

タイトル	坑道実測図，地層断面図の3次元モデル生成法
著者	串山，繁；新井，大地；大辻，康太；KUSHIYAMA, Shigeru；ARAI, Daichi；OHTUJI, Kohta
引用	北海学園大学工学部研究報告(38)：66-78
発行日	2011-02-14

# 坑道実測図，地層断面図の3次元モデル生成法

串山 繁\*・新井大地\*・大辻康太\*

## Creating Method of a Three Dimensional Model of Coal Mine Tunnel Maps and Stratum Cross Sections

Shigeru KUSHIYAMA\*, Daichi ARAI\* and Kohta OHTUJI\*

### 要旨

空間的な拡がりを持つ対象物に対して、3次元（3D）モデルを容易に作成できれば、視覚を通じた理解の深まりが促進される。地球科学など分野によっては、3DモデルをGoogle Earth（GE）などの地理情報と重ね合わせるにより更なる理解の向上が期待される。従来の3Dモデルは、CADの専門オペレーターが有償ソフトで作成したものか長時間の制作手間を掛けた模型が殆どであった。本報告では、炭鉱の坑道実測図、地層断面図を例に、フリーでダウンロード可能なGoogle SketchUp（GSU）を用いて3Dモデルを作成し、それをGEに重ねて表示する手法と作品例を提示して、相対的に安価な本手法の有用性を示す。

## 1 はじめに

GSU<sup>1)</sup> (ver. 7.0) は、特別なトレーニングをせずとも修得が容易なCADソフトである。GSUは、GSUユーザーズガイド<sup>2)</sup>などのマニュアルを参照して手動の操作ができると同時に、一方でGSU API (GSU Application Programming Interface) を介してオブジェクト指向言語のRuby<sup>3)</sup>で記述したスクリプトも実行でき、複雑なモデルも比較的簡単に作成できる。また、GSUユーザーの作成した各種パーツやスクリプトが多数登録<sup>4)</sup>され、フリーでダウンロード可能となっている。

Rubyは、複数のサイトからフリーでダウンロード<sup>5)</sup>でき、Rubyによるスクリプトは、wordpressなど通常のテキストエディタ或いはRuby開発環境としてフリーで提供されているRDE<sup>6)</sup>などで記述できる。なお、本格的なGSUのためのスクリプトを自前で開発するには、スクリプトを記述する観点からGSUの構成について記した解説書があるとよい。例えば、参考文献<sup>7)</sup>は

---

\* 北海学園大学工学部建築学科

\* Department of Architecture and Building Engineering, Faculty of Engineering, Hokkai-Gakuen University

GSUのデータ構造：GSU module, Model class, Entities classの概念や主要なメソッドの使用方法について解説している。

GSUで作成された3Dモデルは、GE (ver. 5.0) に重ね合わせることが出来、正確な位置合わせはGE側で可能である。pathを設定すればデモンストレーション用の動画も作成可能で、pathは手動操作でもKML (Keyhole Markup Language) 言語でも設定でき、完成品はKMZファイルとして保存される。ただし、地表面の半透明化にはGEの最新版でも未だ対応しておらず、地下構造物は上空にリフトアップするかミラー反転して表示せねばならない。

以下には、炭鉱の坑内実測図を例に、3D座標値データファイルの作成、GSUでの3Dモデル作成手法、地層断面図取込み手法について述べ、GEに重ねた3つの炭鉱の作品例を示す。

## 2 3D座標値データファイルの作成

図-1は、北炭幌内炭鉱の坑内実測図 (立坑付近) である。より正確には、1882年11月に全線開通した官営幌内鉄道 (手宮～幌内間) 終点の幌内駅側ではなく、幌内駅とは山を隔て北側2 km, 唐松の新幌内小学校から南西約660mに位置する昭和42年開削された立坑を坑口とする新々幌内炭鉱 (新幌内小学校から東約1.3kmの新幌内炭鉱と統合) の坑内実測図である。この実測図は坑道に加え立坑および斜坑坑口、三角点 (図-1には記載無し) などの基準点座標値と東西南北方向の1 kmグリッドメッシュが記入された2次元平面図である。

2次元平面図から $x$ ,  $y$ 座標値を正確、迅速に読み取るため、本報告ではMatlab言語でマウスをコンピュータ画面上でクリックしてpixel座標値を得るGINPUTコマンド

```
[gx, gy]=GINPUT(100); %Please push Enter-key at the last data
```

を用い、デジタイザーの機能を有するプログラムを作成した。座標値読み取り誤差は極力抑えることが望ましい。本報告では、メッシュの交点が必要2点含まれるよう坑内実測図を5分割し、分割図面毎にコンピュータ上に拡大表示し座標値を読み取った (図-1は分割図no.3)。なお、 $z$ 座標値については、図面上に表示されている基準点 $z$ 座標値、斜坑進入角度、主要水平坑道の海拔深度に基づき、プログラム内で数値や計算式を与えて確定した。坑道の深さに関しては、坑道を完全に復元するために十分な情報が必ずしも記入されておらず推定部位も多数存在するが、これについては坑道の読み取り座標値を2D, 3D表示して互いに連結し矛盾が無いかを確認した。確定した座標値はファイルデータとして保存し、GSUのスク립トに引き渡す。

座標値読み取りプログラムのフローは、次に示す通りである。

- 1) 縮尺倍率, 座標変換マトリックスの規定～図-2に示す様にコンピュータ上の画像は、原点を左上隅,  $x$ 方向水平右向き,  $y$ 方向下向きを正方向とするpixel単位のコンピュータ画像座標系, GSUの画像は原点を適当な基準点 (例えば立坑位置),  $X$ 方向東向き,  $Y$ 方向

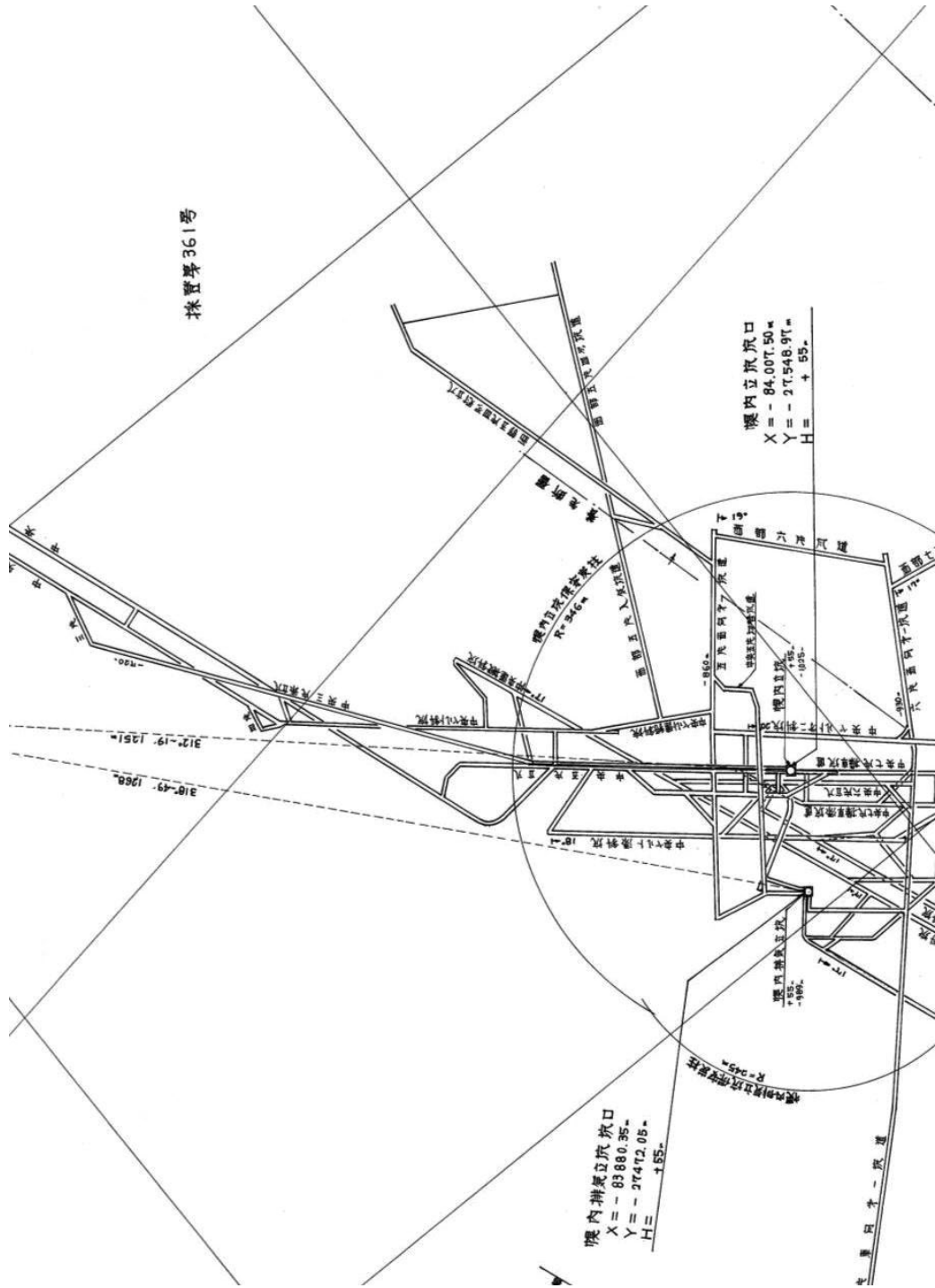


図-1 北炭腰内炭鉱坑内実測図（立坑付近）

北向きを正方向とするm単位のGSU座標系である。したがって、コンピュータ画像座標系からGSU座標系への座標変換が必要となり、幌内の例では南北方向の1kmグリッドメッシュの2つの交点座標値を基に、次式で縮尺倍率と座標変換マトリックスを規定すればよい。

$$\begin{Bmatrix} X \\ Y \end{Bmatrix} = scale \cdot [T] \cdot \begin{Bmatrix} 0 & 1 \\ 1 & 0 \end{Bmatrix} \cdot \begin{Bmatrix} x \\ y \end{Bmatrix}$$

ただし、 $scale$ ：縮尺倍率＝基準点間m単位長さ／基準点間pixel単位長さ、

$$[T] = \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix} : \text{座標変換マトリックス, } \phi = \theta + \pi \text{ (}\phi : x \text{軸とY軸の挟む角)}$$

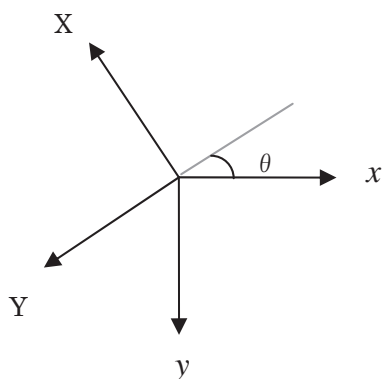


図-2 北炭幌内炭鉱：コンピュータ画像座標系(x-y)とGSU座標系(X-Y)の関係

- 2) 座標値読み取り，座標値データファイルの作成～GINPUTコマンドを用いた座標値の読み取りは坑道の始点，折れ点，分岐点に対して行う。ただし，その読み取りはGSUの`followme`メソッドの使用制約条件（`followme`が可能な対象は，一筆書きの`path`）に従わねばならない。それ故，多数の一筆書き坑道に分割して読み取り，それらのデータは連続通し番号のファイル名を付与して保存する。
- 3) 坑道の位置，連結の確認～Matlabのプログラムで座標値保存データファイルを読み取り，坑道を2D，3D表示して基準点と坑道の位置関係，坑道間の連結，斜坑と水平坑道の連結の整合性を確認，修正する。なお，坑道の連結確認，修正作業については，坑道をラインとして表示するMatlabよりもチューブとして表示するGSUの方が3D表示図の拡大が滑らかで把握し易いので，GSU側で確認してもよい。

### 3 GSUによる坑道モデル，地層断面モデルの作成およびGEの重ね合わせ

#### 3.1 坑道モデルの作成

##### (a) 3D座標値保存ファイルデータの読み込み

3D座標値保存ファイルデータの読み込みは，先ず全ての $x$ ， $y$ ， $z$ 座標値を纏めて適当な変数名に代入し，次に1レコード単位で読み込まれたデータを $x$ ， $y$ ， $z$ 座標値に切り分ける．その後， $z$ 座標値末尾の改行コード $\backslash n$ を`chomp`メソッドで削除する．読み込み時，文字列データとして認識されるデータを数値データに変換した後，座標値データの単位 $m$ を $inch$ に変換する(換算乗数は $10000/254$ )．なお，GSU側で $m$ 単位を使用する場合であっても，スクリプト内の長さの単位は $inch$ 単位を使用するルールとなっている．

### (b) pathの設定

`path`を規定する際，一筆書きの坑道データを構成する座標点数が任意であるため，それに対応できる様なコードとしなければならない．これについては点の座標値を順に結合する方式で，可変長さの連続データを作成し，`add_curve`メソッドを用いて`path`を作成した．

### (c) 坑道断面faceの設定とfollowme

坑道の断面は，円形断面形状とした．矩形の上に半円を重ねたアーチ断面形状についても検討したが，GSUで描画すると，`path`の交差角度によっては一部の坑道底面が表示されないGSU側でのバグが生じたので除外した．立坑は円形断面でその外径は，赤平の立坑工事記録<sup>8)</sup>を参照し，半径 $4.0m$ とした．坑道断面半径は，赤平炭鉱の坑道断面積：最小値約 $10m^2$ ，最大値約 $34m^2$ を参照し， $r=2.6m$ と仮定した．以下に円形断面`face`の設定について述べる．

`face`を作成する際の仮定は，次の通りである．

- ・ `face`は水平面に常に垂直である．したがって，斜坑の坑口も常に鉛直平面上にある．
- ・ `face`は`path`の軸線に直交する．

`path`の起点A，第2点Bの座標値を夫々 $(x_A, y_A, z_A)$ ， $(x_B, y_B, z_B)$ ，半径を $r$ と置き，更に $(x_B - x_A) = a$ ， $(y_B - y_A) = b$ と置く．このとき，円形断面の`face`は，`add_circle`メソッドを用いて，例えば次の様に記述すればよい．

```
circle=entities.add_circle [ax[1], ay[1], az[1]], [c, s, 0], r
```

ただし，`[ax[1], ay[1], az[1]]`：`path`の起点の座標値， $r$ ：円の半径

`[c, s, 0]`：normal vectorの各成分， $c = a/\sqrt{a^2 + b^2}$ ， $s = b/\sqrt{a^2 + b^2}$

なお，立坑断面を定義する際には，円形断面の`face`が水平面上に存在するので，`path`が下向きの場合には，normal vectorは`[0, 0, -1]`となる．`face`作成後，`followme`すれば，ひとつの坑道モデルが描画される．これを一筆書きの坑道数分，繰り返せばよい．

スクリプト例は本報告の末尾に掲載する．スクリプトは，開発中には`Plugins`フォルダ内のサブフォルダ(下記の例：`test`)に保存し，完成後`Plugins`フォルダに移動するとよい．

スクリプト保存先：C：¥Program Files¥Google¥Google SketchUp7¥Plugins¥test

スクリプトの実行は，メニューバーの`window/Ruby Console`選択画面下で次の様に入力する．

スクリプト実行：`load "test/ Ponbetu_CircleDraw_coalmine_path.rb"` Enter-key押下

### 3.2 地層断面図および補助水平面の作成

坑道と併記する地層断面図は、地表面からある深度までの地層の重なりと夾炭層を表す図で、海拔のレベル表示、地層名称の凡例が付加される。通常、東西南北軸とは一致しない任意角度の地層断面図を正確に配置するには、スクリプトによるのが簡単である。必要な入力データは地層断面図四隅の $x$ 、 $y$ 、 $z$ 座標値のみである。レベル表示、地層凡例のためのfaceの追加は地層断面図描画後に追加する側の隣接edgeを延ばすと簡単である。

スクリプトでface作成後、事前にコンピュータに取り込んだ地層断面図（RGB 24-bit jpeg形式）をAdobe Photoshopなどで異なる地層毎に色分け彩色した図をGSUにimportして貼り付け、更にレベル表示、地層凡例を貼り付けて作業完了となる。ただし、この状態では地層断面図の背後に隠れた坑道は見えないので、import画像を貼り付けた台紙のedgeを削除し、更に地層断面図を半透明化する。図-3は、赤平の地層断面図の一例である。

補助水平面は、半透明のシートを適当な海拔で複数枚モデル図に挿入し、坑道モデル図の坑道レベルを知る手掛かりの為のガイドであるが、シートを挿入したが故に坑道が見えづらくなる事態を避け、大きさと色、透明度を調整する。彩色、半透明化は、図-4に示すペイントアイコンを選択し、次画面で同図右上のマテリアル作成アイコンをクリックし、左下のマテリアル画面を表示する。カラーホイールで希望色を選択し、微調整はピッカーの右横欄にてRGBに切り換え実行、不透明度も指定しOKを押す。次画面で選択タグを押すと同図右下のマテリアル作成後画面が表示される。選択色を確認、faceを選択すると彩色、半透明化が完了する。なお、レベル表示のラベルは別途jpeg形式で作成し、シート四隅のいずれか1つに配置する。以上、GSUで3Dモデルが完成したらSKPファイルとして保存する。

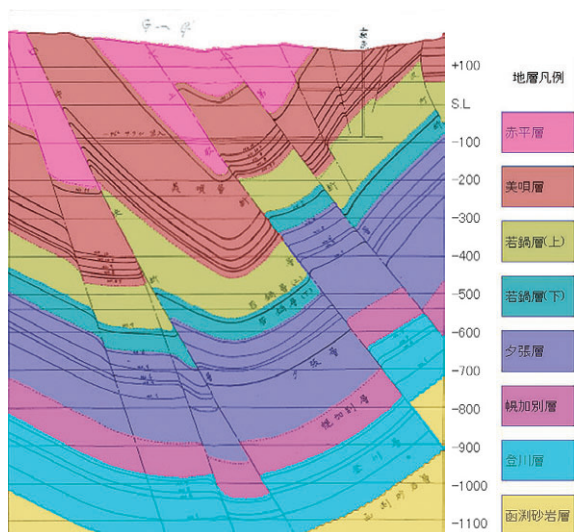


図-3 住友赤平炭鉱：G～G'地層断面図



図-4 彩色・透明化画面

### 3.3 GEの重ね合わせ

まず，GSUを起動して3Dモデルを読み込む．重ね合わせは，GEを起動しモデル配置場所を表示後，GSUに戻りGEの「現在のビュー」を取得し，GEに「モデルを配置」を実行すれば，完了する．GEでの3Dモデルの正確な位置合わせは，先ず適当なパスを追加すると表示されるサイドバーの場所／Modelを右クリック，プロパティを選択する．次画面にてGSUモデルの原点の緯度，経度を移動先の緯度，経度に変更して水平移動する．高度については，標高タグを開き高さを指定してリフトアップすればよい．完成品はKMZファイル形式として保存する．GE上での動画作成は，フリーソフトのNotepad++などを用いればKMLで記述できる．

## 4 作品例

図一5，6は，夫々北炭幌内炭鉱坑道の真上から見た俯瞰図と少し近づき北西上空から見た俯瞰図である．坑道をリフトアップしてGEに重ねると，画像周辺部では坑道が正しい位置からずれて見えるが，同図から立坑より北側にある坑道は幾春別川を挟み唐松地区に拡がり，反対の南側ではベルト斜坑坑口からの長い斜坑が幌内を跨ぎ繋がっている様子が分かる．

図一7，8は，半透明の補助水平面を非表示とした住友赤平炭鉱の俯瞰図である．赤平については坑道全体の実測図を入手出来なかった為，「硐口位置図（全赤平）」記載の坑道のみ表示している．同図から坑道の拡がり，北側の北排気立坑～南側の上歌立坑までの南北約5.3km，二斜坑が空知川を跨ぐ中央部で東西約2.5km幅となっていることが分かる．地層断面図からは，南北方向の軸に沿って向斜構造となっていること，図一8では識別出来ないが先の図一3を参照すると夾炭層は美唄層，若鍋層（下），夕張層，登川層であることが分かる．

図一9は住友奔別炭鉱の坑道図である．赤平同様坑道全体の実測図を入手出来なかった為，奔別立坑工事記録<sup>9)</sup>に記載されている昭和35年以前の坑道概略図から同図を作成した．坑道の拡がり，略図に忠実であるが，坑道レベルについては非表示が多く推測で描いた．同図から弥生橋町の斜坑，奔別旭ヶ丘の斜坑と北の山側での坑道の拡がりを確認できる．奔別炭鉱閉山時には，山側の浅部ではなく幾春別側の深部で採炭されており，その深度は海拔-1000～-1100mであった様だが，それは図一10の地層断面図と符合している．

## 5 まとめ

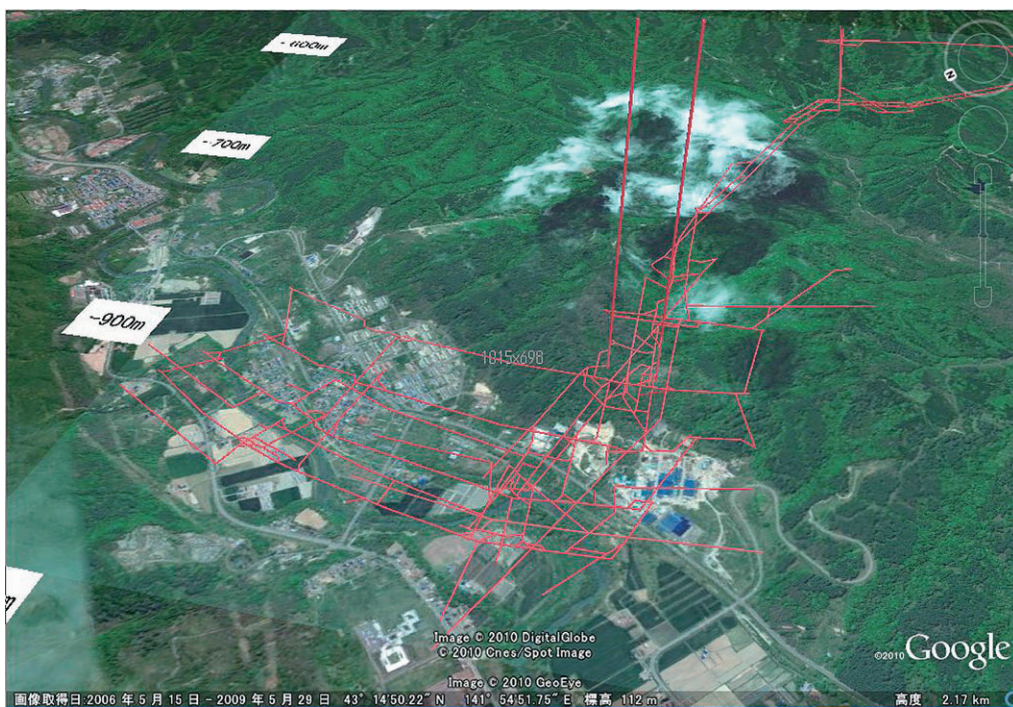
フリーソフトのGoogle SketchUpとGoogle Earthを組み合わせて，炭鉱の坑道図，地層断面図の3Dモデル生成法について述べ，容易に有用なモデルを作成できることを示した．

なお，幌内炭鉱坑内実測図の入手にはNPO法人「炭鉱の記憶推進事業団」吉岡宏高理事長，赤平炭鉱の硐口位置図（全赤平），地層断面図の入手には，らびか館の吉田勲氏，赤平市市会議員の植村真美氏の多大な協力を得ました．記して感謝申し上げます．





図一五 北炭幌内炭鉱 (1250mリフトアップ)



図一六 北炭幌内炭鉱 (1250mリフトアップ)

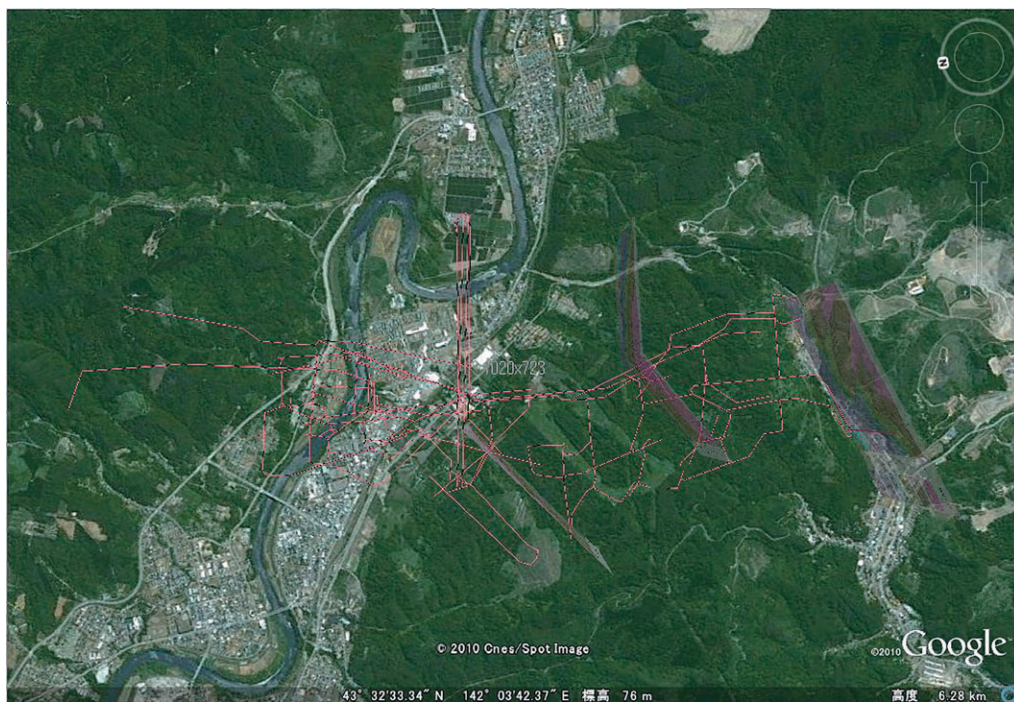


図-7 住友赤平炭鉱 (900mリフトアップ)

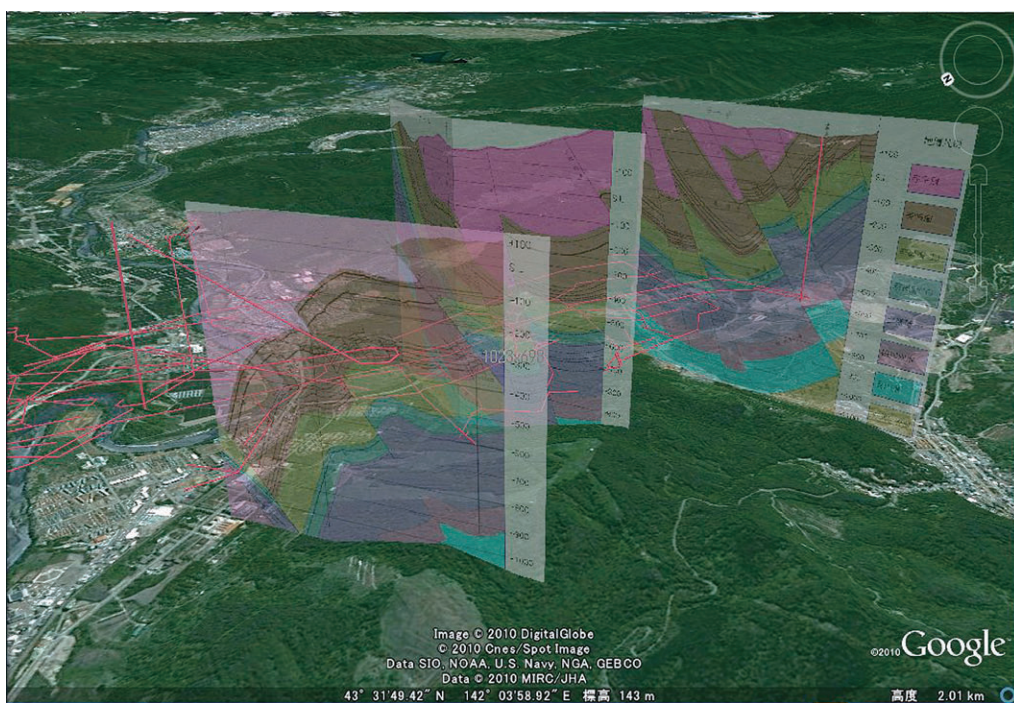


図-8 住友赤平炭鉱 (1200mリフトアップ)

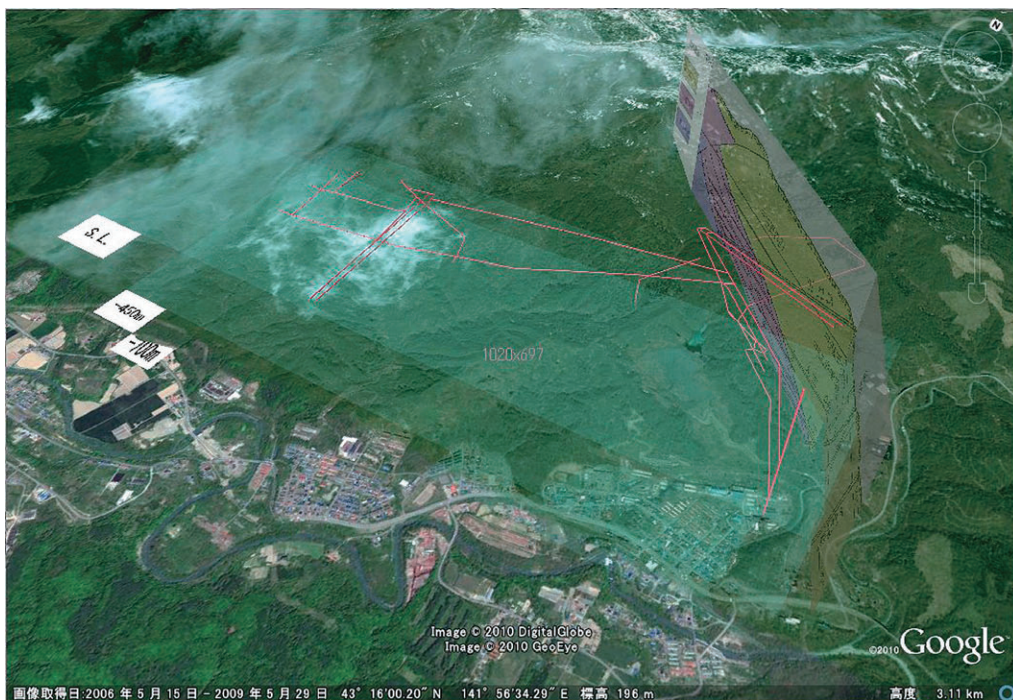


図-9 住友奔別炭鉱 (700mリフトアップ)

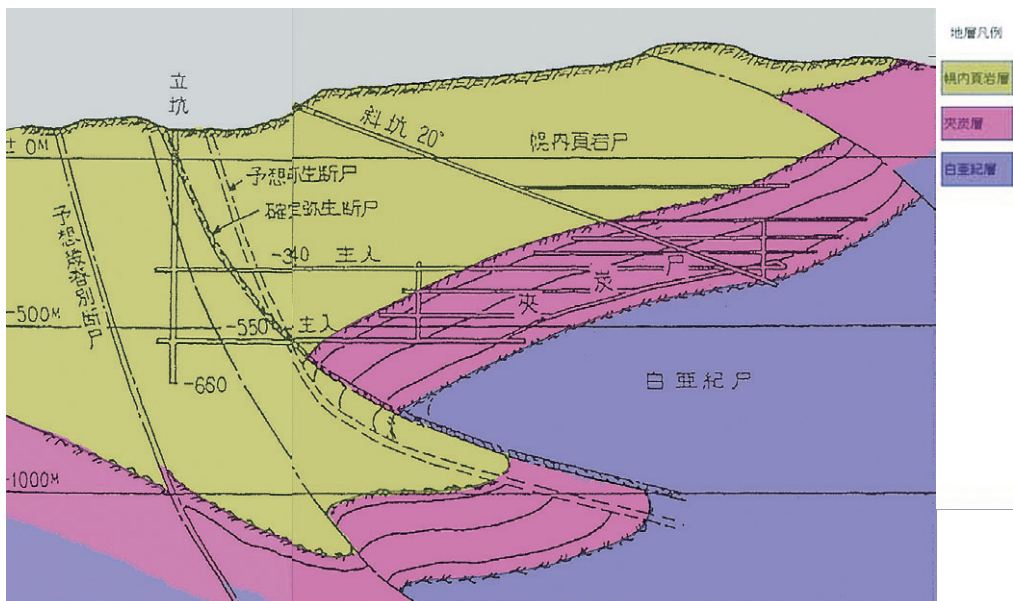


図-10 住友奔別炭鉱地層断面図

## 参考文献

- 1) Google SketchUpダウンロード先：<http://sketchup.google.com/intl/ja/index.html>
- 2) Google SketchUpユーザーズガイド：<http://sketchup.google.com/support/bin/answer.py?answer=116174>
- 3) David Flanagan, まつもとゆきひろ：プログラミング言語Ruby, オライリー・ジャパン, 2009. 1.
- 4) 登録サイトの一例：[http://rhin.crai.archi.fr/RubyLibraryDepot/plugin\\_list\\_cat.php?cat=3](http://rhin.crai.archi.fr/RubyLibraryDepot/plugin_list_cat.php?cat=3)
- 5) Rubyダウンロード先：<http://pub.cozmixng.org/~the-rwiki/rw-cgi.rb?cmd=view;name=Ruby+Install+Guide>
- 6) RDEダウンロード先：<http://www.vector.co.jp/soft/dl/win95/prog/se209196.html>
- 7) Matthew Scarpino : Automatic SketchUp Creating 3D Models in Ruby, Eclipse ENGINEERING LLC, 2010.4.
- 8) 住友石炭鉱業株式会社：立坑特集号（その一）赤平立坑工事記録，井華技報，昭和36年11月。
- 9) 住友石炭鉱業株式会社：立坑特集号（其の一）奔別立坑工事記録，井華技報，昭和35年10月。

## 坑道モデル作成スクリプト例

---

```
# Ponbetu_CircleDraw_coalmine_path.rb
# read data into Ruby script from Matlab data file
st_file_no=1#<-----
end_file_no=21#<-----
if st_file_no>1then
  Sketchup.open_file "c : /SketchUp_model/Ponbetu_coalmine_road.skp"
end
for no in st_file_no..end_file_no
  s1="c : /MATLAB6p5/work/2010_coalmine/Ponbetu/Ponbetu_road"
  s2=no.to_s
  s3=".dat"
  s=s1+s2+s3
  f=File.open(s,"r")
  data=f.readlines
  f.close
  print k=data.length
  print("¥n")#new line
  ax=[] # initialize for array variable
  ay=[]
  az=[]
  model = Sketchup.active_model
  entities = model.active_entities
  coe=10000.0/254.0#multiple coefficient for unit adjusting(inch <- m)
  for i in 1..(k-1)
    st=data[i].split(',')#separate array at the position of ','
    st[2].chomp('/n')#delete /n
    ax[i]=st[0].to_f # .to_f : numeric <--- string
    ay[i]=st[1].to_f
    az[i]=st[2].to_f # adding the negative signe in the case of mirroring
    #print ax[i],ay[i],az[i]
```

```

ax[i]=ax[i]*coe # inch <- .m can't use .m for array
ay[i]=ay[i]*coe
az[i]=az[i]*coe
# Draw a construction line that points diagonally.
if i>=2 then
  s1=s1+[[ax[i],ay[i],az[i]]]
else
  s1=[[ax[i],ay[i],az[i]]]
end
end
path=entities.add_curve s1
# Draw face section of coal_road
a=ax[2]-ax[1]
b=ay[2]-ay[1]
include Math
leng=sqrt(a**2+b**2)
c=a/leng
s=b/leng
if no<21
  radius=2.6
  r=radius.m # inch <- m
  circle_road=entities.add_circle [ax[1],ay[1],az[1]],[c,s,0],r
else
  radius=4.0
  r=radius.m # inch <- m
  circle_road=entities.add_circle [ax[1],ay[1],az[1]],[0,0,-1],r
end
coalmine_circle_face=Sketchup.active_model.entities.add_face circle_road
coalmine_circle_face.followme path
#
#Save the design and open it in SketchUp
Sketchup.active_model.save "c : /SketchUp_model/Ponbetu_coalmine_road.skp" # <-----
Sketchup.open_file "c : /SketchUp_model/Ponbetu_coalmine_road.skp" # <-----
end
# Remove the path
entities.erase_entities path
[EOF]

```