

(62) 汎用3次元データへの変換による 基盤地図情報の活用

山崎 俊夫¹

¹正会員 函館工業高等専門学校 社会基盤工学科 (〒042-8501 北海道函館市戸倉町17-1)

E-mail:toshi_ya@hakodate-ct.ac.jp

国土地理院で一般公開されている基盤地図情報に基づいて、3次元の都市モデルを構築する方法を取りまとめた。具体的には「建築物の外周線」データ(XML形式)をシェープファイルに変換した後、GISソフト(QGIS)に読み込む。GISソフトでこの建物データに階数などの属性情報を付加し、アスキー形式の地図データフォーマット(GeoJSON)で出力する。これを自作プログラムで3次元化したデータを汎用3次元データフォーマット(COLLADA)で出力する。出力した3次元データは、入手が容易な3DCGアプリ(SketchUp)などにより読み込みが可能である。無償の地図データ、GISソフト、プログラム言語、3DCGアプリの活用により、CIM導入時におけるコストを下げ、CIMの裾野を広げることができる。

Key Words : Spatial Information Infrastructure, 3DCG, GIS, GeoJSON, COLLADA

1. はじめに

平成21年度より、国土地理院により整備された「電子国土基本図」が、インターネットにおいて無償で提供されている。電子地図における全ての位置の基準となる情報として、「基盤地図情報」が公開されている。オープンデータである基盤地図情報は、今後、G空間情報を活用するうえで、基盤となる位置情報を与えるものである。さらに、国土交通省が取り組みを推進しているCIMにおいても、基盤地図情報は、基盤となる地形データや建物データを提供するものである。

2. 本研究の目的

筆者は、2005年頃より合意形成ツールとして3DCGによる都市モデルの構築に取り組んでいる。3次元都市モデルの作成方法の概要は、GISソフトをデータベースとして、自作のプログラムにより地図データを3次元化し、それを3DCGアプリに読み込ませるといったものである。

これまでの間、国土地理院より電子地図が無償で提供され、高機能なGISソフトがフリーウェアで公開されるなど、取り巻く状況が随分と整ってきた。さらに、CIMにより設計データ等の3次元化の必要性が高まると共に、G空間情報においても3次元による空間基盤整備の必要

性が高まると考えられる。

こうしたことから本研究では、基盤地図情報に基づいて都市空間の3次元モデルを制作し閲覧する方法を明らかにする。その際に使用するデータやソフトウェアは、インターネットより無償で入手できるものを基本とする。このように低コストで3次元都市モデルを構築する方法を明らかにすることにより、CIMへの取り組みを必要としている業界関係者等の一助になることを期待するものである。

3. 汎用3次元データフォーマットの解説

本研究で定義する汎用3次元データとは、主要な3DCGアプリで読み込みが可能な形状データである。代表的な汎用形状フォーマットとしては、“DXF” “3DS” “OBJ”がある。CIMに関してはAUTODESK社の「Infra Works」や「Auto CAD Civil 3D」がメジャーなアプリケーションである。同社の「3ds Max」は代表的な3DCGアプリであり、「Auto CAD Civil 3D」とのデータ互換性が高い。3DSフォーマットは「3ds Max」のファイルフォーマットであり、多くのCGソフトがサポートする中間ファイルフォーマットであるが、バイナリーフォーマットである。

これに対してCOLLADAファイルはアスキー形式であ

り、インターネットで無償で提供されている3DCGアプリである「SketchUp」での読み込みが可能である。「3ds Max」においてもプラグインを介して読み込み可能である。こうしたことから、本研究ではCOLLADAファイルを汎用3次元データフォーマットとして採用した。

“COLLADA”は、CGソフト間のデータ交換用ファイルフォーマットである。元々はソニー・コンピュータエンタテインメントによりPlay Station 3向けの開発用公式フォーマットとして作成された形式である。“COLLADA”はXMLファイルであり、拡張子はdaeである。

“COLLADA”の仕様書（リリース1.4.1 2006年6月）とリリースノート（リリース1.5.0 2008年6月）がインターネットで入手可能である（2015年6月時点）。仕様書には付録Aとして簡単なCOLLADA文書が記載されている。これと共に「SketchUp」から出力した単純な形状のCOLLADAファイルより、COLLADAファイルの構造を解読した。その結果、COLLADA（ルート）要素の下にasset, library_effects, library_materials, library_geometry, library_visual_scenes, sceneの6つの要素で構成されていることが判った。次いで、「SketchUp」より単純な2個の形状をCOLLADAファイルで出力したものより、assetとsceneの要素が形状の数に無関係であることが判った。そして、assetとscene以外の要素には形状の数に応じて繰り返し記述する必要がある要素があることが確認できた。さらに、url, 各種idについては形状毎にユニークでなければならないことも判った。

R言語で3次元化プログラムを作成する際には、こうしたCOLLADAファイルの仕様を満たすことを目指した。“COLLADA”が“DXF”や“OBJ”よりも優れていると言えるのは、180°以上の内角を持つ複雑な平面形状を簡単に立体化することができることである。

```

<COLLADA>
<asset> ... </asset>

<library_visual_scenes> ...
  <instance_geometry url="#ID1">
    ...データ毎に必要な情報...
  </instance_geometry>
... </library_visual_scenes>

<library_geometries>
<geometry id="ID1">
  <mesh>
    ...データ毎に必要な情報...
    [ 頂点の位置 ][ 頂点の法線 ][ 頂点の識別情報 ][ ポリゴンの情報 ]
  </mesh>
</geometry>
</library_geometries>

<library_materials>
<material id="ID1-material">
  ...データ毎に必要な情報...
</material>
</library_materials>

<library_effects>
<effect id="ID1-surface">
  ...データ毎に必要な情報...
</effect>
</library_effects>

<scene> ... </scene>
</COLLADA>

```

図-1 COLLADAファイルの構造

“DXF”や“OBJ”の場合、180°以上の内角を持つ多角形は複数の三角形に分割しなければならない。そのためプログラムが複雑になる。しかし、“COLLADA”の場合はそのまま3次元化することが可能であり、多角形を三角形に分割するルーチンが必要ないという利点がある。

4. アスキー形式地図データフォーマットの解読

GISにおいてはESRI社の「ArcGIS」が代表的なアプリケーションである。ゆえに、「ArcGIS」の標準ファイルフォーマットであるシェープファイル(Shape File)が業界標準となっている。しかし、シェープファイルはバイナリーファイルであるため、アスキー形式で出力される地図データを必要とした。

GeoJSONはさまざまな地理データ構造をコード化するためのオープンフォーマットである。GeoJSONはJSON(JavaScript Object Notation)に基づいていることから命名されている。あらゆるGeoJSONデータ構造がJSONオブジェクトであり、JSONツールをGeoJSONデータの処理に使用できる利点がある。

JSONはテキストベースのデータ交換のために設計されている。JSONでは基本的に2種類のデータ構造がサポートされている。1つはオブジェクトのように名前と値のペアからなるコレクションであり、もう1つは配列のように値を要素として持つ順序付きリストである。

図-2はGeoJSONファイルの基本的なフォーマットである。

- オブジェクト：オブジェクトはブレース({ })で始まりブレース(})で終わる。
- オブジェクトのメンバ：オブジェクトの要素はコロンの(:)で区切られた文字列と値で構成される。メンバが複数ある場合は、カンマ(,)で区切られる。

```

{
  "type": "FeatureCollection",
  "features":
  [
    { "type": "Features", ← 「データ数のカウントのキー」
      "properties":
      {
        (属性データ)
      },
      "geometry":
      { "type": "Polygon",
        "coordinates":
        [
          (座標データ) ← 「頂点数をカウントする必要がある」
        ]
      }
    }
  ]
}

```

図-2 GeoJSONファイルの構造

- ・ **値**：値は文字列、数値、そしてtrueやfalse、nullといったリテラルのいずれかである。文字列は二重引用符で囲まなければならない、内側にはUnicodeの文字とバックスラッシュまたはエスケープ文字を含むことができる。

このようにXMLと比べても単純な構造であり、解読は比較的容易であった。“type”：“Features”のオブジェクトが個々の形状データの区切り（単位）となっている。“properties”の下位に建物の属性データを示すオブジェクトのメンバが並んでいる。また、“coordinates”の下位に建物の頂点座標値がX座標とY座標のペアでブラケット（[]）で囲まれて並んでいる。

R言語では、GeoJSONファイルから建物の属性データと頂点座標値を読み込んで、屋根形状ならびに頂点数に応じてモデリングするプログラムを作成した。

5. 基盤地図情報との対応方法の検討

基盤地図情報とは、電子地図における位置の基準となる情報のことであり、2007年（平成19）に成立した地理空間情報活用推進基本法で規定され、整備がはじめられた。2015年6月現在、国土地理院のウェブサイト<<http://fgd.gsi.go.jp/download/>>において、基盤地図情報のダウンロードが可能である。ダウンロード可能なファイルは、JPGIS(GML)形式の「基盤地図情報基本項目」と「基盤地図情報数値標高モデル」である。

3次元都市モデルの構築では、主に「建築物の外周線」のデータを使用する。以下に、GISソフト(QGIS)にデータを読み込むまでの手順について説明する。

国土地理院のウェブサイトにアクセスし、基盤地図情報基本項目のJPGIS(GML)形式を選択する。次いで、地図もしくはリストから対象地を選択してファイルをダウンロードする。

また、同じウェブサイトから「基盤地図情報ビューア」もダウンロードする。そして、ダウンロードした「基盤地図情報ビューア」を起動する。ファイルメニューから「新規プロジェクトの作成」を選択し、表示されたダイアログの追加ボタンを押して、zip形式で圧縮されたダウンロードしたフォルダを選択し、地図データを画面に表示する。そして、必要範囲を画面に適切に表示し、エクスポートメニューから「エクスポート」を選択する。変換する要素のチェックマークをチェックし、変換結果はシェープファイルを選択し、「現在表示されている要素のみを出力」で変換を行う。出力されたシェープファイルをGISソフト(QGIS)に読み込む。

建築物の外周線には、idを含む10項目の属性情報が既に存在する。3次元都市モデルを構築するためには、建

物の高さのデータを追加する必要がある。個々の建物の詳細な高さデータを入手することは困難であるが、住宅地図より建物の階数を得ることができる。よって、GISソフト(QGIS)により建物の階数を属性データとして追加する。なお、対象となる建物の件数は数百件と膨大な量になるため、1件ずつ属性データを入力することは非常に煩雑な作業になる。そこで、レイヤーメニューの「名前を付けて保存」からコンマ区切りファイル(CSV)の形式で出力する。このファイルを「Excel」で開き、階数(story)等の項目名で列を追加し階数データを入力する。一般的には2階建の家屋が多いため、階数を2階として一括入力しておき、マンション等の中高層建築物についてはGISソフト(QGIS)において個別に対応すればよい。属性データを追加したCSVファイルをレイヤーメニューの「デリミティッドテキストファイルを追加」により、「ジオメトリなし（属性のみのテーブル）」で読み込む。そして、レイヤに読み込まれた建物属性データを元の「建築物の外周線」レイヤと結合させる。

このように階数データを「建築物の外周線」に追加することにより、基盤地図情報の正確な座標データに基づいて建物を立体化し、3次元の都市モデルを構築することが可能になる。

6. 建築物の外周線データの3次元化

(1) 立体データのモデリング方法

平面形状の座標データと高さのデータから3次元の立体をモデリングする方法は以下のとおりである。

建物の平面形状は四角形の場合、頂点が4点ある。これら4点は(x1,y1)~(x4,y4)の座標値を持つ。これらを3次元の空間座標に展開した場合、高さの値は全て同じZ値となる。

次に、この建物の高さがhである場合、建物の平面形状のデータをそのままZ方向にhだけ移動させれば、その建物の上面データになる。この時、上面4頂点の座標値は(x1,y1,z1+h)~(x4,y4,z4+h)となる。

3次元形状のモデリングは、平面形状が四角形の場合、8つの頂点座標を用いて6つの面を定義するものとなる。

図-3は4頂点の四角形を立体化する概念を図示したものである。なお、この考え方は日本で多く見られる切妻屋根の家屋にも適用可能である。屋根形状や屋根勾配、軒・けらばの出などを屋根部分の属性データとして与え、平面形状の4頂点の座標値と建物高さから屋根形状のモデリングを行うものである。

このように、平面の多角形を立体にするという方法を、全ての建物データに適用することで3次元の都市モデルの製作が可能になる。

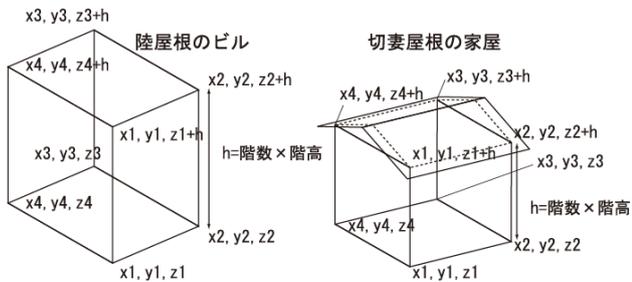


図-3 モデリング方法の概念

(2) 「R言語」によるモデリング

プログラム言語にはR言語を採用した。R言語を採用した理由は、無償のソフトウェアであるからである。当初、R言語は統計解析向けのプログラミング言語とされているため、3次元のモデリングには不向きではと思われたが杞憂であった。

これまで「REALbasic」や「Visual Studio C#」を使って本システムの開発を行ってきた。R言語を使用することの利点として、まず最初にあげられるのはCSVファイルが読み込めることである。

アスキー形式の地図データファイルを扱う場合、最初に障壁となるのは、いかに属性データや座標データを文字列の中から読み取るかということである。これまでは地図データファイルの中から、鍵（キー）となる文字列を検索し、そこから何文字目から何文字目までをデータとして読み取るということを行ってきた。

こうしたテキスト処理の方法では、地図データの出力において各データの文字数がどこかで1文字異なるだけで、データの読み込みでエラーが生じることになる。

GeoJSONファイルを「Excel」に読み込むことで、二重引用符に囲まれた文字列は文字データとして、カンマ

(,)で区切られた文字列は数値として読み込まれる。R言語はCSVファイルを読み込めるので、「Excel」に読み込んだGeoJSONファイルからブラケット()などの不要な列を削除し、必要なデータのみを整理したデータをCSVファイルで出力し、R言語に読み込ませる。

この方法を採用することにより、読みこんだファイルからデータを読み取るプログラムは、大いに簡素化することができたと共に、プログラムエラーも起こりにくくなった。

R言語はフリーウェアであることから、今後、3次元化プログラムを一般に公開することで、協力者を得ることも可能になるのではないかと考えられる。

7. おわりに

本論では、基盤地図情報を汎用3次元データに変換することで、簡易的に3次元都市モデルが構築できることを示した。さらに、本方法では、オープンデータである基盤地図情報をはじめとして、シェープ・ファイルへのコンバータ、建物の属性情報を付加してアスキー形式の地図データを出力するGISソフト、その地図データを3次元化するプログラム言語、3次元化したデータを表示する3DCGアプリまで、ほぼ全てが無償で入手可能である。国土交通省が取り組みを進めているCIMに対して、業界関係者等における導入時のコストを下げることにより、CIMの裾野を広げることが期待できると考えられる。

なお本研究は、一般財団法人日本建設情報総合センターの研究助成を受けて進めている。

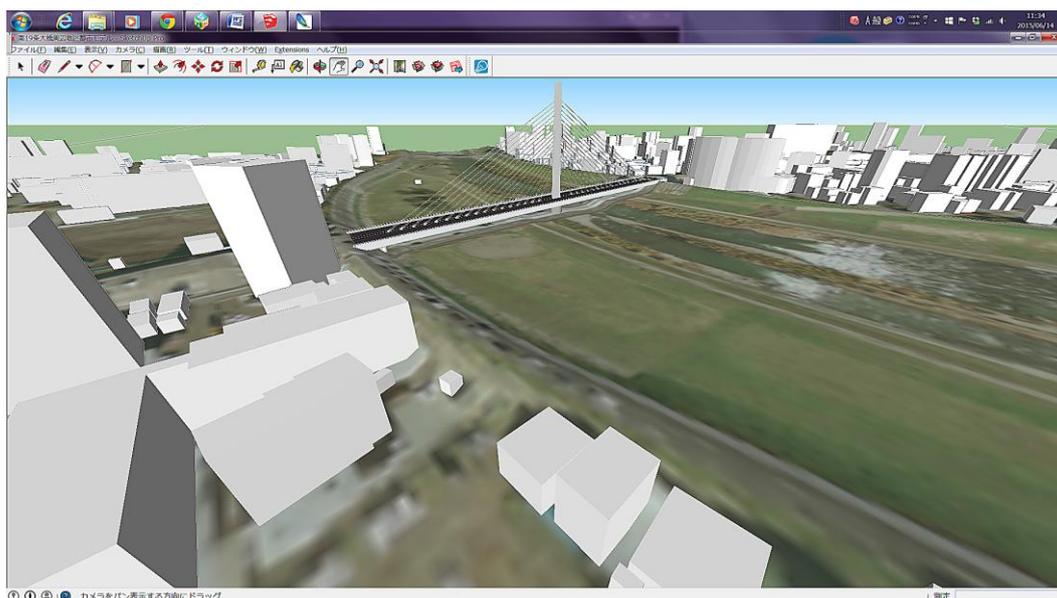


図-4 本システムで作成した3次元都市モデルと予備設計想定桥梁モデル