

眺望の良さ推定に基づく眺望ベースルート推薦システム

川俣 光司[†] 奥 健太[†]

[†] 龍谷大学理工学部 〒520-2194 滋賀県大津市瀬田大江町横谷 1-5

E-mail: †t18m057@mail.ryukoku.ac.jp, ††okukenta@rins.ryukoku.ac.jp

あらまし ドライブの楽しみ方の一つとして、海沿い景観や田園景観など好きな景観を眺めながら走りたいという要求もある。このような要求に対し、著者らはこれまでに四つの景観要素（田園系, 山林系, 水辺系, 都市系）に着目した景観ベースルート推薦システムの研究に取り組んできた。ただし、このシステムでは道路の近景のみを景観要素として用いており、道路からの眺望の良さを考慮できていないという課題がある。本研究では、眺望の良さを導入することで、これまでの景観ベースルート推薦システムを拡張する。ここで、眺望の良さは道路リンクごとに推定する。推定に用いる道路リンクの特徴として、OSM の道路属性に加え、独自に定義した道路の景観ベクトル、可視直線距離、周囲との高低差、可視領域率を用いる。これらの推定結果を踏まえ、本研究で提案するルート推薦システムでは、出発地と目的地が与えられたときに、眺望の良いルートを優先したルートを推薦する。評価実験においては、最短ルートと比較することで眺望の良いルートを推薦することの有効性を検証する。

キーワード ルート推薦, 景観特徴, 眺望の良さ推定

1. はじめに

自動車は単なる移動手段だけでなく、ドライブすること自体が娯楽の一つとなっている。ドライブの楽しみ方の一つとして、海沿い景観や田園景観など好きな景観を眺めながら走りたいという要求もある。このような要求においては、必ずしも最短ルートなど効率性を重視したルートを提示することが最適であるとは限らず、多少迂回してでもユーザの要求に合うルートを提示することが求められる。

ルート推薦システムは、出発地と目的地を指定すると、その間のルートを提示する。既存のルート推薦システムは、最短ルート [1] [2] や最速ルート [3] [4] [5] [6] [7], 人気ルート [8] [9] [10] [11] を推薦するものが多い。ここで、先述のように、最短ルートや最速ルートは必ずしも先の要求を満たすとは限らない。人気ルート推薦は、多くの人々が関心をもっているルートを推薦する手法である。Wei ら [10] は、ユーザの軌跡データから多くのユーザが関心をもつ道路リンクをマイニングすることで、人気ルートの抽出を行っている。このような推薦システムは集合知に基づき道路リンクの魅力を定義しているが、その道路リンクがどのような特徴を有するか内容に踏み込んだ解析はなされていない。

我々は景観を重視したルートを推薦する景観ベースルート推薦システムの実現を目指している。例えば、海沿い景観が好きなユーザには海沿い景観を優先したドライブルートを、田園景観が好きなユーザには、田園景観を優先したドライブルートを推薦する。このようなシステムの実現には景観を重視したルート探索手法を開発する必要がある。

我々はこれまでに景観ベクトル付き道路ネットワーク [12] を前提とした景観ベースルート推薦システムを提案してきた [13]。この研究では、ルート探索時間とユーザ評価の観点から提案システムの有用性を示した。しかし、先行研究でのシステムで

は、景観特徴として田園景観、山林景観、水辺景観、都市景観といった景観種別に着目しており、景観の良さそのものは考慮していない。しかしながら、目指すべき景観ベースルート推薦システムとしては、景観種別に加え、その景観の良さも考慮に入れる必要がある。

本稿では、景観の良さに関連する一つの要素として眺望の良さを導入することで、これまでの景観ベースルート推薦システムを拡張する。ここで、眺望の良さは道路リンクごとに推定する。推定に用いる道路リンクの特徴として、OpenStreetMap (OSM) の道路属性に加え、独自に定義した道路の景観ベクトル、可視直線距離、周囲との高低差、可視領域率を用いる。これらの推定結果を踏まえ、本研究で提案するルート推薦システムでは、出発地と目的地が与えられたときに、眺望の良さを優先したルートを推薦する。評価実験においては、最短ルートと比較することで眺望の良いルートを推薦することの有効性を検証する。

2. 関連研究

最短経路探索アルゴリズムとして、ダイクストラ法 [14] や A* アルゴリズム [15] が挙げられる。これらのアルゴリズムでは、入力された始点と終点において、道路ネットワーク上のリンクに付与されたコストに基づき、総コストが最小となるようなルートを選択する。

最速経路探索 [5] は、リンクの距離ではなく、リンクを通過するための旅行時間に着目し、旅行時間が最小となるようなルートを選択する。Wei ら [5] は、GPS の軌跡データから速度パターンをマイニングすることで旅行時間を推定している。

人気ルート推薦は、多くの人々が関心をもっているルートを推薦する手法である。Wei ら [10] は、ユーザの軌跡データから多くのユーザが関心をもつルートをマイニングすることで、人気ルートの抽出を行っている。

個人の嗜好に応じたルートを推薦する、個人化ルート推薦の研究もある。MyRoute [16] は、ユーザが熟知しているルートやランドマークに基づき、ユーザ個人向けのルートを作成している。MyRoute では、ランドマークをユーザ自身で入力する必要があるのに対し、Going My Way [17] では、ユーザの個人的な GPS ログデータから、ランドマークを自動的に特定している。

景観を考慮したドライブルート推薦として、Zhang ら [18] の研究がある。この研究では、景観スポットを「富士山、エッフェル塔、天安門などの遠景として楽しめる景観」と定義している。まず、Web から景観スポットを取得する。Digital Elevation Map (DEM) を基に構築した仮想 3D 空間において、Z-Buffer という考え方をを用いて、各道路ノードから、そのスポットへの可視率を算出する。出発地から目的地に向かうルート候補を、その可視率スコアが高い順にランキングする。彼らの手法では、景観スポットを考慮しているが、道路リンクの景観特徴は用いられていない。

3. 定義

本稿において用いる記号を以下のとおり定義する：

$G = (V, E)$: 道路ネットワーク。有向重み付きグラフで表現される。

$v_i \in V$: 道路ノード。交差点や道路の終端を表す。

$e_i \in E \subseteq V \times V$: 道路リンク。道路リンク $e_i = (v_j, v_k) \in E$ は、始点ノード v_j から終点ノード v_k へ向かう有向リンクである。

$L(e_i)$: 道路リンク e_i の距離。

$\theta(e_i)$: 道路リンク e_i の方向角。

w_i : 道路リンク e_i のコスト。

$\mathbf{r}(e_i) = (r_{i1}, r_{i2}, \dots, r_{i9})$: 道路リンク e_i の道路属性ベクトル。該当する属性のみに 1、それ以外の属性には 0 が設定される。

$\mathbf{s}(e_i) = (s_i^r, s_i^m, s_i^w, s_i^u)$: 道路リンク e_i の景観ベクトル。ここで、 $\sum_k s_i^k = 1$ 。

$l(e_i)$: 道路リンク e_i の可視直線距離。

$\mathbf{d}(e_i) = (d_i^{\max}, d_i^{\min})$: 道路リンク e_i の周囲との高低差ベクトル。

$a(e_i)$: 道路リンク e_i の可視領域率。

\mathbf{x}_i : 道路リンク e_i の特徴ベクトル。

$y_i \in \{0, 1\}$: 道路リンク e_i の眺望ラベル。

$\hat{y}_i \in \{0, 1\}$: 道路リンク e_i の予測眺望ラベル。

4. 道路ネットワークの眺望推定

与えられた道路ネットワーク $G = (V, E)$ 上の各道路リンク $e_i \in E$ の眺望を推定する。眺望の推定にはクラス分類器を用いる。クラス分類器を本問題に適用するためには、道路リンクの特徴量および教師ラベル（眺望の良さ）が必要になる。4.1 節で道路リンクの特徴化、4.2 節で教師ラベルの付与について述べる。4.3 節で眺望に基づく道路ネットワーク上の道路リンクへのコスト付与について述べる。

4.1 道路リンクの特徴化

道路からの眺望推定に用いる道路リンクの特徴量として、OSM の道路属性に加え、独自に定義した道路の景観ベクトル、可視直線距離、周囲との高低差、可視領域率を用いる。以下、各特徴量について説明する。

4.1.1 道路属性

OSM 道路属性とは、OpenStreetMap^(注1) の道路ネットワークに含まれる道路リンクの道路種別である。道路種別は type 属性として与えられている。本研究では、type 属性が { 'secondary', 'primary', 'trunk', 'motorway', 'motorway_link', 'trunk_link', 'primary_link', 'secondary_link', 'bridge' } の九つの道路属性を道路リンクの特徴として用いる。

道路リンク e_i の道路属性を 9 次元ベクトルとして $\mathbf{r}(e_i) = (r_{i1}, r_{i2}, \dots, r_{i9})$ と定義する。ここで、 $\mathbf{r}(e_i)$ の各要素は先の九つの道路属性に対応する。該当する要素のみが 1 となり、それ以外の要素は 0 となる。例えば、道路属性が 'secondary' である場合、 $\mathbf{r}(e_i) = (1, 0, 0, 0, 0, 0, 0, 0, 0)$ となる。

4.1.2 景観ベクトル

我々は、先行研究 [12] において道路リンクの景観ベクトルを定義した。まず、景観を構成する基本的な要素として、四つの景観要素 (rural, mountainous, waterside, urban) を定義している。景観ベクトルは、各景観要素に対応した 4 次元の確率ベクトルとして定義される。道路リンク e_i の景観ベクトルを $\mathbf{s}(e_i)$ と定義し、 $\mathbf{s}(e_i) = (s_i^r, s_i^m, s_i^w, s_i^u)$ と表す。ベクトルの各要素は、対応する景観要素が e_i に含まれる確率を表す。したがって、全要素の総和は 1 となる。道路リンクの景観ベクトル化については文献 [12] を参照されたい。

4.1.3 可視直線距離

曲がりくねった道路よりも、直線状に伸びた道路の方が眺望が良いことは経験的にも明らかである。そこで、道路リンクの眺望に寄与する要素の一つとして道路リンクの可視直線距離を導入する。本研究では、ドライバが鮮明に見える角度内にある連続した道路リンクの距離を可視直線距離として定義する。ここで、ドライバが鮮明に見える角度は、文献 [19] を参考に、まっすぐ見て -2.5° から 2.5° とする。道路リンク e_i の可視直線距離を $l(e_i)$ と定義する。

道路リンク e_i の可視直線距離 $l(e_i)$ は、道路リンクの方向角 $\theta(e_i)$ を基に算出する。ここで、道路リンクの方向角 $\theta(e_i) (-90 < \theta(e_i) \leq 90)$ は、真北を 0° 、時計回りを正とした度数法による角度とする。可視直線距離に基づく道路リンクの特徴化の疑似コードをアルゴリズム 1 に示す。

以下、アルゴリズム 1 に沿って可視直線距離に基づく道路リンクの特徴化を説明する：

(1) 各道路リンク $e_i \in E$ の可視直線距離 $l(e_i)$ を算出する (lines 1–4)。まず、道路リンク e_i の可視直線距離 $l(e_i)$ を道路リンク e_i 自体の距離 $L(e_i)$ で初期化する (line 2)。つづいて、 $\text{linearity}(e_i, l(e_i))$ 関数を呼び出すことで、可視直線距離を算

(注1) : <http://download.geofabrik.de/>

Algorithm 1 可視直線距離に基づく道路リンクの特徴化

```

1: for each link  $e_i \in E$ 
2:    $l(e_i) \leftarrow L(e_i)$ 
3:    $l(e_i) \leftarrow \text{linearity}(e_i, l(e_i))$ 
4: end for
5: function LINEARITY( $e, l$ )
6:   adjLinkList  $\leftarrow$  getAdjLinks( $e$ ): Get links adjacent to  $e$ .
7:   for each link  $e_j$  in adjLinkList
8:     if  $|\theta - \theta(e_j)| \leq 2.5$  or  $180 - |\theta - \theta(e_j)| \leq 2.5$  then
9:        $l \leftarrow l + \text{linearity}(e_j, l)$ 
10:    end if
11:  end for
12:  return  $l$ 
13: end function

```

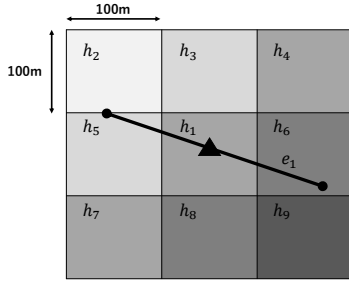


図1 周囲との高低差算出に用いるメッシュの例である。 h_i は、 i 番目のメッシュの標高値である。メッシュの標高値 h_i は濃淡で表しており、濃度が低いほど標高値は高くなっている。 e_1 の三角は、 e_1 の中点を説明している。

出する (line 3)。

(2) **linearity()** 関数は、引数に渡された道路リンク e_i の可視直線距離 $l(e_i)$ を算出し、それを返す。まず、getAdjLinks(e) 関数を呼び出し、道路リンク e と隣接している道路リンク集合を取得し、それをリストとして **adjLinkList** に格納する (line 6)。ここで、道路リンク e_i と e_j が一つの同一のノードに接続しているとき、 e_i と e_j は隣接しているものとする。

(3) **adjLinkList** 内の各道路リンク e_j について考える (lines 7–11)。ここで、先述したとおり、二つの道路リンクの方向角の差が 2.5° 以内のとき、これらの道路リンクは直線的であるとすると。したがって、 $|d - d(e_j)| \leq 2.5$ または $180 - |d - d(e_j)| \leq 2.5$ のとき、**linearity**(e) が再帰的に呼び出される (lines 8–9)。

以上の繰返し処理が終了すると、道路リンク e_i について、直線的な道路リンクすべての総距離が $l(e_i)$ に代入されることになる。上記の処理は、道路ネットワーク内の全てのリンクが可視直線距離に基づいた道路リンクの特徴化が終了するまで、繰り返される。

4.1.4 周囲との高低差

周囲の標高よりも相対的に高い地点の方が眺望が良いことも経験的に明らかである。そこで、周囲との高低差を道路リンクの特徴として導入する。道路リンク e_i の周囲との高低差を2次元ベクトルとして $\mathbf{d}(e_i) = (d_i^{\max}, d_i^{\min})$ と定義する。

まず、対象地域を 100m 四方のメッシュに分割する。このうち、道路リンク e_i の中点を含むメッシュ (メッシュ番号 1 とする) と、その周囲 8 方向に隣接するメッシュ (メッシュ番号 2–9 とする)、計 9 個のメッシュを考える。各メッシュに対応する標高を $\{h_1, h_2, \dots, h_9\}$ とする。ここで、標高データ (DEM)^(注2) は、国土交通省により国土数値情報として公開されている。ただし、国土数値情報として公開されている標高は 250m メッシュに対応しているため、100m メッシュの標高に対応させる。

道路リンク e_i の周囲との高低差ベクトル $\mathbf{d}(e_i) = (d_i^{\max}, d_i^{\min})$ は次式により求める：

$$d_i^{\max} = \max_k h_k - h_1 \quad (1)$$

$$d_i^{\min} = \min_k h_k - h_1 \quad (2)$$

ここで、 h_1 は道路リンク e_i を含むメッシュである。つまり、 d_i^{\max} は周囲の最高標高と道路リンク e_i との高低差であり、 d_i^{\min} は周囲の最低標高と道路リンク e_i との高低差である。ここでは、正負の符号は特徴量として無関係であるため、それぞれ絶対値をとっている。

図1より、周囲との高低差算出に用いるメッシュの例を示す。 h_i は、 i 番目のメッシュの標高値である。メッシュの標高値 h_i は濃淡で表しており、濃度が低いほど標高値は高くなっている。 e_1 の三角は、 e_1 の中点を説明する。図1の場合、 d_i^{\max} は h_1 と h_2 の高低差、 d_i^{\min} は h_1 と h_9 の高低差となる。

4.1.5 可視領域率

周辺に視界を遮るような建物等の有無が眺望の良さに影響を及ぼすと考えられる。そこで、周辺に遮蔽物が存在しない範囲を表す特徴として可視領域率を導入する。可視領域率は、道路リンクから中景または遠景を眺める場合に、近傍建造物に視界を遮られない領域の割合と定義する。道路リンク e_i の可視領域率を $a(e_i)$ と定義する。

道路リンク e_i の可視領域率 $a(e_i)$ は、道路リンクの近傍構造物を基に算出する。ここで用いる建造物データ^(注3)は、国土地理院により基盤地図情報として公開されている。建造物データは、ポリゴンで表現されたものである。近傍建造物を用いた可視領域率に基づく道路リンクの特徴化手順を下記に示す。

(1) $a(e_i) = 0$ とする。

(2) 道路リンク e_i の中心から半径 100m 以内に存在する建造物を取得する。

(3) 着目する方向角を θ (真北を 0° 、時計回りを正とした度数法による角度) とし、初期値 $\theta = 0^\circ$ を設定する。

(4) 道路リンク e_i の中心から方向角 θ の直線上に、(1) で取得した建造物が一つも存在しない場合は $c \leftarrow c + 1$ とする。

(5) 方向角を $\theta \leftarrow \theta + 5^\circ$ とする。このとき、 $\theta \geq 360^\circ$ となれば終了する。そうでなければ、(4) の処理を繰り返す。得られた c に対して 0 から 1 の範囲で正規化を行い、その結果を可視領域率 $a(e_i)$ とする。

最終的に、道路リンク e_i の可視領域率 $a(e_i)$ ($0 \leq a(e_i) \leq 1$) は

(注2) : <http://nlftp+.mlit.go.jp/ksj/gml/datalist/KsjTmplt-G04-d.html>

(注3) : https://fgd.gsi.go.jp/download/ref_kihon.html

次式により求める：

$$a(e_i) = \frac{c}{72} \quad (3)$$

$a_i = 0$ のとき、道路リンク e_i の近隣が建造物に囲われており、中景と遠景を見ることのできない道路リンク e_i となる。一方で、 $a_i = 1$ のとき、近隣に建造物が存在しないことを表し、360° 開けた景観であることを意味する。以上の処理をすべての道路リンクについて繰り返す。

4.1.6 特徴量の結合

以上で定義した特徴量を結合し一つのベクトルとして作成したものを道路リンクの特徴ベクトルとする。すなわち、道路リンク e_i の特徴ベクトルは、 $\mathbf{x}_i = (\mathbf{r}(e_i), \mathbf{s}(e_i), l(e_i), \mathbf{d}(e_i), a(e_i))$ となる。したがって、道路リンク特徴ベクトル \mathbf{x}_i は、17 次元となる。なお、作成したベクトルについて要素ごとに $[0, 1]$ の範囲になるように正規化しておく。

4.2 教師ラベルの付与

本研究では、クラウドソーシングにより、道路リンクの眺望の良さに関する教師ラベルを付与する。道路ネットワークから任意の道路リンクを抽出し、その地点のストリートビュー画像をワーカーに提示する。ワーカーには、眺望の良さについて、リッカート尺度 {5. 非常に良い, 4. 良い, 3. どちらでもない, 2. 悪い, 1. 非常に悪い} を用いて 5 段階で評価してもらう。評価にあたり、「眺望の良さ」は海や山、街などを広範に一望できることと定義する。道路リンク 1 件に対し N 名のワーカーに眺望の良さを評価してもらう。

N 名のワーカーによる平均評価値をその道路リンクの眺望の良さとする。ただし、2 クラス分類学習問題への適用を想定し、道路リンク e_i の教師ラベルを $y_i \in \{0, 1\}$ で表す。道路リンク e_i に対する平均評価値が 4 以上のとき、「眺望が良い」（正事例）とし $y_i = 1$ とする。平均評価値が 2.5 以下のとき、「眺望が悪い」（負事例）とし $y_i = 0$ とする。

4.3 道路ネットワークグラフへの眺望コスト付与

2 クラス分類器を用いて、与えられた学習データセット $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots\}$ から学習モデルを構築する。構築した学習モデルを基に、未知の道路リンク e_i の眺望 $\hat{y}_i \in \{0, 1\}$ を予測する。

予測された眺望 \hat{y}_i に基づき、道路リンク e_i に対してコストを付与する。ルート探索において眺望が良いルートが選ばれやすくなるため、 $\hat{y}_i = 1$ となる道路リンクのコストが小さくなるように設定する。

道路リンク e_i のコスト ω_i は次式で算出する：

$$\omega_i = L(e_i)(1 - \alpha \cdot \hat{y}_i) \quad (4)$$

ここで、 $L(e_i)$ は道路リンク e_i の距離である。 $\alpha < 1$ はパラメタである。

5. 眺望ベースルート推薦

眺望ベースルート推薦システムでは、道路リンクからの眺望の良いルートを推薦する。ユーザはマップビューにマーカーを置くことで出発地および目的地を入力する。眺望の良いルート

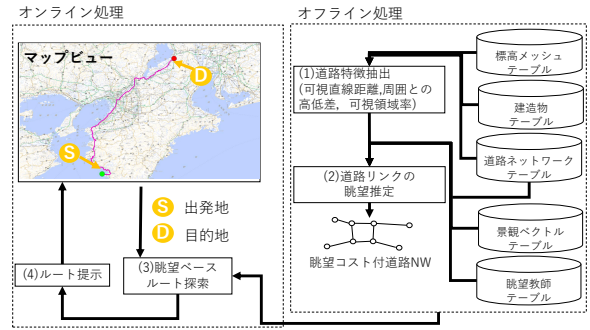


図2 システム構成図

は、マップビューに提示される。図2にシステム構成図を示す。以下、各処理について説明する。項目番号は図中の番号に対応する：

- (1) 道路ネットワーク内の各道路リンクを特徴化する。
- (2) 眺望コスト付きの道路ネットワークを構築する。
- (3) 眺望コスト付きの道路ネットワークから眺望ルートを探索する。

- (4) 眺望ルートをマップビューに提示する。

ここで、(1) および (2) の処理については、4. 章で説明した内容であり、入力に依存しないためオフラインで事前処理される。(3) では、与えられた眺望コスト付き道路ネットワークにおいて、入力要求（出発地、目的地）を満たす眺望ルートを探索する。具体的には、眺望コスト付き道路ネットワークにおいて、Dijkstra 法～[14] などの既存の最短経路アルゴリズムを用いて、リンクのコストの合計を最小化するルートを見つける。(4) では、探索されたルートをマップビューに提示する。

6. 評価

6.1 準備

本実験で用いるデータとして、OpenStreetMap の道路ネットワークデータから、関西圏の主要道路のデータを用いる。本データセットには、1,340,011 件の道路リンクが含まれる。このうち、正事例と負事例の比率が 1:1 となるように、ランダムに 460 件の道路リンクを学習データとして抽出した。本実験では、クラス分類手法として、Support Vector Machine (SVM) を用いた。SVM は、少ない学習データからでも高精度な結果が得られるという利点がある。

まず、先の 460 件の学習データに対し、5 分割交差検定法を実施した。その結果、平均予測精度は 75.8% であった。

つづいて、同 460 件の学習データを基に、SVM の学習モデルを構築し、1,340,011 件の道路リンクの眺望を予測した。結果、 $y_i = 1$ と予測された道路リンクは 527,212 件、 $y_i = 0$ と予測された道路リンクは 812,799 件であった。

式 (4) により、道路リンクの眺望コストを算出した。ここで、 $\alpha = 0.8$ とした。算出したコストに基づき、眺望コスト付き道路ネットワークを構築した。

6.2 景観要素ごとの眺望の良さ

本節では、景観要素（田園景観、山林景観、水辺景観、都市

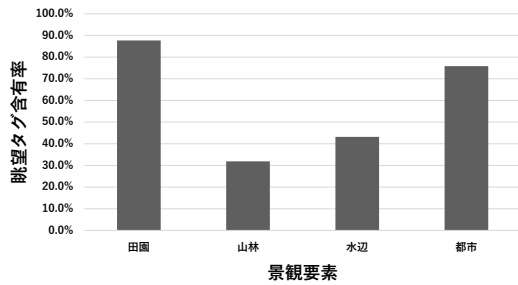


図3 景観要素ごとに、道路リンクの全体距離を100%としたとき、眺望が良い道路リンクの距離の割合を示したものである。各景観ベクトル要素がその景観の平均要素値以上の道路リンクをその景観要素の道路リンクとみなしている。

景観) ごとに眺望の良さについて考察する。図3は、景観要素ごとに、道路リンクの全体距離を100%としたとき、眺望が良い道路リンクの距離の割合を示したものである。ここでは、各景観ベクトル要素がその景観の平均要素値以上の道路リンクをその景観要素の道路リンクとみなしている。

図3より、田園景観、都市景観においては、眺望が良い道路リンクが多く含まれていることがわかる。そのため、田園景観と都市景観は、眺望が良い道路リンクが多いことがわかる。一方で、山林景観、水辺景観において、眺望が良い道路リンクが少ないことがわかる。

以下、各景観について定性的に分析する。ここでは、実際の道路のストリートビュー画像(図4)を用いて分析を行う。

田園景観。 図4Aに田園景観の道路のストリートビューを示す。図4(Aa)は、周辺に遮蔽物がなく、眺望が良い。そのため、眺望が良い道路と推定されたと考える。図4(Ab)は、左側に建造物があり眺望が悪い。可視領域率が小さいため、眺望が悪い道路と推定されたと考える。

山林景観。 図4Bに山林景観の道路のストリートビューを示す。図4(Ba)は、前方の山を見渡すことができ、眺望が良い。直線距離が長い、眺望が良い道路と推定されたと考える。図4(Bb)は、周辺に木々が生い茂っており、眺望が悪い。直線距離が短く、周囲の山林景観要素が多いため、眺望が悪い道路と推定されたと考える。

水辺景観。 図4Cに水辺景観の道路のストリートビューを示す。図4(Ca)は、周辺に遮蔽物がなく、眺望が良い。そのため、眺望が良い道路と推定されたと考える。図4(Cb)は、近郊に海はあるが海との間に森林があるため、眺望が悪い。水辺景観要素が多い、しかし、山林景観要素も多い。そのため、眺望が悪いと推定されたと考える。

都市景観。 図4Dに都市景観の道路のストリートビューを示す。図4(Da)は、見通しが良く、眺望が良い。直線距離が長く、道路幅が広い、眺望が良い道路と推定されたと考える。図4(Db)は、道路幅が狭く、両側に建造物があるため、眺望が悪い。道幅が狭く、両側に建造物があるため、眺望が悪いと推定されたと考える。都市景観道路での眺望推定では、道路幅が大きく関わっていると考え。しかし、特徴量として道路幅はな

A.田園



(Aa)眺望が良い



(Ab)眺望が悪い

B. 山林



(Ba)眺望が良い



(Bb)眺望が悪い

C.水辺



(Ca)眺望が良い



(Cb)眺望が悪い

D.都市



(Da)眺望が良い



(Db)眺望が悪い

図4 眺望推定結果ごとの各景観要素の強い道路のストリートビュー画像。Aは田園景観、Bは山林景観、Cは水辺景観、Dは都市景観となっている。aは眺望が良いと推定された道路、bは眺望が悪いと推定された道路のストリートビュー画像である。写真の出典: Google ストリートビュー画像をキャプチャしたものを掲載。

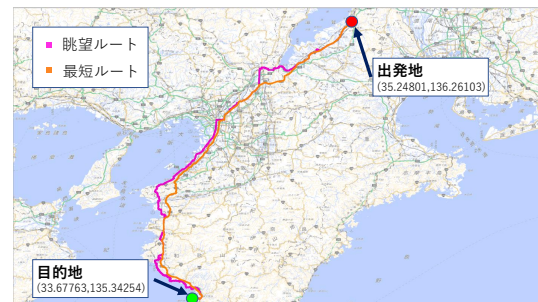


図5 出発地点を北緯 35.24801°, 東経 136.26103°, 目的地点を北緯 33.67763°, 東経 135.34254° としたときの推薦ルート。桃線は眺望の良さを重視したルートである。地図画像の出典: 地理院タイル(淡色地図)をシステム上に提示し、キャプチャしたものを掲載。

い。これは、可視領域率から道路幅が推定できたため、眺望の良さが推定できたと考える。

以上のとおり、各景観要素において、眺望の良い道路リンクに対して、眺望が良いと推定されていることがわかる。

6.3 眺望ベースルート推薦の妥当性

眺望ベースルート推薦システムは、与えられた出発地点および目的地点において、これら2地点を結ぶ眺望ルートを推薦する。ここでは、提示されたルートの妥当性を定性的に評価する。定性評価として、眺望ルートと最短ルートの比較を行う。図

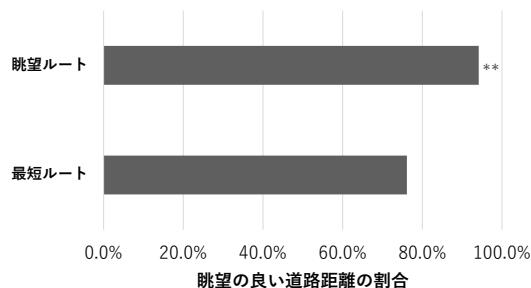


図 6 各推薦ルートに含まれる眺望タグの割合。各ルートの全長を 100% としたとき、それぞれの眺望タグが付与されている道路リンクの距離の割合を表している。* および ** は最短ルートと比較したとき、対応のある t 検定 (片側検定) による有意差 ($p < 0.05, p < 0.01$) が確認できたことを表す。

5 は、出発地点を北緯 35.24801°, 東経 136.26103°, 目的地点を北緯 33.67763°, 東経 135.34254° としたときの推薦ルートである。桃線は眺望ルート、橙線は最短ルートである。

図 5 から、眺望ルートは、最短ルートと比べ、海岸線に沿ったルートが提示されている。また、天津市付近のルートに着目すると湖岸沿いのルートとなっている。さらに、山林地帯の通過を避け、田園地帯を多く通過するルートが選ばれている。このように、最短ルートに比べ、遮蔽物の少ない地域を通過するようなルートとなっており、眺望の良さを考慮したルートが推薦されていることがわかる。

次に各ルートに含まれる道路リンクの傾向について定量的に評価する。ここで、実験に用いる出発地と目的地の組合せとして次の 5 組を用意する。

- (1) 彦根 (35.24801, 136.26103) → 白浜 (33.67763, 135.34254)
- (2) 大阪 (34.70099, 135.49651) → 伊勢 (34.45917, 136.72246)
- (3) 神戸 (34.69209, 135.19557) → 城崎 (35.61992, 134.81209)
- (4) 京都 (34.98910, 135.75980) → 吉野 (34.37601, 135.85409)
- (5) 大津 (34.98128, 135.92084) → 淡路 (34.25374, 134.68011)

図 6 は、推薦ルートの全長を 100% としたとき、そのルート中に眺望の良い道路リンクが含まれる割合を示している。ここでは、上記 5 組の推薦について、その平均値を示している。また、図 6 中の * および ** は最短ルートと比較したとき、対応のある t 検定 (片側検定) による有意差 ($p < 0.05, p < 0.01$) が確認できたことを表す。

以上のとおり、眺望ベースルート推薦システムでは、眺望の良さを重視したルートが推薦されることを定性的かつ定量的に確認した。

7. ユーザ評価

提案システムの有効性をユーザの観点から定量的に評価する。6.3 で用いた 5 パターンの出発地と目的地の対において、2 種類のルート—最短経路 (ベースライン)、眺望の良いルート—を評価する。このユーザ評価手順は、[20] に記載されている評価手順を参考にしている。

図 7 に示すように、各パターンにおいて被験者に 2 種類のルートを提示する。被験者には、各ルートがどちらのルートで

あるかは知らせていない。提示するルート情報は、地図画像上にルートを強調表示した画像、航空写真上にルートを強調表示した画像、中間地点におけるストリートビュー画像である。ここで、中間地点はルートを距離に基づき 6 等分したとき、出発地点と目的地点を除く端点を選んだ。これら中間地点のストリートビュー画像を図 8 のように提示する。

各パターンにおいて、被験者に次のようなシナリオを提示する。例えば、(1) 彦根 → 白浜のパターンにおいては、以下のシナリオを提示する：

(s1) あなたは「彦根」から「白浜」へのドライブを計画しています。その際、せっかくなので眺めの良いルートを走りたいと考えています。

出発地と目的地の五つのパターンに (1)–(5) に合わせて、(s1) から (s5) の五つのシナリオを提示する。被験者は、各シナリオにおいて 2 種類のルートに対し、ドライブルートとしての相応しさを評価する。ルートの評価にはリッカート尺度 { (5) 非常に相応しい, (4) 相応しい, (3) どちらでもない, (2) 相応しくない, (1) まったく相応しくない } を用いる。

7.1 被験者

被験者はクラウドソーシングにより募集した。1 パターンあたり 30 名の異なる被験者に評価してもらった。5 パターンで計のべ 150 名の被験者に評価してもらった。なお、1 名の被験者が複数の対に回答することを許容している。

男女比率は 65%–35% である。30 歳–49 歳の年齢層が全体の 63% を占める。30 歳未満は 2%, 50 歳以上は 35% である。ドライブ経験については、「景色を楽しみながらドライブする」と回答した人は全体の 38% である。一方で、「娯楽目的のドライブはほとんど (まったく) しない」、「自動車の運転自体ほとんど (まったく) しない」と回答した人は 18% である。

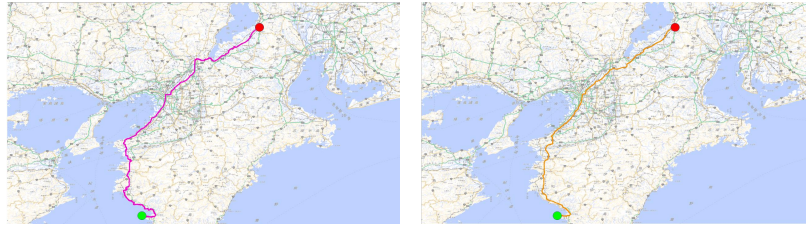
7.2 結果

リッカート尺度を間隔尺度とみなし、棒グラフにプロットする。図 9 は、シナリオごとに各ルート種別の評価値の平均をグラフに示したものである。

すべてのシナリオにおいて、眺望ルートの評価値がベースラインの評価値を上回った。これらの結果は、どのルートがどの景観ルートであるかを教示することなしに、シナリオに合った眺望ルートをシステムにより推薦できていることを示す。なお、s1, s2, s3, s4 においては、対応のある z 検定 (両側) を行った結果、 p 値が $p = 8.95 \times 10^{-11}, 5.72 \times 10^{-4}, 1.57 \times 10^{-2}, 5.77 \times 10^{-9}$ となり有意差を確認した ($p < 0.05$)。一方で、s5 においては、 p 値が $p = 0.08$ となり有意差は確認できなかった。s5 の各ルートにおいて、京都市、大阪市、神戸市といった都市部を通過するルートが提示された。図 3 より、都市景観の道路リンクには眺望の良い道路リンクが多く含まれる。そのため、最短ルートであっても眺望が良い道路リンクを通る可能性が高かった。その結果、眺望ルートと最短ルートの間で有意差が現れにくかったということがいえる。

8. 議論

6. 章より、眺望ベースルート推薦システムにおいて、眺望の



(a) 眺望ルート

(b) 最短ルート

図 7 被験者に提示する出発地と目的地を結ぶ 2 つの異なるルート。地図画像の出典：地理院タイル（淡色地図）をシステム上に提示し、キャプチャしたものを掲載。



(a) 眺望ルート 1.

(b) 眺望ルート 2.

(c) 眺望ルート 3.

(d) 眺望ルート 4.

(e) 眺望ルート 5.

図 8 眺望を考慮したルート内における、5 地点のストリートビュー画像。写真の出典：Google ストリートビュー画像をキャプチャしたものを掲載。

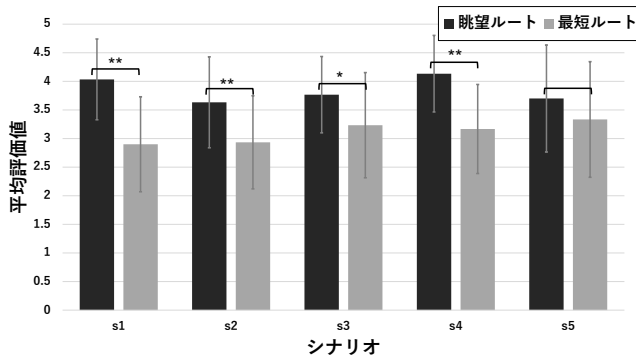


図 9 ユーザ評価結果。この図では、各シナリオにおける、各ルートの平均評価値を表す。エラーバーは、各評価値の標準偏差を表す。エラーバー上の * および ** は、Z 検定 (両側検定) における眺望ルートと最短ルートの有意差 ($p < 0.05$, $p < 0.01$) があることを示す。エラーバーにマークがない場合は、有意差がないことを確認できたことを示す。

良さを重視したルート探索が可能であることを示した。先行研究 [12] のシステムでは、景観特徴として田園景観、山林景観、水辺景観、都市系景観といった各景観要素を重視したルート探索を行った。この二つのルート探索手法を用いて、景観要素と眺望の良さを考慮した眺望景観ベースルート探索の簡易的な評価を行う。ここでは、各景観ごとの眺望景観ルートと景観ルートを比較し、評価する。

図 10 より、各景観ごとの眺望景観ルートと景観ルートを示す。淡緑線は田園系、濃緑線は山林系、青線は水辺系、赤線は都市系をそれぞれ重視したルートである。

図 10 から、眺望景観ルート、景観ルートともに各景観要素を重視した多様なルートが推薦されていることがわかる。田園系重視ルートはの奈良県の南部の田園地帯を通過している。山林系重視ルートはの鈴鹿山脈付近の山間地帯を通過している。水辺系重視ルートは川岸、湖岸、海岸線に沿ったルートとなっ

ている。都市系重視ルートは京都市、大阪市の都市部を通過するルートとなっている。

図 10(a) は図 10(b) に比べ、比較的眺望の良い田園地帯や海岸線を通過しているルートとなっている。また、図 5 の眺望ルートと比較したとき、目的地付近のルートにおいて重複するルートを推薦している。

以上より、各景観と眺望を考慮した眺望景観ルート探索が可能であると観測された。今後、眺望景観ルート探索において、より詳細に評価していきたいと考える。

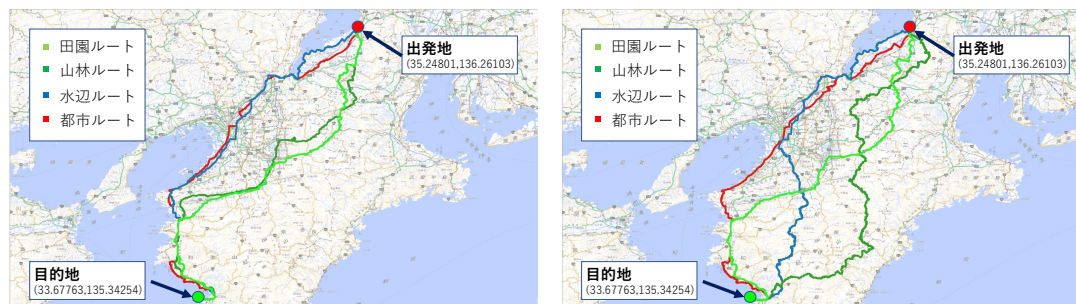
9. おわりに

本研究では、眺望の良さを重視したルートを探る眺望ベースルート推薦システムを提案した。提案システムでは、出発地・目的地を与えたとき、眺望の良さを重視したルートを提示する。推定に用いる道路リンクの特徴として、OSM の道路属性に加え、独自に定義した道路の景観ベクトル、可視直線距離、周囲との高低差、可視領域率を用いた。これらの特徴量を用いて眺望の良さは道路リンクごとに推定し、眺望の良さを優先したルートを推薦する。

OpenStreetMap から抽出された関西圏の主要道路の道路ネットワークデータ (道路ノード 728,684 件、道路リンク 1,340,011 件) を用いた評価実験を行った。評価実験の結果、下記の観点から提案システムの有用性を示した：

- 眺望タグの妥当性を定性的かつ定量的に示した。
- 提案システムにより提示された推薦ルートの妥当性について定性的かつ定量的に評価した。
- ユーザの観点から眺望ベースルート推薦システムを定量的に評価した。この結果から、特定のシナリオに対して眺望ベースルートを適切に推薦できることを示した。

また、簡易的な評価ではあるが、各景観と眺望を考慮した眺望景観ルート探索が可能であると観測された。今後、眺望景観ルート探索において、より詳細に評価していきたいと考える。



(a) 各景観ごとの眺望景観ルート

(b) 各景観ごとの景観ルート

図 10 出発地点を北緯 35.24801°, 東経 136.26103°, 目的地点を北緯 33.67763°, 東経 135.34254° としたときの各景観ごとの眺望景観ルートと景観ルート。淡緑線は田園系、濃緑線は山林系、青線は水辺系、赤線は都市系をそれぞれ重視したルートである。地図画像の出典：地理院タイル（淡色地図）をシステム上に提示し、キャプチャしたものを掲載。

謝 辞

本研究は JSPS 科研費 19K12567 の助成を受けたものです。
ここに記して謝意を表します。

文 献

- [1] Andrew V Goldberg and Chris Harrelson. Computing the shortest path: A search meets graph theory. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, No. March 2003, pp. 156–165, 2005.
- [2] Michalis Potamias, Francesco Bonchi, Carlos Castillo, and Aristides Gionis. Fast shortest path distance estimation in large networks. In *[CIKM2009]Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 867–876, 2009.
- [3] Evangelos Kanoulas, Yang Du, Tian Xia, and Donghui Zhang. Finding fastest paths on a road network with speed patterns. In *[ICDE2006]Proceedings of the 22nd International Conference on Data Engineering*, Vol. 2006, p. 10, 2006.
- [4] Hector Gonzalez, Jiawei Han, and Xiaolei Li. Adaptive fastest path computation on a road network: a traffic mining approach. In *[VLDB2007]Proceedings of the 33rd international conference on Very large data bases*, pp. 794–805, 2007.
- [5] Ling-yin Wei, Wen-chih Peng, Chun-shuo Lin, and Chen-hen Jung. Exploring Spatio-Temporal Features. In *[Springer]Advances in Spatial and Temporal Databases, Lecture Notes in Computer Science*, pp. 399–404, 2009.
- [6] Jing Yuan, Yu Zheng, Chengyang Zhang, and Wenlei Xie. T-drive: driving directions based on taxi trajectories. In *[SIGSPATIAL2010]Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 99–108, 2010.
- [7] Jing Yuan, Yu Zheng, Lihang Zhang, Xing Xie, and Guangzhong Sun. Where to find my next passenger. In *Proceedings of the 13th international conference on Ubiquitous computing*, pp. 109–118, 2011.
- [8] Zaiben Chen, Heng Tao Shen, and Xiaofang Zhou. Discovering Popular Routes from Trajectories. In *[ICDE2011]Proceedings of the 2011 IEEE 27th International Conference on Data*, Vol. 4, pp. 900–911, 2011.
- [9] Wuman Luo, Haoyu Tan, Lei Chen, and Lionel M. Ni. Finding time period-based most frequent path in big trajectory data. In *[SIGMOD2013]Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pp. 713–724, 2013.
- [10] Ling-yin Wei, Wen-chih Peng, Bo-chong Chen, and Ting-wei Lin. Eleventh International Conference on Mobile Data Management PATS : A Framework of Pattern-Aware Trajectory Search. In *[MDM2010]Mobile Data Management*, pp. 362–377, 2010.
- [11] Ling-Yin Wei, Yu Zheng, and Wen-Chih Peng. Constructing popular routes from uncertain trajectories. In *[KDD2012]Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 195–203, 2012.
- [12] Kenta Oku and Ryosuke Yamanishi. Roadscape vectorization of road links based on extraction of roadscape elements from land cover map (in Japanese). In *IPSJ SIG Technical Report*, Vol. DBS-165, pp. 1–6, 2017.
- [13] Koji Kawamata and Kenta Oku. Roadscape-based route recommender system using coarse-To-fine route search. *Electronic Preprint for Journal of Information Processing*, Vol. 27, , 2019.
- [14] Edsger Wybe Dijkstra. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, Vol. 1, pp. 269–271, 1959.
- [15] Rina Dechter and Judea Pearl. Generalized best-first search strategies and the optimality of A*. *Journal of the ACM*, Vol. 32, No. 3, pp. 505–536, 7 1985.
- [16] Kayur Patel, Mike Y. Chen, Ian Smith, and James A Landay. Personalizing Routes. In *UIST 2006: Proceedings of the 19th annual ACM symposium on User interface software and technology*, pp. 187–190, 2006.
- [17] Jaewoo Chung and Chris Schmandt. Going My Way : A User-aware Route Planner. In *[CHI2009]Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1899–1902, 2009.
- [18] Jianwei Zhang, Hiroshi Kawasaki, and Yukiko Kawai. A tourist route search system based on Web information and the visibility of scenic sights. In *Proceedings of the 2nd International Symposium on Universal Communication*, pp. 154–161, 2008.
- [19] 篠原修. 景観用語事典. 彰国社, 2007.
- [20] Daniele Quercia and Luca Maria Aiello. The Shortest Path to Happiness : Recommending Beautiful , Quiet , and Happy Routes in the City Categories and Subject Descriptors. In *[HT2014]Proceedings of the 25th ACM Conference on Hypertext and Social Media*, pp. 116–125, 2014.