一般社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
SWIM2018-9,SC2018-16（2018-08）

ミクロな人口統計データの活用を容易化するWeb-APIの開発

香川　拓大†　佐伯　幸郎†　中村　匡秀†

† 神戸大学 〒657–8501 神戸市灘区六甲台町 1–1
E-mail: †kagawa@ws.cs.kobe-u.ac.jp, ††sachio@carp.kobe-u.ac.jp, †††masa-n@cs.kobe-u.ac.jp

あらまし　人口統計データは，その都市や町の性格を最もよく表すデータの一つである．特に自治体が政策を考える際に人口データは非常に重要な情報である．神戸市では人口統計データをオープンデータとして公開しており，町丁目ごとのミクロなレベルで男女別，年齢別の人口を知ることができる．しかしながら，現状ではデータは機械判読できないデータ形式で公開されており，利用者にとって複雑な表構造の理解が必要である．そこで本研究では，神戸市の人口統計データに簡単にアクセスするためのWeb-API，Kobe Demographics APIを開発する．このAPIを用いることで，期間，場所，性別や年齢など，様々な観点や粒度で人口統計データを取り出すことができる．また，開発したAPIを用いて，神戸市におけるミクロレベルの人口の可視化を行う．
キーワード　オープンデータ，人口，Webサービス，スマートシティ，可視化

Developing Web-API for Integrating Micro-Level Demographic Data

Takuhiro KAGAWA†, Sachio SAIKI†, and Masahide NAKAMURA†

† Kobe University, Rokkodai 1–1, Nada, Kobe, Hyogo, 657–8501 Japan
E-mail: †kagawa@ws.cs.kobe-u.ac.jp, ††sachio@carp.kobe-u.ac.jp, †††masa-n@cs.kobe-u.ac.jp

Abstract　Demographic data is one of the best data that characterizes a city or town. Especially, when local governments work out a policy, demographic data is very informative. Kobe city has published fine-grained demographic data which is grouped by gender and age for each ward, town, town block. However, the current major format of demographic data is not machine-readable format. Thus, an application developer has to understand a complicated table structure. In this paper, we develop Web-API, which is called *Kobe Demographics API*, for easy access to demographic data of Kobe city. It enables users to efficiently query demographic data at various viewpoints and granularity such as period, location, gender, and age. Using the API, we also visualize the micro-level population in Kobe city.
Key words　open data, demography, Web service, smart city, visualization

1. はじめに

その都市，町にどれだけの人が住んでいるかを表す人口統計データは，都市，町の性格を最もよく表すデータの一つである．自治体が政策を考える際や，企業が地域の特性に応じた戦略を展開する時，もしくは一般市民がイベントを開催する際など，人口統計データは様々な場面で活用される[1] [2]．近年，特に政府や自治体において，公開可能な公共データをオープンデータとして広く公開して，人々の生活向上や企業活動の活性化を狙う活動が見られ，人口統計データも政府や自治体のホームページ等で得ることができる．例として，総務省統計局はホームページで人口統計データを公開しており，日本における人口の推移，都市別の人口，年齢別の人口など，様々なデータを得ることができる[3]．統計局が公開している人口統計データの一つを表1に示す．

しかしながら，人口統計データは，現状ではXLS形式などの表形式で公開されていることがほとんどである．そのため，注目するデータの取り出しには複雑な表構造の理解が必要であり，一般のユーザにとって利用が難しい．その上，XLS形式は機械判読が難しく，外部のWebアプリ等から参照することもできない．また，統計局が公開している人口統計データは，都道府県レベル，もしくは都市レベルに留まり，それよりミクロな町丁目レベルのデータは含まれていない[4]．ミクロな人口統計データが得られなければ，地域に寄り添った政策や，イベントの開催は困難である．

一方，神戸市では，オープンデータを公開するためのポータルサイトを公開し，人口などの市政情報，施設の情報，市営地下鉄の時刻表など，様々なデータを配信している[5]．その中でも，人口統計データに関しては，市や区単位でなく，町や丁目ごとの非常にミクロなレベルでデータを公開している．ま

表 1　人口統計データの例 (2016 年)

都道府県	国勢調査人口					平成28年推計人口		
	平成22年	27年	人口集中地区	人口密度	人口増減率（平成22～27	総人口	人口性比（女性100に対する男性）	人口増減率（対前年）（人口1,000につき）
	(1,000人)	(1,000人)		(人／km²)	(%)	(1,000人)		
			1)	2)				
全国	128,057	127,095	86,868	a)340.8	-0.8	126,933	94.8	-1.3
北海道	5,506	5,382	4,047	a)68.6	-2.3	5,352	89.1	-5.6
青森	1,373	1,308	610	135.6	-4.7	1,293	88.6	-11.3
岩手	1,330	1,280	408	83.8	-3.8	1,268	92.8	-9.1
宮城	2,348	2,334	1,495	320.5	-0.6	2,330	95.6	-1.6
				.				
				.				
				.				

た，男女別の人口や，1 歳ごとの年齢別の人口や世帯数といった情報も知ることができる．その上，神戸市では四半期ごとに人口統計データを公開しているため，時系列による比較が可能である．しかし，統計局のデータ同様，神戸市における人口統計データは，機械判読が困難な PDF 形式や XLS 形式で公開されている．これらの形式のデータは利用時にまず表構造の理解を必要とする．また，外部のアプリ，サービスからデータを参照することができないため，民間による自発的なサービスの創造は期待しにくい状況である．

このような課題を解決するため，本研究では神戸市のミクロな人口統計データにアクセスするための API，**Kobe Demographics API** を開発する．この API は年月や場所，性別，年齢等をパラメータとして人口を検索し，その結果を返す．これにより，表構造の理解を必要とせず，町丁目別，年齢別等の様々な観点や粒度で人口統計データを取り出すことが可能になる．

API の利用例としては様々なものが考えられるが，その一つとして，人口統計データの可視化が考えられる．我々は，開発した API と政府が公開する町丁目の区画データを組み合わせ，人口の可視化を行うシステムを開発した．これにより，神戸市において，ある年代の人口がどの地域で多いのかを町丁目レベルで知ることができる．

2. Kobe Demographics API

2.1 概　　要

1. で述べたように，現在公開されている人口統計データは，利用時に複雑な表構造の理解を必要とし，外部のサービス等から参照することもできない．このような課題を解決するために，我々は神戸市のミクロな人口統計データに簡単にアクセスするための API，Kobe Demographics API を開発する．開発にあたり，我々は表形式で公開されている人口統計データを扱いやすいよう構造化してデータベースに格納する．その後，データベース内のデータにアクセスするための Web-API を実装する．API を用いることで，場所，年齢，年月，性別などの様々な観点や粒度で人口統計データを取り出すことが可能である．これにより，表構造を理解することなく，データを活用することができ，外部アプリからの参照も可能になる．また，他のデータと掛け合わせた分析も容易化される．

2.2 データベース

神戸市の人口統計データは PDF 形式や XLS 形式で公開されているため，そのままでは外部のサービスから参照することができない．そのため，まず我々は表形式で公開されているデータを扱いやすいように構造化して関係データベースに挿入する．ここでは，町丁目ごとのミクロな人口データを格納する population テーブル，及び神戸市の町丁目のデータを格納する town テーブルを作成する．

population テーブルのスキーマは以下の通りである．

[**population**]

- **townCode:** 町丁目を識別するためのコード
- **year:** データが公開された年
- **month:** データが公開された月
- **household:** 世帯数
- **male:** 男性の人口
- **female:** 女性の人口
- **total:** 全人口
- **0:** 0 歳の人口
- **1:** 1 歳の人口
- **2:** 2 歳の人口
- **... :** (1 歳ごとの人口が格納される)
- **99:** 99 歳の人口
- **100:** 100 歳の人口

population テーブルは町コードと年，月を主キーとし，人口データを属性として持つ．1 歳ごとの人口を属性として持つことで，ある町丁目のある年月の 1 歳ごとの人口といった非常にきめ細やかな情報まで扱うことができる．

town テーブルのスキーマは以下の通りになっている．

[**town**]

- **townCode:** 町丁目を識別するためのコード
- **address:** 町丁目の住所
- **wardName:** 町丁目が位置する区
- **townName:** 町丁目の名称
- **wardCode:** 町丁目が位置する区を識別するためのコード
- **latitude:** 住所から算出した緯度
- **longitude:** 住所から算出した経度

town テーブルは町コードを主キーとし，神戸市の町丁目の住所などの情報を格納する．Google Maps などの外部サービスから町丁目を参照する場合，住所ではなく緯度，経度が必要となる．そのため，データベースへの格納の際，住所から緯度と経度を地理情報 Web サービスを利用して算出し，情報として付与している．

これら二つのテーブルに加え，人口密度を算出する際を考慮し，町丁目の面積を格納する area テーブルを作成する．ま

た，ミクロな人口データを利用する際には，ミクロな地図情報が必要なユースケースが多い事が予想される．したがって，町丁目単位の地理データを格納する shape テーブルを合わせて作成する．面積データ，地理データとしては，日本政府が公開している e-Stat のデータを用いる．e-Stat では政府が持つ様々な統計データが公開されているが，その中に全国の町丁目の基本情報を記述した KML ファイルがある．このデータを利用しやすいように加工して使用する．

area テーブルのスキーマは以下の通りである．

[**area**]

- **townCode:** 町丁目を識別するためのコード
- **area:** 町丁目の面積
- **perimeter:** 町丁目の周辺長

area テーブルは町コードを主キーとし，面積と周辺長の情報を格納する．特に面積は人口密度などの情報を算出する際に使われる．

また，shape テーブルのスキーマは以下の通りである．

[**shape**]

- **townCode:** 町丁目を識別するためのコード
- **seq:** 通し番号
- **lat:** 境界を表す点の緯度
- **lng:** 境界を表す点の経度

一つの町丁目の境界線は約数十の境界点からなる．それらの点をつなぎ合わせた線の内部がその町丁目の領域を表す．この境界を示すデータを我々はシェイプデータと呼ぶ．

データベースの実装には MySQL を使用した．また，データベースへの挿入には Perl スクリプトを利用することで，元データを自動的に解析して構造化し，効率的に挿入できるようにしている．

2.3 Kobe Demographics API の開発

前節で述べたデータベースに格納されたデータにアクセスするために，getPopulation()，searchTown()，getArea()，getShape() の四つの API を開発する．

getPopulation() は指定されたパラメータで人口情報を検索し，その結果を JSON 形式で返す．getPopulation() のパラメータは以下の通りである．

getPopulation(townCode, wardCode, year, month, gender, from, to)

parameters:

- **townCode:** 町コードを指定する
- **wardCode:** 区コードを指定する
- **year:** 調査された年を指定する
- **month:** 調査された月を指定する
- **gender:** 性別を指定する
- **from, to:** 集計する年齢の範囲を指定する

人口統計データを扱う際，単純な人口の多さではなく人口密度を扱う場合も多い．また，高齢化率など，ある年齢の人口の全人口に対する割合も重要な情報である．そのため，API の返り値としては，指定された町の情報と検索された人口に加え，指定された年齢の人口を町の面積で割った人口密度，及びその町全体の人口に対する割合を JSON 形式で返す．このとき，面積の取得には後述する getArea() を利用する．例えば，町コードが 101001001 の町の 2018 年 3 月の 0 歳から 10 歳の人口を検索する場合，getPopulation(101001001, -, 2018, 3, -, 0, 10) を実行する．API は以下の JSON を返す．


```
[JSON]....{"address":"神戸市東灘区魚崎北町 1 丁目",
"density":0.001792286052654514,
"from":0,
"gender":"total",
"latitude":34.7181473,
"longitude":135.2759896,
"month":3,
"percentage":0.0991636798088411,
"population":83,
"to":10,
"townCode":101001001,
"townName":"魚崎北町 1 丁目",
"wardCode":28101,
"wardName":"東灘区",
"year":2018}
```


返却値から，神戸市東灘区魚崎北町 1 丁目の 2018 年 3 月時点の 0 歳から 10 歳の人口は 83 人であり，0 歳から 10 歳の人口密度は約 0.0018 人/m^2 で，町全体の人口の約 10%であることが分かる．API のパラメータを変えることで，期間や場所，性別，年齢等の様々な観点や粒度で人口統計データを取り出すことが可能である．

getPopulation() は町コードをパラメータとするが，一般のユーザにとって検索したい町丁目の町コードを直接指定するのは困難である．そこで，二つ目の API，searchTown() を実装する．

searchTown(address, wardName)

Parameters:

- **address:** 住所を検索するためのキーワード
- **wardName:** 検索する区を指定する

この API はパラメータとしてキーワードが与えられると，そのキーワードを住所に含む町や丁目を検索し，その町丁目の情報を JSON で返す．例として，魚崎北町 1 丁目の町コードを検索したい場合，searchTown(”魚崎北町 1 丁目”) を実行すると，以下の JSON が返される．

[JSON]....{"address":"神戸市東灘区魚崎北町 1 丁目",
"latitude":34.7181473,
"longitude":135.2759896,
"townCode":101001001,
"townName":"魚崎北町 1 丁目",
"wardCode":28101,
"wardName":"東灘区"}

魚崎北町 1 丁目の町コードが 101001001 であることが分かった．

三つ目の API，getArea() は町コードをパラメータとして面積と周辺長を JSON 形式で返す．これを利用することで，人口密度の計算等に役立てることができる．

getArea(townCode)

Parameters:

- **townCode:** 町コード

町コード 101001001 をパラメータとして getArea() を呼び出すと，以下の JSON が返される．

[JSON]....{"townCode":"101001001",
"area":"46309.572",
"perimeter":"866.564"}

四つ目の API，getShape() は町コードをパラメータとし，シェイプデータを JSON 形式で返す．

getShape(townCode)

Parameters:

- **townCode:** 町丁目を識別するコード

町コードが 101001001 の丁目のシェイプデータを取得したい場合，getShape(101001001) を実行する．返却される JSON は以下の通り．

[JSON]....{
{"townCode":"101001001","seq":"1",
"lng":"135.276767965","lat":"34.716835936"},
{"townCode":"101001001","seq":"2",
"lng":"135.2771268815","lat":"34.7169571043"},
....,
{"townCode":"101001001","seq":"23",
"lng":"135.276767965","lat":"34.716835936"}
}

これら四つの API を Web サービスとしてデプロイした．外部のサービス，アプリから API を通して神戸市のミクロな人口統計データや面積データ，シェイプデータを取得することができるようになる．

2.4 実　　装

Kobe Demographics API の開発にあたって使用した技術は以下の通りである．

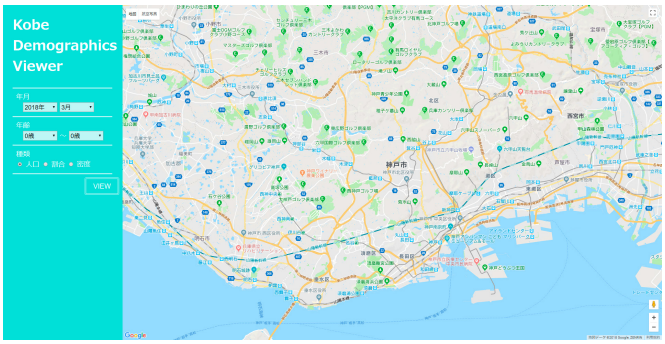

図 1　Kobe Demographics Viewer

- **人口統計データ:** Open Data Kobe - 住民基本台帳 (日本人・外国人) 町丁目別・年齢別人口(注1)
- **面積，周辺長データ，シェイプデータ:** e-Stat - 地図で見る統計 GIS (注2)
- **データ解析スクリプト:** Perl, Yahoo! GeoCoder API
- **データベース:** MySQL
- **API:** Java JDK8, Jersey

3.　ミクロな人口統計データの可視化

人口統計データの利用例としては様々なものが考えられるが，データを可視化することでどの地域に人が多く住んでいるかといった情報が一目で把握できる．ここでは，**2.** で開発した API の利用例として，神戸市におけるミクロな人口統計データの可視化を行う．

3.1 概　　要

ミクロな人口統計データの可視化のために，我々は人口，密度，もしくは割合に応じて町丁目を色分けして可視化する新たなシステムを開発した．我々はこのシステムを **Kobe Demographics Viewer** と呼ぶ．Kobe Demographics Viewer のスクリーンショットを図 1 に示す．システムのユーザは画面左のプルダウンメニューから年月，年齢を指定する．また，ラジオボタンで人口，密度，割合の中から可視化したい種類を選択する．その後，「VIEW」ボタンを押すと指定した条件に応じた可視化が開始される．

システムが神戸市における人口統計データを可視化する手順は以下の通りである．

Step1:　Kobe Demographics API を用いてある町丁目の人口，密度，割合のいずれかを算出する

Step2:　算出した値に応じて，色を決定する

Step3:　その町丁目のシェイプデータを取得する

Step4:　シェイプデータで決定される領域を Step2 で決定した色で色付けする

(注1)：http://www.city.kobe.lg.jp/information/data/statistics/toukei/jinkou/juukijinkou.html
(注2)：https://www.e-stat.go.jp/gis

Step5: 神戸市の全ての町，丁目について Step1〜4 を繰り返す

3.2 人口，密度，割合の算出

Kobe Demographics API を用いて，ある町丁目の人口，密度，割合を算出する．算出には町コードが必要なため，まず searchTown() を用いて町コードを取得する．ここでは検索する区名をパラメータとして与える．例えば，「東灘区」をパラメータとして与えると，東灘区全ての町，丁目の情報が得られる．それら一つ一つの町丁目について人口，密度，割合を調べていく．得られた町コードと指定された年月，年齢をパラメータとして getPopulation() を呼び出すことで，人口，密度，割合の値を得る．

3.3 色の決定

3.2 で算出した値に応じて，町丁目を色付けするための色を決定する．まず，人口，密度，割合いずれかの値に応じて町丁目を 10 段階に分類する．人口を用いる場合は，神戸市の全ての町丁目の中での人口の最大値を基準として用いる．町丁目 t における人口を $p(t)$，人口の最大値を M_p としたとき，t を以下の 10 段階に分類する．

(1) $p(t) = 0$
(2) $0 < p(t) \leqq M_p/32$
(3) $M_p/32 < p(t) \leqq M_p/16$
(4) $M_p/16 < p(t) \leqq M_p/8$
(5) $M_p/8 < p(t) \leqq 3 * M_p/16$
(6) $3 * M_p/16 < p(t) \leqq M_p/4$
(7) $M_p/4 < p(t) \leqq 3 * M_p/8$
(8) $3 * M_p/8 < p(t) \leqq M_p/2$
(9) $M_p/2 < p(t) \leqq 3 * M_p/4$
(10) $3 * M_p/4 < p(t) \leqq M_p$

密度を用いる場合も同様に，神戸市の全ての町丁目の中での密度の最大値を基準とする．町丁目 t における密度を $d(t)$，人口の最大値を M_d としたとき，t を以下の 10 段階に分類する．

(1) $d(t) = 0$
(2) $0 < d(t) \leqq M_d/32$
(3) $M_d/32 < d(t) \leqq M_d/16$
(4) $M_d/16 < d(t) \leqq M_d/8$
(5) $M_d/8 < d(t) \leqq 3 * M_d/16$
(6) $3 * M_d/16 < d(t) \leqq M_d/4$
(7) $M_d/4 < d(t) \leqq 3 * M_d/8$
(8) $3 * M_d/8 < d(t) \leqq M_d/2$
(9) $M_d/2 < d(t) \leqq 3 * M_d/4$
(10) $3 * M_d/4 < d(t) \leqq M_d$

図 2 は 2018 年 3 月の神戸市の各町丁目ごとの人口を表しているが，上位数%の町丁目の人口が飛び抜けて多くなっている．そのため，人口，及び人口密度の分類の基準は，人口，もしくは人口密度の大きいグループ ((9)，(10) など) の範囲を広めに，逆に小さいグループ ((1)，(2)，(3) など) の範囲は細かく設定している．

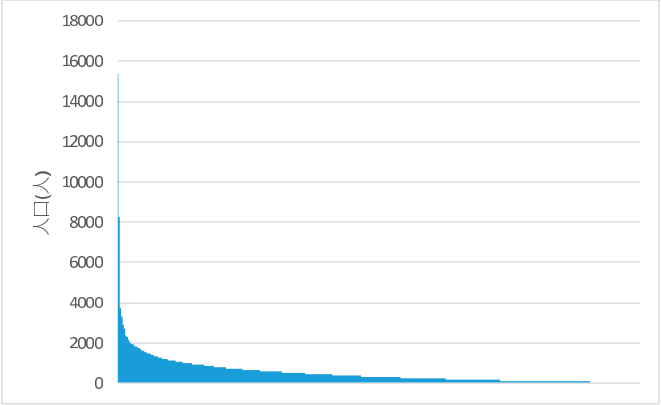

図 2 神戸市の各町丁目ごとの人口 (2018 年 3 月)

割合を用いる場合は，**3.1** で述べたプルダウンメニューで選択される年齢の幅 w を用いる．$from$ 歳から to 歳を指定した場合，w は以下の式で求められる．

$$w = to - from + 1$$

幅 1 歳分を 1%と考えた $W(= w/100)$ を基準とする．町丁目 t における割合を $r(t)$ とすると，t は以下の 10 段階に分類される．

(1) $r(t) = 0.0$
(2) $0.0 < r(t) \leqq W/8$
(3) $W/8 < r(t) \leqq 2 * W/8$
(4) $2 * W/8 < r(t) \leqq 3 * W/8$
(5) $3 * W/8 < r(t) \leqq 4 * W/8$
(6) $4 * W/8 < r(t) \leqq 5 * W/8$
(7) $5 * W/8 < r(t) \leqq 6 * W/8$
(8) $6 * W/8 < r(t) \leqq 7 * W/8$
(9) $7 * W/8 < r(t) \leqq W$
(10) $W < r(t) \leqq 1.0$

上記の 10 段階の分類が [灰色，青，群青，水色，青緑，緑，黄緑，黄色，橙，赤] に対応する．例えば，(1) に分類された町丁目は灰色で，(2) に分類された町丁目は青で色付けされる．

3.4 人口のマップへの可視化

色を決定した後に，**2.3** で述べた getShape() を用いて町丁目のシェイプデータを取得する．このシェイプデータで決定される領域を **3.3** で決定した色で色付けする．これを神戸市の全ての町丁目について繰り返すことで，それぞれの町丁目が 10 色のうちいずれかで色付けされることになる．2018 年 3 月における神戸市の全ての町丁目の 0 歳〜10 歳の人口を可視化したものを図 3 に示す．

山間部や工場地帯等，人が住んでいない地域では人口データそのものが存在しないため，一部色付けされていない地域もあるが，神戸市において，人口の多い地域，少ない地域がミ

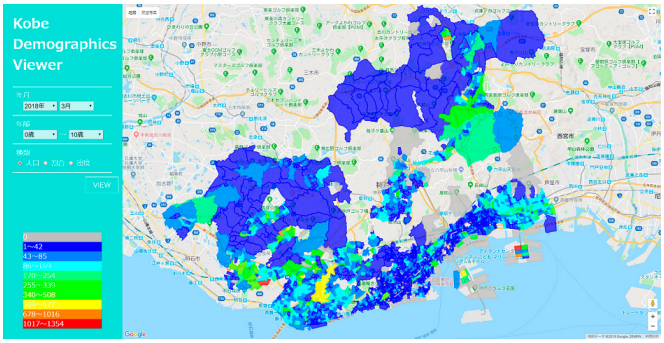

図 3　神戸市における 0 歳から 10 歳の人口の可視化 (2018 年 3 月)

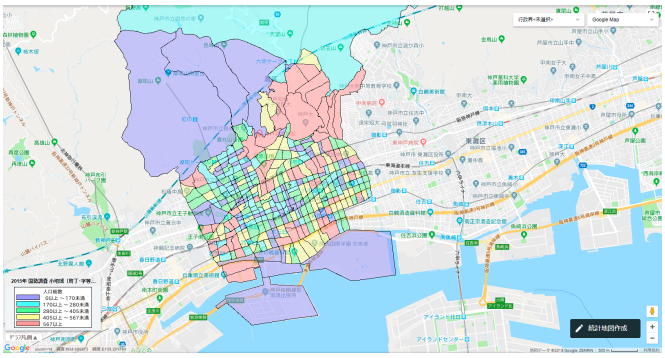

図 4　jSTAT MAP を用いた神戸市灘区における人口の可視化 (2015 年)

クロなレベルで可視化されている様子が分かる．

4. 考　　察

4.1 想定される他のユースケース

Kobe Demographics API の他のユースケースとして，一般のユーザが地域の人々向けのイベントを開催する場合を考える．例えば，地域の高齢者向けのイベントを開催する際，開催場所は参加者が集まりやすい場所であることが望ましい．Kobe Demographics API を用いると年齢をパラメータとして人口を検索できるため，どの町丁目にどれだけの高齢者が住んでいるかを調べることができる．また，Kobe Demographics Viewer を用いることで，高齢者の人口をミクロなレベルで可視化することができる．

その他に，これまでの町丁目単位の人口の推移を調べることができるため，この先の人口の推移を見越して，適切な場所に消防署や病院等を建てるといった利用法も考えられる．

4.2 関 連 研 究

Kobe Demographics API とよく似たサービスに，RESAS [6] の人口データがある．RESAS とは産業構造や人口動態，人の流れなどに関するビッグデータを集約し，可視化を試みるシステムで，日本の内閣府によって提供されている．また，API 機能の提供が 2016 年に開始され，RESAS の様々なデータに API を用いて誰でも簡単にアクセスすることができる．RESAS によって提供されるデータの中に人口構成のデータがあり，都道府県や市区町村をパラメータとして指定することで，調べたいエリアの人口を検索することができる．しかし，RESAS が提供する人口データは市区町村単位のものまでであり，それより細かな丁目別のデータは提供されていない．また，性別や 1 歳ごとの年齢別のデータも提供されていない．

Kobe Demographics Viewer とよく似たサービスに，総務省統計局が公開する jSTAT MAP [7] がある．jSTAT MAP は Web 上で無料で使用できる地理情報システムであり，政府が持つ統計データを簡単に地図上に可視化することができる．統計データの中には町丁目レベルの人口統計データがあり，Kobe Demographics Viewer と同様にミクロな人口統計データを可視化可能である．jSTAT MAP を用いて 2015 年の神戸市灘区における人口統計データを可視化したものを図 4 に示す．しかしながら，jSTAT MAP で使用される人口統計データは API 化されていないため，外部のサービスから呼び出すことはできない．また，国勢調査を基にした人口統計データを使用しているため，使用できるデータは 5 年ごとのデータのみである．

5. お わ り に

本稿では，神戸市のミクロな人口統計データに簡単にアクセスするための API，Kobe Demographics API を開発した．API は場所，年齢，年月，性別などをパラメータとし，人口，密度，割合を算出して JSON 形式で返却する．API を通して，様々な観点や粒度で人口統計データを取り出すことが可能である．また，開発した API を活用し，ミクロな人口統計データを可視化するシステムを開発した．可視化を行うことで，どの町丁目に人口が集中しているかが非常に掴みやすくなる．

今後の研究としては，人口統計データと他のデータとを掛け合わせた分析を考えている．掛け合わせるデータの例として，犯罪等の事件データや消防局が持つ緊急出動のデータが挙げられる．人口の多い地域，少ない地域でどのような傾向があるかといった分析は，様々な分野で活用できると考える．

謝辞　この研究の一部は，科学技術研究費（基盤研究 B 16H02908，18H03242, 18H03342，基盤研究 A 17H00731），および，立石科学技術振興財団の研究助成を受けて行われている．

文　　献

[1] 森川　洋，“2010 年の人口移動からみた日本の都市システムと地域政策，”人文地理，vol.68，no.1，pp.22–43，2016.
[2] 金山　剛，“熊本市におけるまちづくり及び政策への統計データの活用，”熊本大学政策研究，vol.2，pp.105–116，March 2011.
[3] 総務省統計局，“総務省統計局ホームページ，” http://www.stat.go.jp/data/nihon/02.html，July 2018.
[4] 磯田則彦，“都道府県別人口移動統計の整備状況について，”地理学評論 Ser. A，vol.66，no.10，pp.639–644，1993.
[5] 神戸市，“オープンデータ一覧，” http://www.city.kobe.lg.jp/information/opendata/catalogue.html，July 2018.
[6] 内閣府，“RESAS，” https://resas.go.jp/, Aug. 2018.
[7] 総務省統計局，“jSTAT MAP，” https://jstatmap.e-stat.go.jp/, Aug. 2018.