

# 車両通過データからの 動的な交通ヒートマップ生成システム

金子 邦彦\*

Dynamic Traffic Heat Map Generation System from Vehicle Passing Data

Kunihiko KANEKO\*

## ABSTRACT

Traffic heat map dynamically plots the amount of vehicle traffic and the direction and magnitude of the traffic flow on a map. Vehicles are observed in real time and traffic heat map is generated dynamically and automatically. In the map, individual vehicles are not plotted. The density, speed, and flow of a group of vehicles are treated. The density, speed, flow rate, etc. are calculated and superimposed on the map. Since the vehicles move along the road, the way for calculating and plotting are different from other types of spatial data. In this preliminary report, a system for the generation of a traffic heat map using Python is presented.

Translated with [www.DeepL.com/Translator](http://www.DeepL.com/Translator) (free version)

キーワード：車両通過データ，交通流，交通ヒートマップ，交通状況把握

**Keywords:** vehicle passing data, traffic flow, traffic heat map, traffic situation understanding

## 1. はじめに

交通状況の把握と共有は、交通安全の維持、交通渋滞のよる諸問題の緩和、生活環境の維持、交通誘導などのさまざまな用途が見込まれ重要である。本報でいう交通状況は、個々の車両位置を追跡したり状況を把握したりすることではなく、多数の車両観測の結果から、特定の道路地点における交通量、交通流の方向、交通流の速さ、渋滞の有無などの交通状況を統計的に把握することである。

近年は、観測技術、画像等を人工知能で解析する技術が発展し、車載器が発する電波、ナンバープレートの画像、車両の画像、車両の通過音により、車両が特定地点をいつ通過したか、方向がどちらであったか、その日時がいつであったかを精度よく観測できるよ

うになってきた。車両の速度については、近接する複数観測点において車両の観測と車両の再識別を行うことにより可能である。車両の再識別は、ある地点を通過した車両について、別の地点において同一車両の特定を行うことである。車両の再識別には、車載器が発する電波の利用、車両画像の特徴点を追跡する画像特徴量やモーショントラッキングの技術の利用が考えられる。このとき、プライバシーや個人を特定する情報は収集されない。例えば、車両の中の人物の顔貌、人物の様子、ナンバープレート番号を収集することなく、車両の再識別ができるようになってきた。以上のように、車両の交通量、密度、交通流の方向と速さなどの交通状況をデジタルでリアルタイムに観測し、オンラインで収集できるようになってきた。

交通状況を把握することは、広域では、渋滞の頻発

する高速道路や主要道路や都市交通での交通状況把握に役立つ。狭域では、団体や個人等の駐車場（常設や臨時）、交通ターミナル（駅前での種々の交通機関の乗り換え場所など）での交通状況把握に役立つ。

交通ヒートマップは、本報では、交通情報を地図に重ね合わせ表示したものとする。その生成は簡単ではない。まず、車両観測の結果から、密度や速度や流量などを算出する必要がある。次に、交通ヒートマップの表示で、地図の上に重ね合わせて表示するために実装上の工夫が必要である。本報では、これらの説明を行う。2章では、交通ヒートマップと、それ以外での交通状況の提示法について説明する。3章では、リアルタイムかつ自動で交通ヒートマップを生成するシステムである動的交通ヒートマップ生成システムを説明する。

## 2. 交通ヒートマップ

交通状況をオンラインで配信し提示したり、サイネージ等で提示することは、すでに我々の生活に浸透している。

### ● VICS での渋滞情報提示

VICS での渋滞情報提示では、手元の車載器に、渋滞情報が配信される。そして、渋滞している区間が赤線等で表示される。このとき上り、下りなどの車線は区別して表示される。運転者にとっては、渋滞回避のために別経路をとったり、休憩などの判断に役立つ。

### ● 幹線道路等での到着予測時間提示

幹線道路等では、特定の地点までの到着予測時間がサイネージ等で表示される場合がある。これも、運転者にとっては、渋滞回避のために別経路をとったり、休憩などの判断に役立つ。

### ● 駐車場での満空情報提示

駐車場では、入り口の前に満空情報が表示される。運転者は、別の駐車場を探したりなどの行動をとる。

以上のように、交通情報は、運転者による的確な判断を助けるデータとなる。道路管理者にとっては、道路の改良の必要性や効果を裏付けるデータとなる。

交通ヒートマップは、交通量、交通の密度、交通流の方向、交通流の速さ、渋滞の程度という種々の量を、量に対応した色を付けた透明もしくは不透明の点で地図上にプロットするものである。道路管理等の運用では、有無よりも精密な情報が有用であり、急減速発生回数を多段階の色で表示することで交通安全対策に役立てる[1]ことも始まっている。本報も、さま



図 1. OpenStreetMap 地図上に表示した交通ヒートマップの例。

Fig.1 Example of a traffic heat map displayed on an OpenStreetMap map.

ざまな交通様態を交通ヒートマップとしてプロットすることを目指すものである。その詳細と生成法は3章で説明する。なお、表示されるものは、多数の観測から統計的に得られた量であって、個々の車両の様態を表示するものではない。また、1章でも説明したように、プライバシーや個人を特定する情報収集は考えないので、車両の追跡は行わず、「ある地点〇〇を出発した車両の最終目的地を統計的に調べる」といった、車両の集団としての振る舞いを追跡、解析するものでもないものとする。

## 3. 動的交通ヒートマップの生成システム

本章では、動的交通ヒートマップシステムでのデータ形式、データ処理手順、算出法、地図への表示法を説明する。

### (1) 車両観測に関するデータ形式

車両の観測において、車両の通過のたびに1つのレコードデータとして記録、そして、車両の再識別ができるたびに1つのレコードデータとして記録するものとする。レコードデータ集合には名前があるものとする。

### ● 車両通過のレコードフォーマット

レコード名：通過

属性：ID, lat, lon, at (それぞれ道路 ID, 緯度, 経度, 日時)

これは、ある道路 ID の道路のある緯度, 経度に、

ある日時に車両が通過したことを示す。

- 車両再識別のレコードフォーマット

レコード名：再識別

属性：ID1, lat1, lon1, at1, ID2, lat2, lon2, at2

これは、同一車両が、道路 ID 1 の道路のある地点 (lat1, lon1) を日時 at1 に通過し、道路 ID2 の道路の別の地点(lat2, lon2) を日時 at2 に通過したことを示す。

### (2) 道路に関するデータ形式

地図でのプロットにおいては、道路の上の点の列で表示する。観測地点の変動があったとしても、地図でプロットするときの道路の上の点の場所は動かないものとする。そのレコードフォーマットは次の通りである。

レコード名：道路点

属性：ID, km, lat, lon (それぞれ、道路 ID, 道路始点からの道路上での旅行距離, 緯度, 経度)

### (3) Python を用いたヒートマップ表示

Python の Pandas を用いて、レコードデータを簡単に扱うことができる。例えば 4 属性(lat, lon, value, at)を持つレコードデータを、Python のオブジェクト a に格納するプログラムは次のようになる。

```
import pandas as pd
d = pd.DataFrame([[43, 141, 0.1, 2018], [43, 141, 0.2, 2019], [43, 141, 0.3, 2020], [40, 140, 0.9, 2018], [40, 140, 0.6, 2019], [40, 140, 0.4, 2020]], columns=['lat', 'lon', 'value', 'at'])
```

Python の folium には、OpenStreetMap 地図上へのヒートマップ表示の機能がある。例えば、上の Python オブジェクトで(lat, lon) の場所に、value の値を使って点を表示するヒートマップを表示するプログラムは次のようになる。実行結果を図 2 に示す。

```
import pandas as pd
import folium
from folium.plugins import HeatMap
def plot_overlay(map_lat, map_lon, map_zoom, df, lat, lon, radius, filename):
    m = folium.Map(location=[map_lat, map_lon], zoom_start=map_zoom)
    HeatMap(df[['lat', lon]], radius=radius).add_to(m)
    m.save(filename)
    return m
```

```
plot_overlay(42, 140, 6, a[a['at']==2020], 'lat', 'lon', 20, 'map.html')
```



図 2. OpenStreetMap 地図上に表示したヒートマップの例。

Fig.2 Example of a heat map displayed on an OpenStreetMap map.

上に示したプログラムは、Pandas データフレーム a 内の全レコードをプロットするのではなく、条件「a['at']==2020」で、表示するデータを絞り込んで表示している。

レコードデータの中に、複数の日時に関するデータが入っているとき、日時に古いものから新しいものへ順にアニメーション表示することも、Python の folium で行うことができる。プログラムは次のようになる。

```
import pandas as pd
import folium
from folium.plugins import HeatMapWithTime
def plot_anim_overlay(map_lat, map_lon, map_zoom, df, lat, lon, weight, at, index, filename):
    result = []
    for i in index:
        result.append(df[df[at]==i][['lat', lon, weight]].values.tolist())
    m = folium.Map(location=[map_lat, map_lon], zoom_start=map_zoom)
    HeatMapWithTime(result, auto_play=True).add_to(m)
    m.save(filename)
    return result
result = plot_anim_overlay(42, 140, 6, a, 'lat', 'lon', 'value', 'at', [2018, 2019, 2020], 'map2.html')
```

このプログラム中で「[2018, 2019, 2020]」のように指定しているのは、アニメーションの1フレーム目が日時 2018, アニメーションの2フレーム目が日時 2019, アニメーションの3フレーム目が日時 2020 のように指定している。

#### (4) 交通情報ヒートマップ作成システム

Python を用いて交通情報のヒートマップ作成システムを試作し、動作させて考察を行った。考察の結果は次の通りである。

##### ● 範囲制約

速度であれば、範囲が0から60までになるという範囲制約がある。観測時の誤差、観測ミスにより範囲制約を違反するデータがあり得る。次のようなPythonプログラムで、範囲内に収まるように処理するか、あるいは、データ自体を取り除く必要がある。

```
a[a > 60] = 60
```

##### ● ヒートマップでの色付けの種類

Python の folium を用いたヒートマップ表示では、folium の HeatMapWithMap メソッドのオプションで、次のように、値と色の関係を種々変えることができる。見栄えを見てから、値と色の関係を変えて、より良い見栄えを探すことも大切であると感じた。

```
HeatMapWithTime(result, auto_play=True, gradient={0.1: 'black', 0.5: 'lime' })
```

##### ● マッチングの処理

車両通過と車両再識別のレコードにおける道路 ID と緯度と経度の情報から、道路点（(2)で説明）を探すというマッチングの処理が必要である。マッチング自体は、最近探索法などで行うことができ、本稿の範囲外とする。観測において、道路 ID が確定できる場合には、当該道路 ID の最近傍の道路点とマッチする。観測において道路 ID が分からない場合には、全道路の最近某の道路点とマッチする。

##### ● 観測での緯度経度の体系と、表示での緯度経度の体系の差異

表示での緯度経度は、(2)道路点で説明した通り、道路上の点の並びでの緯度経度である。観測での緯度経度の体系と、表示での緯度経度の体系は独立である必要がある。さもなければ、観測機器が動くたびに表示が変わり扱いづらい、表示において観測機器の場所が明瞭に分かるのは運用上の困難を招くなどのデメリットがある。そこで、観測での緯度経度を、表示での緯度経度に変換する処理を行う。

- ① まずは、観測の緯度経度で、集約を行う。集約によって、台数、方向の分布、速度の分布が求まる。
- ② 外れ値は除去するは、範囲内に収まるように変換する。
- ③ 観測での緯度経度から、(2)の道路点のマッチングを行い、道路点を特定する。
- ④ ③までの処理を終えたら、次の処理の前に、全データを、道路ごとに分ける。
- ⑤ 道路始点からの旅行距離を使い、①の結果を補間処理する。その結果、道路点における台数、方向の分布、速度の分布が求まる。
- ⑥ 分布のままでは扱いづらいため、中央値をとるなどの処理を行う。

#### 4. 終わりに

Python の機能を用いて、自動で、交通ヒートマップを生成できることを示した。システムの試作と評価により、観測での緯度経度の体系と、表示での緯度経度の体系は独立したものであるべきことを見出し、その実現のために、どのような処理を行うかを明らかにできた。本報は、いずれ、ETC 等での車両通過データがオープン化されるであろう将来を見据えてのものである。性能評価等はそのときに改めて実施したい。

#### 謝辞

交通ヒートマップの要件や想定される用途について、ETC システム株式会社片山賢治氏にさまざまなご示唆をいただいたことに感謝する。

- [1] 尾崎悠太・川瀬晴香・瀬戸下伸介, ETC2.0 プロジェクトの交通安全対策への活用, 59-4, 2017.