録される。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 ファイルの保存形式は魚眼と 2 分割（上半分が車外、下半分が車内の様子）から選択できるが、本研究ではデータの扱いやすさを考慮し 2 分割を選択した。また、ドライブレコーダはアクセサリ（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した。

### 3.2 車内センシング

ドライブレコーダに加えて、車内の状況を記録するためのセンサ、スイッチ類を設置する。このために、PC<sup>\*3</sup> と Arduino を搭載し、それぞれを USB 接続した。Arduino には、ドアの開閉、搭乗者の検出、手動撮影の指示のために、以下のセンサスイッチと押しボタンスイッチを接続した。

ドアの開閉を検出するためには、ドア部分の車体側にリードスイッチを設置し、ドア側に磁石を設置した。また、座席に人が座っていることを検出するためにマットスイッチ<sup>\*4</sup>を、車内/車外の記録を残したい場面であることをユーザが意思表示できるように、車内外用それぞれに押しボタンスイッチを設置した。また、ドライブ中のこれらのセンサ・スイッチ情報を PC 上のファイルに記録するソフトウェアを用意した。

<sup>\*3</sup> MacBook Pro 2.9 GHz デュアルコア Intel Core i5  
macOS Big Sur バージョン 11.1

<sup>\*4</sup> RS PRO, 圧力マット RS Pro 圧力パッドスイッチ (918-4756)

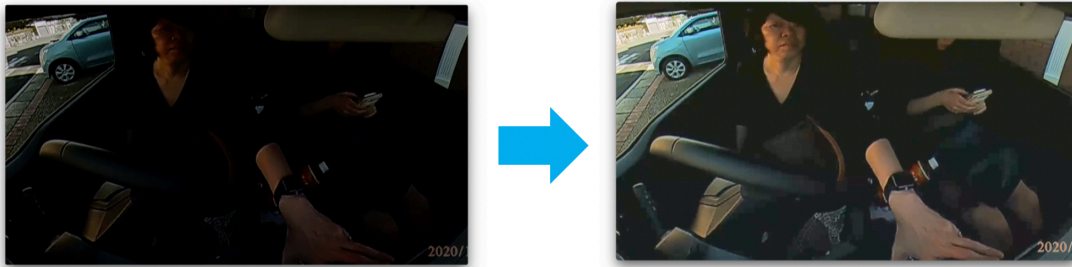


図 2 車内映像の明るさ調整. 補正前(左)と補正後(右).

Fig. 2 Brightness adjustment of the image inside the car. Before adjustment (left) and after adjustment (right).

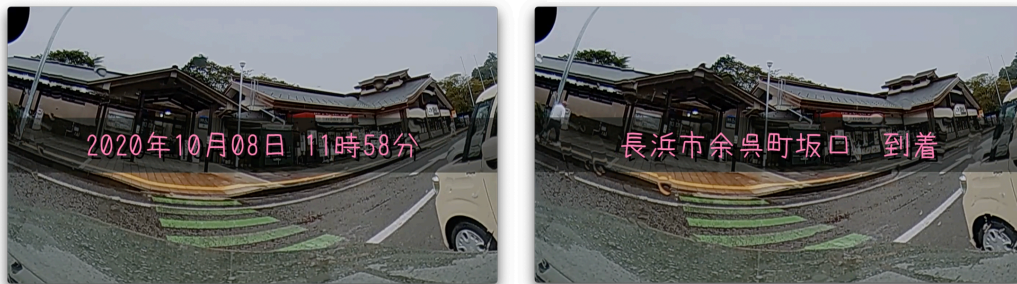


図 3 本システムで自動編集された到着時の映像. 前半 3 秒(左)と後半 3 秒(右).

Fig. 3 Video of arrival automatically edited by the proposed system. First 3 seconds (left) and second 3 seconds (right).

## 4. 自動編集システム

自動編集システムは、ドライブレコーダと Arduino により得られた動画ファイル、GPS・センサ・スイッチのログファイル、ユーザが用意した旅先での写真をもとに、ドライブ紀行ムービーを自動作成する。本システムは、Python3.7.9 によって記述され、コマンドラインツールである FFmpeg<sup>\*5</sup>を利用して動画処理を行なう。今回の試作では、車載システムで使用した PC と同じ PC で開発し動作させた。

本システムは以下の 5 つの処理を行う。

- (1) 動画のサイズと明るさの調整
- (2) ハイライトカットの自動判定と切り取り
- (3) 旅先で撮影した写真の追加
- (4) 地図の生成
- (5) 動画の連結とエフェクトの追加

以下、それぞれの処理の詳細を述べる。

### 4.1 動画のサイズと明るさの調整

ドライブレコーダで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。ここから、通常の HD ビデオと同じ 16 : 9 の縦横比の領域を切り出して使用する。実際には、撮影された画像をできる限り利

用する方針で、車外の映像を使用する時は上半分中央部の 1360 × 765 画素を、車内の映像を使用する時は下半分中央部の 1360 × 765 画素を切り出して使用した。ただ縦横画素数が奇数の場合、FFmpeg の処理に適さないため、用途に合わせてリサイズした。切り出しとリサイズ処理は、該当する映像をムービーで使用する場合にのみ行なった。例えば、ドライブレコーダの車外動画をフルサイズで表示する場面では、車外映像部分のみを切り出し、これを 1344 × 756 画素にリサイズした。なお、本システムが出力するムービーの画素数も、ドライブレコーダの映像をできる限り活用することを目的に、1344 × 756 画素とした。

ドライブレコーダは車外の撮影が目的であるため、窓から見える景色の光量に合わせた露出で撮影される。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、次に示す 3 つの処理を行った (図 2)。

- (1) ガンマ補正

$\gamma = 2.3$  とし画面全体を明るく調整した。

- (2) 彩度調整

ガンマ補正により低下した彩度を調整した。

- (3) コントラスト調整

ガンマ補正により低下したコントラストを調整した。

### 4.2 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

\*5 <https://ffmpeg.org>



図 4 本システムで自動編集された県境を通過したときの映像。  
Fig. 4 Video of passing the prefectural border automatically edited by the proposed system.



図 5 本システムで自動編集された名所付近を通過したときの映像。  
Fig. 5 Video of passing the place of interest automatically edited by the proposed system.

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 8 種類である。これらの事象は、ドライブレコーダが記録した 1 分録画ファイルの範囲で探索した。注目すべき事象が発生した場合、以下の手順に従って該当する車内/外の映像を切り取り、テロップ・合成音声などの処理を施したハイライトカットファイルとして保存した。

#### (1) 出発時

ACC 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 4 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を用いて、文字テロップと合成音声\*6の挿入を行った。

#### (2) 到着時

ACC 電源が OFF になったタイミングを到着時と判定し、そのときの車内の映像を 4 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情報、場所情報 API\*7を用いて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 3）。

#### (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



図 6 会話が盛り上がったタイミングを自動判定し、そのときの会話内容を字幕として表示する。

Fig. 6 The system automatically detects when the conversation is lively, and displays the conversation as subtitles.

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 5）。

#### (5) 会話が盛り上がっているとき

会話が盛り上がっているタイミングを音声データから判定し、そのときの車内の様子を 4 秒切り取った。MP4 ファイルから得られる音声データを、高速フーリエ変換し、人の音声周波数帯域とされる 300~3400Hz を切り取った。その後、逆高速フーリエ変換をして時間領域の信号に戻し、振幅が閾値以上の音声データを切り取った。また、SpeechRecognition ライブラリ\*8を利用して抽出した音声データの文字起こしを行い、字幕を表示した（図 6）。

#### (6) ドアが開いたとき

ドアに設置したリードスイッチによりドアの開閉を検出し、Arduino で検出時刻を記録した。そして、ドライブレコーダによる録画動画の該当箇所を 4 秒切り取った。

#### (7) 離着席時

座席に設置したマットスイッチにより離着席を検出し、Arduino で検出時刻を記録した。そして、ドライブレ

\*6 <https://pypi.org/project/gTTS/>

\*7 <https://developer.yahoo.co.jp/webapi/map/openlocalplatform/v1/placeinfo.html>

\*8 <https://pypi.org/project/SpeechRecognition/>



コードによる録画動画の該当箇所を4秒切り取った。

#### (8) 手動ボタン押下時

任意のタイミングで映像の切り取りを行うために、車内撮影用と車外撮影用の押しボタンスイッチを助手席に設置した。Arduinoでそれぞれのスイッチの押下時刻を記録する。ボタンが押されている間、ドライブレコーダによる録画動画の該当箇所を切り出した。

### 4.3 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った。画像の表示時間は1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。

### 4.4 地図の生成

車での旅行では、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分録画ファイル部分に対しては、地図を伴った表示を行うことにした。このために、foliumライブラリ<sup>\*9</sup>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(図7)。

この地図を車外映像と組み合わせるために、以下の2種類の表現方法を採用した。すなわち、車外の景色の変化が小さい場合には、車外映像を100倍速にして、透過した地図を重ね合わせた。また、外の景色の変化が大きい場合には、地図を主体に表示し、車外映像は1秒のコマ落として表示した。これを行うために、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(図7)、これを0.3秒程度表示する。車外映像領域は1秒のコマ落としてにした。

#### 差分が閾値未満だった場合

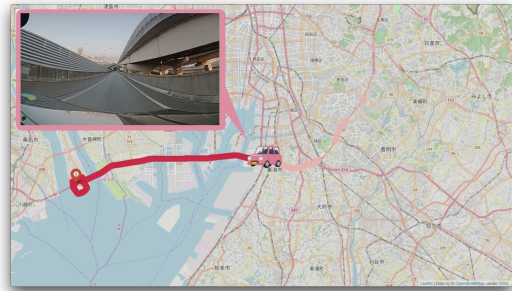


図7 地図上での可視化の様子。  
Fig. 7 Visualization on a map.



図8 100倍速にした動画に地図をアルファブレンディング。  
Fig. 8 Alpha-blending a map to a 100x speed video.

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8)。

### 4.5 動画の連結とエフェクトの追加

ここまでの手順により、データレコーダの1分録画ファイルのそれぞれに対して、ハイライトカットの切り出し、もしくは地図との合成が完了する。こららを時系列に繋げることで、紀行ムービーを作成することができる。しかし、時刻によっては、複数の注目事象が発生するために、一つの1分録画ファイルから複数種類のハイライトカットが用意される場合もある。また、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、画面が頻りに切り替わり見づらい動画になってしまう。そこで、ハイライトカットの優先順位を調整し、カットと地図画面を滑らかに連結させるために以下の処理を行なった。

#### 4.5.1 処理の優先順位

まず、1分録画ファイルの数に応じた大きさの1次元配列を用意し、それぞれの録画ファイルに対する処理を、以下のように数値で記録した。たとえば、1つ目の1分録画データから出発時のハイライト動画を作成した場合、配列の0番の中に、3を格納する。

- 1: この1分録画ファイルは使用しない
- 0: 景色の変化が少ないとき (以下、100倍速動画)
- 1: 景色の変化が多いとき (以下、地図主体表示)

<sup>\*9</sup> <https://python-visualization.github.io/folium/>

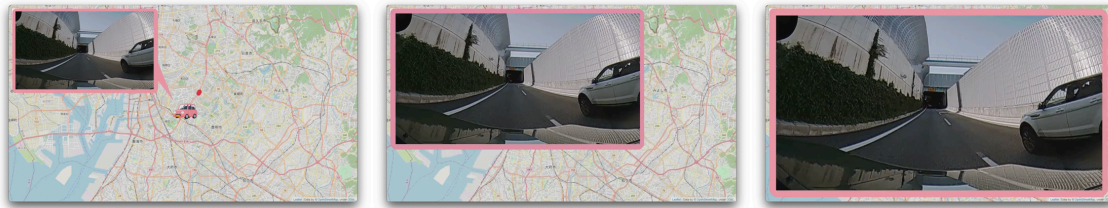


図 9 吹き出しのエフェクトによるトランジション (拡大).  
Fig. 9 Transitions with balloon effects (Zoom-in).



図 10 放射状に広がるワイプエフェクト.  
Fig. 10 Radiant wipe effect.

- 2: この時刻に旅先で撮影した写真が用意されたとき
- 3: 出発時 (ACC 電源が ON になったとき)
- 4: 到着時 (ACC 電源が OFF になったとき)
- 5: 県境を通過したとき
- 6: 名所付近を通過したとき
- 7: 会話が盛り上がっているとき
- 8: ドアが開いていたとき
- 9: 離着席したとき
- 10: 車内スイッチ作動
- 11: 車外スイッチ作動時

この数値は、前述の地図画面作成とハイライトカット切り出しを行う際に更新され、以前の処理記録は上書きされる。今回の実装では、最初に全ての1分録画ファイルに対して地図画面作成処理を行い（この段階で全てが0または1になる）、次にハイライトカット処理を行なった（一部が2から11に割り当てられる）。そのため、ハイライトカットがある場合は、地図作成処理の記録が上書きされる。また、特定の1分録画ファイルに対して複数種類のハイライト処理が実施される場合は、配列は基本的に後から行った処理番号に上書きされる。ただし、1つの1分録画ファイルから複数のハイライトカットが作成可能であるため、通常配列が上書きされてもハイライトカットは上書きされない。ハイライトカットの切り取り開始時刻が一致した場合に限り、以下の優先順位に基づいてハイライトカットが上書きされる。

優先順位:  $11 > 10 > 9 > 8 > 5 > 6 > 7 > 1 = 0$

また、2, 3, 4 の処理によって生成されたハイライトカットは他の処理で上書きされることはない。

#### 4.5.2 地図表示手法の調整

先に説明したように、ハイライトカットが無い場合、車

外の景色変化の大きさに合わせて、地図に合成する車外映像を、100倍速動画と1秒コマ落としに切り替えている。しかし過剰に切り替わると見辛いムービーになってしまう。そこで1分動画に対する処理を指定する配列に対して、以下のローパスフィルタ処理を行い変化を抑えた。処理の優先順位は (1) > (2) > (3) である。

(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。

例:  $[ \dots, 0, 1, 0, \dots ] \rightarrow [ \dots, 0, 0, 0, \dots ]$

(2) 100倍速動画が3連続未満なら、それら全て地図主体表示に変更する。

例:  $[ \dots, 1, 0, 0, 1, \dots ] \rightarrow [ \dots, 1, 1, 1, 1, \dots ]$

(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。

例:  $[ \dots, 0, 1, 5, \dots ] \rightarrow [ \dots, 0, -1, 5, \dots ]$

#### 4.5.3 トランジション効果

以上で決定された編集方針に従って、それぞれの1分動画ファイルから生成される素材を結合する。その際に必要に応じて以下の視覚音響効果を適用した。

##### 吹き出し

地図情報（配列に格納された数字が1の場合）と動画（数字が0または2以上の場合）の切り替えをスムーズ行うため、吹き出しのエフェクトを使用した。具体的には、地図情報から動画へ切り替える場合、地図部分にプロットされた車内の映像を表示した吹き出しを徐々に拡大することで切り替えを行う。反対に動画から地図情報へ切り替える場合は、フルサイズの車内の映像を地図情報にプロットされた吹き出しのサイズまで徐々に縮小することで切り替えを行なう（図9）。

##### フェードイン・フェードアウト

表 1 旅行時間と提案システムの実行時間の詳細.

Table 1 Details of travel time and proposed system execution time.

(出発地) - (到着地)	旅行時間	乗車時間	プログラム実行時間	動画再生時間	スイッチ類の使用
愛知-三重	8 時間 54 分 47 秒	約 2 時間 39 分	1 時間 18 分 28 秒	3 分 29 秒	不使用
愛知-石川	1 日 12 時間 23 分 28 秒	約 10 時間 38 分	4 時間 58 分 35 秒	12 分 49 秒	不使用
愛知-兵庫	1 日 11 時間 40 分 34 秒	約 14 時間 45 分	6 時間 46 分 53 秒	15 分 34 秒	不使用
愛知-三重	5 時間 18 分 53 秒	約 1 時間 38 分	41 分 31 秒	3 分 58 秒	使用

ACC 電源が OFF の間は記録されないため、目的地到着から再出発までには時間的な隔りがある。そこで、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。

### 放射状に広がるワイプエフェクト

通常の速度の動画から 100 倍速動画に切り替わるとき、再生速度が大きく変化する。そこで、これらの動画がスムーズに繋がるように放射状のワイプエフェクトを加えた (図 10)。

### ジングル

県境と名所付近を通過したときの映像、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、地図情報にはミディアムテンポのジングルをつけた。

#### 4.5.4 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. 評価

筆者の一人が母親と 2 名で実施した 4 回のドライブ旅行において、提案システムを使用し紀行ムービーを作成した (表 1)。車外および車内 (運転席・助手席) の様子を記録している。また、編集は帰宅後に行なった。ドアセンサ、座席センサ、手動録画スイッチは、4 回目の旅行の際に実装した。

この結果を表 1 にまとめた。センサ・スイッチを実装した 4 回目の旅行では、乗車時間に対して他の回よりも長い動画が作成された。センサ・スイッチの事象を除外した場合は 3 分程度の動画であったが、センサ・スイッチの事象を含んだ動画は 4 分弱であった。増加した部分は主にユーザにより手動録画された部分で、意味のある動画が得られていた。全体として、プログラムの所要実行時間は乗車時間の約 40~50 %であった。長時間ではあるが、その間の

ユーザの手間は不要であるため実用性はあると感じた。また乗車時間の約 0.02 ~ 0.04 %の動画が生成された。

自動生成された動画は旅行の内容を適切に要約しており、十分鑑賞に耐えるムービーになっていた。しかし、以下の点で課題がある。

- 高速道路など景色の変わらないところを長時間走ると冗長なものになりやすい。
- 観光名所の取得は有用であった。しかし、場所によっては不必要な地名も多く取得されてしまう。
- 夜間の映像はかなり暗いため、映像が不鮮明である。
- 強い雨や環境音が大きい場合、録音される音量も大きくなり、音声と誤判断され、ムービーに取り込まれる箇所が増えてしまう。
- 作成した紀行ムービーを SNS にアップロードする場合、プライバシーの観点から他人の顔や車のナンバープレートにモザイク処理を施す必要がある。

今後は、切り取り条件の追加や夜間の映像やナンバープレートなどに対する画像処理、周囲の環境音に応じた音声処理を行なう必要があると考える。また、乗車時間に画像処理プログラムの一部を実行し、リアルタイム解析・編集するよう設計すれば、帰宅後の処理時間を大幅に軽減できると考えている。

## 6. 関連研究

動画の自動要約に関する研究は多い。動画要約の手法として、特異値分解 (SVD) とクラスタリングによる映像要約技術 [4] や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 [5]、ビデオフレー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 [6] などがある。また、生活の記録から体験や思い出を要約する研究も多数されている。例えば、驚いたときに現われる反応である startle response を検出し、生活の中で利用者の注意を喚起した出来事を自動で記録する試み [7] や脳波とウェアラブルカメラを用いて、生活の記録動画を自動要約する試み [8]、ビデオカメラやマイク、赤外線 LED 付きの ID タグと赤外線信号追跡装置などを用いて、ユーザ体験のビデオサマリを作成する試み [9] がある。本研究では、ドライブレコーダ動画からの要約に絞って、適切で実用的な要約手法を検討した。

身近な製品を使用することで、生活シーンの切り取り

や要約を実現する研究も多い。ComicDiary[10]は携帯情報端末(PDA)やキオスク端末などを利用し、個人プロフィールや訪問記録、他のユーザとの交流から学術会議参加などの様子を漫画形式の日記として自動生成する。Video-Recording Your Life[11]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。マイクロソフトのプロジェクト「MyLifeBits」\*<sup>10</sup>[12]で使用されたウェアラブルカメラSenseCam\*<sup>11</sup>[13]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。おまかせ電子日記[14]は電子メール、PC上やポータブルプレイヤーで再生した音楽、検索キーワード、デジタルカメラで撮影した写真、GPSによる位置情報から情報を抽出し、それらをテンプレート文に当てはめることで日記を自動生成する。Kadiary[15]は観光地で撮影した写真とその写真から得られるExif情報、利用者が入力した写真に対するコメントなどを用いて、観光の記録をまとめた電子日記を自動作成する。本研究ではドライブレコーダの活用に着目し、車での旅行に関するビデオサマリの作成を目的とした。

## 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、旅のハイライトカットを自動検出して思い出動画を自動生成する思い出ムービー自動編集システムの提案と実装を行った。本システムを使用することで、車での旅行の思い出を簡単に紀行ムービーとしてまとめることができる。今後は、5章で紹介した本システムの課題の克服のほか、笑顔や笑声の検出および発話キーワードの検出を行い、切り取り箇所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。自動運転車には多数の高画質カメラ、センサ類が搭載されている。これらを利用して紀行ムービー作成システムを実現することで、現在のドライブレコーダを使った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] 国土交通省:自動車用の映像記録型ドライブレコーダー装置について, 入手先<<https://www.mlit.go.jp/monitor/R1-kadai01/24.pdf>>(参照 2021-01-19).
- [2] Abigail Sellen, Andrew Fogg, Mike Aitken, Steve Hodges, Carsten Rother, and Ken Wood: Do Life-Logging Technologies Support Memory for the Past? An Experimental Study Using Sensecam, *CHI '07: Proceedings of the SIGCHI Conference on Human Factors in*

- Computing Systems*, pp. 81–90 (2007).
- [3] Kohei Matsumura and Yasuyuki Sumi: What Are You Talking About While Driving?: An Analysis of In-car Conversations Aimed at Conversation Sharing, *AutomotiveUI '14: Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pp. 1–8 (2014).
- [4] Yihong Gong and Xin Liu: Video summarization using singular value decomposition, *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000*, Vol. 2, pp. 174-180 (2000).
- [5] Chong-Wah Ngo, Yu-Fei Ma, and Hong-Jiang Zhang: Automatic video summarization by graph modeling, *Proceedings Ninth IEEE International Conference on Computer Vision*, Vol. 1, pp. 104-109 (2003).
- [6] Sandra Eliza Fontes de Avila, Ana Paula Brandão Lopes, Antonio da Luz Jr, and Arnaldo de Albuquerque Araújo: VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method, *Pattern Recognition Letters*, Vol. 32, Issue 1, pp. 56–68 (2011).
- [7] Jennifer Healey and Rosalind W Picard: StartleCam: a cybernetic wearable camera, *Digest of Papers. Second International Symposium on Wearable Computers*, pp. 42–49 (1998).
- [8] Kiyoharu Aizawa, Ken-Ichiro Ishijima, and Makoto Shina: Summarizing wearable video, *Proceedings 2001 International Conference on Image Processing*, Vol. 3, pp. 398-401 (2001).
- [9] Yasuyuki Sumi, Sadanori Ito, Tetsuya Matsuguchi, Sidney S Fels, Shoichiro Iwasawa, Kenji Mase, Kiyoshi Kogure, and Norihiro Hagita: Collaborative capturing, interpreting, and sharing of experiences, *Personal and Ubiquitous Computing*, Vol. 11, pp. 265-271 (2007).
- [10] Yasuyuki Sumi, Ryuuki Sakamoto, Keiko Nakao, and Kenji Mase: ComicDiary: Representing Individual Experiences in a Comics Style, *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing*, pp. 16–32 (2002).
- [11] Daniel Buschek, Michael Spitzer, and Florian Alt: Video-Recording Your Life: User Perception and Experiences, *CHI EA '15: Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 2223–2228 (2015).
- [12] Jim Gemmell, Gordon Bell, Roger Lueder, Steven Drucker, and Curtis Wong: MyLifeBits: Fulfilling the Memex Vision, *MULTIMEDIA '02: Proceedings of the tenth ACM international conference on Multimedia*, pp. 235–238 (2002).
- [13] Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood: SenseCam: a Retrospective Memory Aid, *UbiComp'06: Proceedings of the 8th international conference on Ubiquitous Computing*, pp. 177-193 (2006).
- [14] 伊神 徹, 林 貴宏, 尾内 理紀夫: おまかせ電子日記: 行動履歴に基づいた電子日記自動生成システム, 第49回プログラミングシンポジウム予稿集, Vol. 2008, pp. 129-140 (2008).
- [15] 熊野 圭馬, 宮川 怜, 國枝 孝之, 山田 哲, 後藤田 中, 紀伊 雅敦, 八重樫 理人: 観光の振り返りを支援する観光日記生成/印刷システム「KaDiary/カダイアリー」の開発と運用, 情報処理学会研究報告, Vol. 2017-GN-100, No. 46, pp. 1-6 (2017).

\*<sup>10</sup> <https://www.microsoft.com/en-us/research/project/mylifebits/>

\*<sup>11</sup> <https://www.microsoft.com/en-us/research/project/sensecam/>