

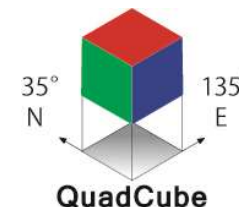
Dehenkenの近況と、 ロボットや自動車等の自動運転を実現する 「クアッドキューブ全位置検索技術」のご紹介

畑中 豊司
クアッドキューブ株式会社
代表取締役

2018年5月19日 @OSC Nagoya

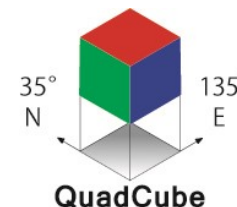
会社概要 (About us)

商号	株式会社データ変換研究所	英語表記) Dehenken Limited
資本金	2,000万円	
創業・設立 目的	1999年9月2日(平成11年) 1. ソフトウェア業 2. 前号に付帯する一切の事業	
本社所在地	〒604-8155 (最寄駅 四条烏丸 徒歩2分) 京都市中京区錦小路通室町東入占出山町308 ヤマチュウビル1F TEL 075-254-8780 (FAX 075-254-8790) E-mail info_ml@dehenken.co.jp	



「 Dehenken 祇園祭 2018 」
今年は後祭り、7月23日(月) 18時-22時

Dehenkenの主力商品 (1999年創業より)



デ変研TF / MFXライブラリ (コア・コンピタンス)

MS-Office、PDF、一太郎などのファイルフォーマットからテキスト情報を抽出する変換ソフトウェアで、当社のコアコンピタンスです。MFXはメールとファイル展開。

Mail Cyclope (メールサイクロップ)

メールの全文検索ソフトウェアです。メールに添付された文書や圧縮ファイルも展開し内容が漏れなく検索でき、メールセキュリティの会社にOEM提供をしています。

デ変研監査ライブラリ

個人情報の含まれるファイルを検出するソフトウェアです。ITセキュリティのため保護すべきファイルを探し出す用途に利用します。多くの情報セキュリティ会社にOEM提供しています。本年はマイナンバーの検出に対応し好評です。

デ変研MMライブラリ (新商品)

マクロ無害化のために、マクロを取り除くライブラリ。HP とプレスリリースはまだですが、OEM客先に好評。

Mail Filter / Archive / Forensic

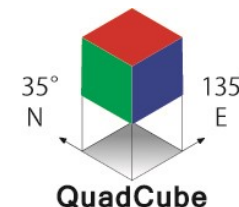


Search System / Group Ware



会社概要 (About us)

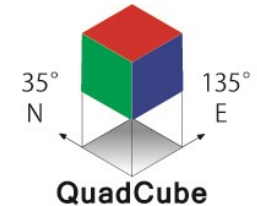
商号	クアッドキューブ株式会社
資本金	100万円 (株主 株式会社データ変換研究所 100%)
創業・設立	2017年2月3日 (平成29年)
役員	代表取締役 畑中豊司 (株式会社データ変換研究所 代表取締役)
本社所在地	〒144-0052 (最寄り駅 JR蒲田駅、京急蒲田駅 徒歩6分) 東京都大田区蒲田4-42-3イースタンコーポ蒲田201 Biz Kamata
TEL	03-6715-8404
FAX	03-6715-8405
E-mail	quadcube[at]dehenken.co.jp [at]を@に置換え
URL	http://www.derimo.net/



東京進出お考えの方用。登記可。
優良シェアオフィス物件 有。



自己紹介



代表者略歴

代表取締役 畑中豊司(1961年4月30日生 大阪生まれ)

在住:メイン 京都府長岡京市。サブ 東京都大田区大森町(月7)。

1984年 3月

滋賀大学教育学部 卒業 (教育心理)

1984年 4月

立石ソフトウェア株式会社 入社(現オムロンソフトウェア)。

1999年 3月

オムロンソフトウェア株式会社 退社(勤続15年)

1999年 9月

有限会社データ変換研究所 創業。代表に就任中(勤続18年超え)

2017年 2月

クアッドキューブ株式会社 設立。代表を兼務。



つくばチャレンジの参加ロボット。
左は筑波大学・右は明治大学

つくばチャレンジ歴

2011年 1月

ArduinoをUSBでAndroid に接続するM2M環境。Derimo 開発開始

2013年11月

つくばチャレンジ2013 では 大阪工業大学チームとして参加。

2014年 7月

つくばチャレンジ実行委員を拝命。モニタリング実施を表明。

2014年11月

Derimo TC2014 にて、23 ロボットの登録。位置と方向を表示。

2015年11月

Derimo TC2015 にて、14 ロボットの登録。

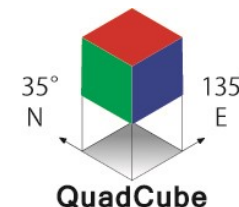
2016年11月

Derimo TC2016 にて、13 ロボットの登録。

2017年11月

Derimo TC2017 にて、16 ロボットの登録。

(つくばチャレンジ2018 本走行 11/11、実験走行 6/30 8/4 9/15 10/13 10/14 11/4 11/9-10)



つくばチャレンジの動画

「つくばチャレンジ」で検索 → www.tsukubachallenge.jp
ホームページの右上、過去の記録のうち「2017」を選択

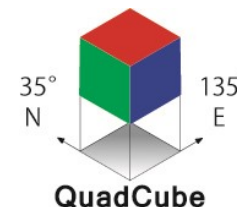


Youtubeの動画を再生



※ Youtubeにて、「つくばチャレンジ」を検索すると、沢山の動画が検索できます。

簡易なもの見え方表現



座学からはいるのではなく、
現地を見ていろいろと考えました。
そこから、独自の考案を試みています。

簡易な構造物の定義

クアッドキューブ
(Quadcube)

3Dのための、視点の位置

VPI
(Viewpoint position information)

3Dビューとしてみるための格子座標

LVA
(Lattice View Angle)

※ QCの動画

クアッドキューブとは

クアッドキューブ (QuadCube) は、

緯度経度に平行な占有格子の四辺形 (LLQuad) に

高さの情報を加え

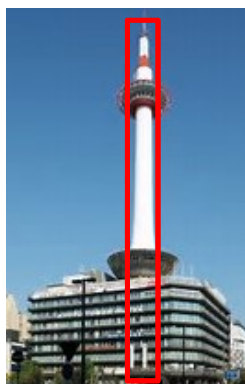
立方体の表現したものをいいます。

例えば京都タワー(高さ 131m)であれば

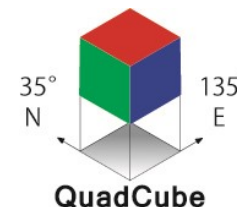
Q4,135.7594, 34.9876,0,0,131

リアルな地上の構造物を

軽量のCSVデータで表現します。



LLQuadとは



LLQuadは、

Latitude longitude quadrilateralの略語で、

緯度経度四辺形といいます。

緯度と経度と平行な線で占有格子地図の

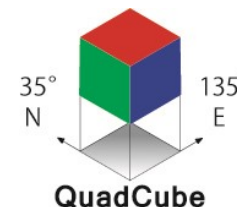
縦横を囲んで面を表現します。

1つの格子の大きさは、緯度経度の度数の小数点以下の数により、右のように、7種類のタイプに分けています。

「Q4」は、LLQuad10mを示します。

LLQuadタイプ	略称	対応する地球の度数
LLQuad10km	Q1	0.1
LLQuad1km	Q2	0.01
LLQuad100m	Q3	0.001
LLQuad10m	Q4	0.0001
LLQuad1m	Q5	0.00001
LLQuad10cm	Q6	0.000001
LLQuad1cm	Q7	0.0000001

VPI（視点位置）



VPI(視点位置情報:
Viewpoint position information

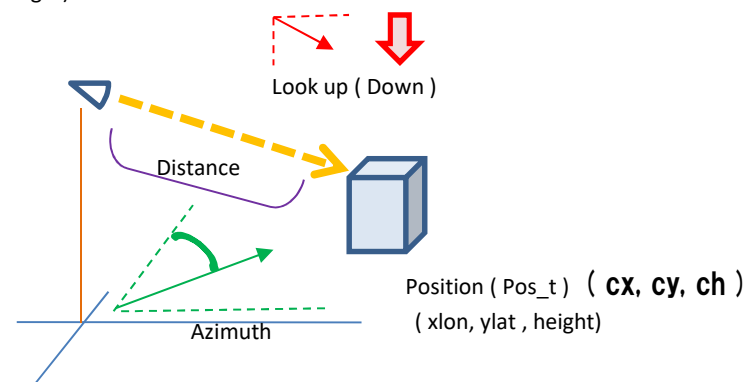
モニタリングの目の位置を示し

位置情報(xlon, ylat)、
hi(height: 高さ)、
az(azimuth: 方位角)と
lu(look up: 仰俯角)

で構成されます。

Position (Pos_t) (**cx, cy, ch**)
(xlon, ylat , height)

Looking (Ald_t) (**az, lu, ds**)
(azimuth, lookup, distance)



Pos_t 位置情報構造体

```
typedef struct pos {  
    double   cx;   center x pos 経度。Xlon。QCの場合、経度中心点。  
    double   cy;   center y pos 緯度。Ylat。QCの場合、緯度中心点。  
    double   ch;   center height 高さ。QCの場合、立方体の中心高。  
} pos_t;
```

Ald_t 立方体の方位・仰角・距離

```
typedef struct ald {  
    double   az;   (azimuth) 立方体の方位に傾く角度。0は東。  
    double   lu;   (look-up) 立方体の上下に傾く角度。0水平。下-90度。  
    double   ds;   (distance) 視点と立方体との距離。単位はm。  
} ald_t;
```

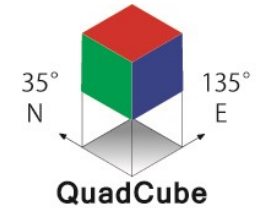
オブジェクト構造体

VPIオブジェクト

```
typedef struct vpi {  
    int      flag; /* View point information */  
    pos_t    pos;  VPI用の指定文字をVF / VTビットにして格納する  
    ald_t    ald;  絶対緯度経度、もしくはオブジェクト相対位置  
    int      id;   絶対方位仰角、もしくはオブジェクト相対位置。  
                VPIのID指定。CSVで格納位置指定しAPIで活用。  
} vpi_t;
```

C言語 構造体の説明

LVAとは

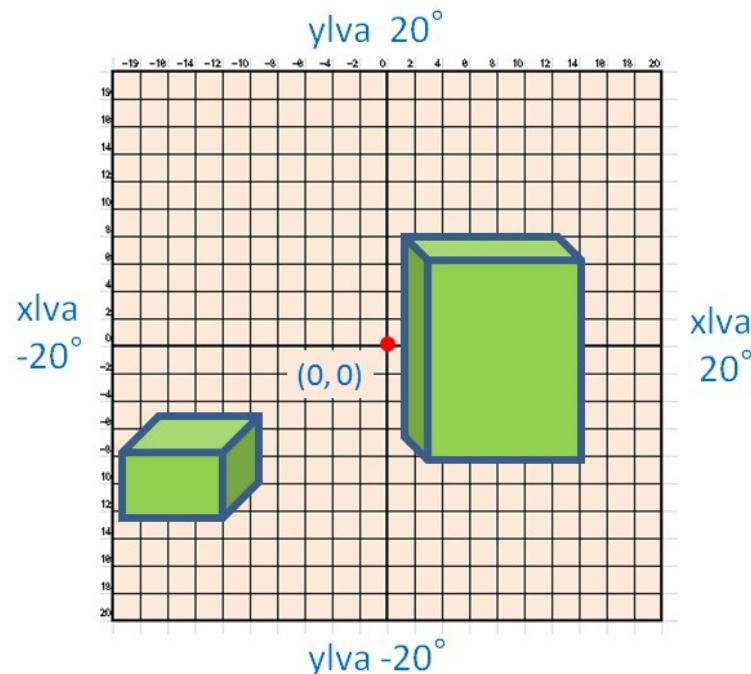


格子窓座標 (LVA: Lattice View Angle Coordinates)

格子窓座標は、格子窓(障子(しょうじ)の棧(さん))を座標位置に見立て、
視角 ± 20 度の上下左右の範囲を、描画位置 ± 20 の座標に描画させる座標系。

座標の(xlva, ylva)の1.0 LVAは視角1.0度と一致。

LVA の中点(0, 0)は、VPIで指定した視点方向と一致。



C言語 構造体の説明

Lva_t 格子窓座標へのプロット位置

typedef struct lva { 格子窓座標へのプロット位置

double x; 横軸(x)への座標点。±20度で、中心は0。

double y; 縦軸(y)への座標点。±20度で、中心は0。

} lva_t;

1つの点オブジェクト

typedef struct point {

int p_flag; 立方体の1つの点の情報構造体

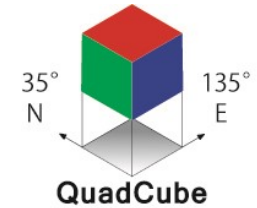
pos_t p_pos; lva(±20度)からの、はみだし情報(20度内外)

ald_t p_ald; 位置情報(xlon, ylat)

lva_t p_lva; VPIから、この1つの点の az lookup distance

} point_t; LVA座標点の x,y の位置情報

3Dの点を結ぶ、基本原理



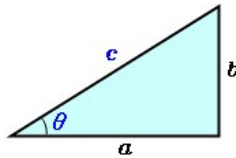
点v : 見る点 (VPI)

点q : 見られる点 (クアッドキューブ)

- (1) 点vと点qの距離を計算
平面の2点間の距離

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- (2) さらに線vqと高さとの距離を求める。

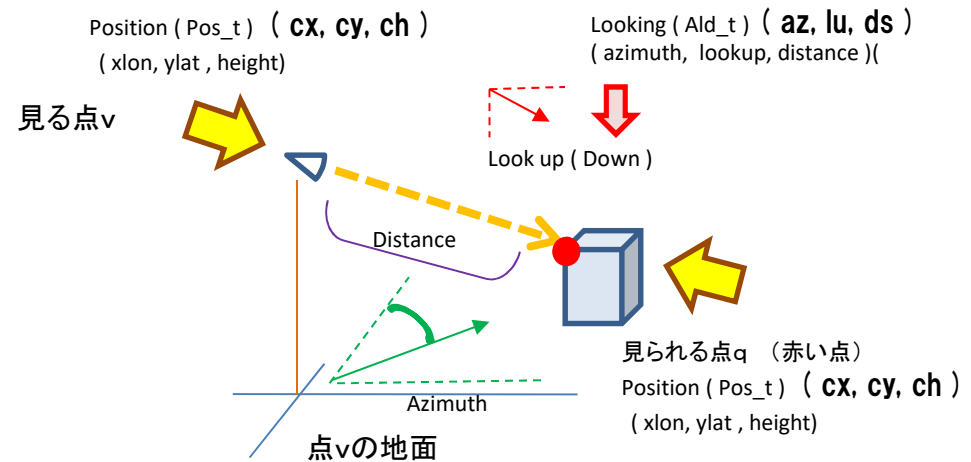


$$c = \sqrt{a^2 + b^2}$$

- (3) Azimuthの求め方
東0度、北90度とする点vから点qの角度

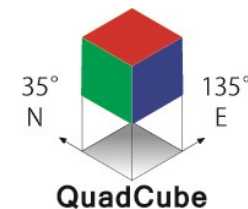
- (4) Lookupの求め方
直角三角形のθの角度
(ただし下向きなので負の値)

$$\theta = \tan^{-1} \frac{b}{a}$$

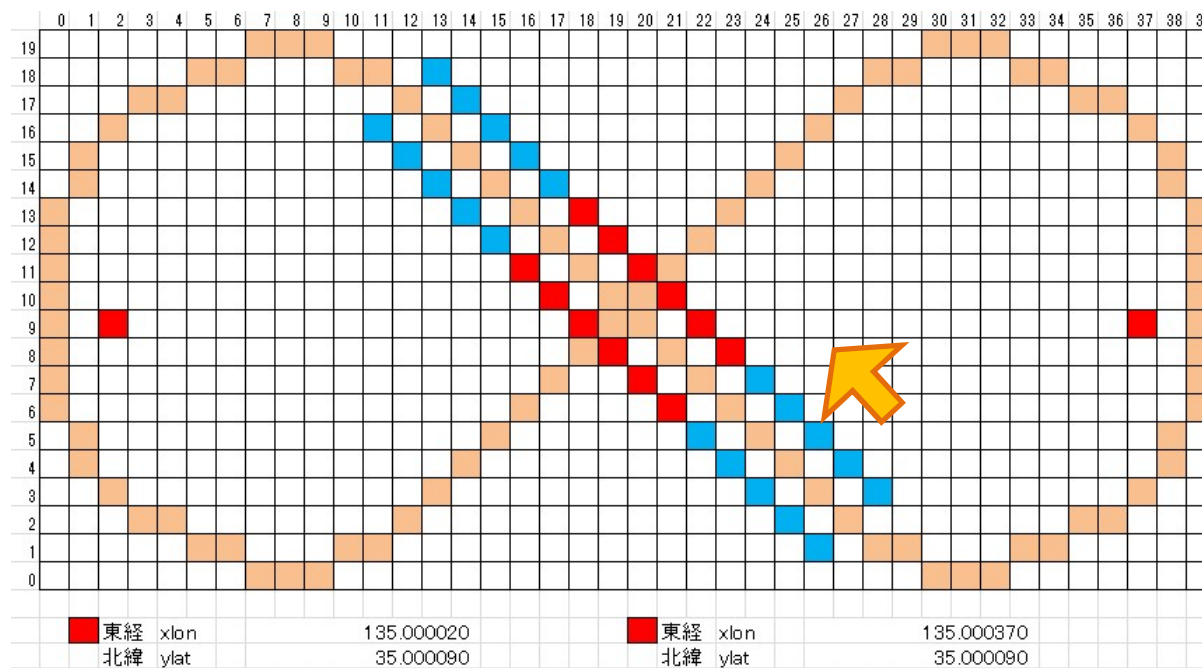


これらに加えて、
緯度により経線の長さが違う
補正を加えています。
ただし、地球を真の球体として。

京都駅東経135.7594度、北緯34.9876度	11.1m × 9.0m
札幌駅東経141.3507度、北緯43.0687度	11.1m × 8.1m



8の字コースによる、 VPIの番号指定のデモ



上にのぼり、
やがてくだる。

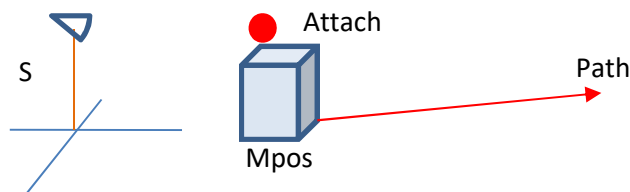
VPI (視点位置) の定義

VPIのカスタマイズ位置のデータ 01_conf.cf ファイルで設定

10 個の視点 ID を登録できます。

視点の位置は、
「 どの位置から、どこを見る 」ということを決めます。

- S 固定位置 (Static)
- M 移動体デバイスの位置 (Mpos)
- A 移動体に装着した補助装置 (Attachment)
- P パス (Path) 一時的な走行目標点



SS目線

固定の位置から、固定の位置を見る

SM目線

固定の緯度経度位置から
指定された移動体を見る

SP目線

固定の位置から現在Pathパス位置を見る

.....

01_conf.cf

.....

#!CF 1.0

Dehenken Derimo Series : Configuration file

(C) Copyright Dehenken Limited 2016-2018.

All rights reserved.

#

vpi : VP, num, +xlon, +ylat, +hi, +az, al, str

#

SM, 0, 135.000200, 35.000100, 10, 90, -90, Static Mpos (To)

MP, 1, 0.000000, -0.001000, 2, 0, 0, Mpos (From) Path (To)

AM, 2, 1.20, 10.00, 0.5, 0, 0, Relative Mpos (To)

AM, 3, 0.20, 1.00, 0.5, 0, 0, Relative Mpos (To)

SS, 4, 135.000200, 35.000100, 40, 90, -90, Static North

SM, 5, 135.000200, 35.000100, 40, 90, -90, Static Mpos (To)

SP, 6, 135.000200, 35.000100, 40, 90, -90, Static Path (To)

PM, 7, 0.000000, -0.001000, 2, 0, 0, Path (From) Mpos (To)

MP, 8, 0.000000, -0.001000, 2, 0, 0, Mpos (From) Path (To)

MM, 9, 0.000000, 0.000000, 20, 0, -90, Mpos (From) To) (Navi)

MP目線

移動体 (Mpos) からPath (パス) を見る

PM目線

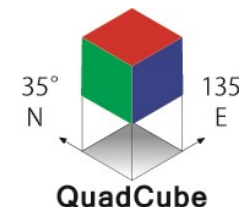
Path (パス) から移動体 (Mpos) を見る

AM目線

Mposにある移動体の位置から
移動体の筐体の装着物からの視線方向

MM目線

移動体 (Mpos) の進行方向 (Navi目線)



Soda (操舵) データとは

Sodaデータは、

S6、緯度経度、路面の高さ、路面の幅、制限速度を記述。

移動体の移動走行時の経路を緯度経度で記述する ウェイポイント (Way Point) といわれるものです。

緯度経度は、xlon, ylat で経度緯度を指定。

路面の高さは、地上高。

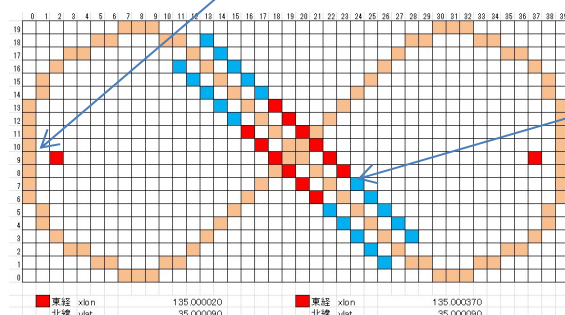
0の場合、地上表面からの高さ0となる。

路面の幅は、mにて、走行路面の幅(長さ)を記述。

0の場合、移動体の幅の2倍を設定。

制限速度は、時速表現にて、そのソーダ点を通過するときの制限時速を記述。

0の場合、その移動体の制限速度が適用されます。



.....

04_soda.sd

.....

#!SD 1.0

Dehenken Derimo Series : LLSoda data

(C) Copyright Dehenken Limited 2016-2018.

All rights reserved.

#

#

Tl,8 char racing course

#

S6,xlon,ylst,grand h,bottom h, speed,char

#

S6,135.00000,35.00010,0,0,0,a

S6,135.00000,35.00006,0,0,0,a

S6,135.00001,35.00004,0,0,0,a

S6,135.00007,35.00000,0,0,0,a

S6,135.00010,35.00000,0,0,0,a

S6,135.00029,35.00019,0,0,0,a

S6,135.00032,35.00019,0,0,0,a

S6,135.00038,35.00015,0,0,0,a

S6,135.00039,35.00013,0,0,0,a

S6,135.00039,35.00006,0,0,0,a

S6,135.00038,35.00001,0,0,0,a

S6,135.00032,35.00000,0,0,0,a

S6,135.00029,35.00000,0,0,0,a

S6,135.00023,35.00006,0,5,0,a

S6,135.00016,35.00013,0,5,0,a

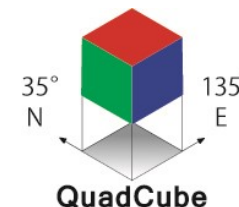
S6,135.00009,35.00019,0,0,0,a

S6,135.00007,35.00019,0,0,0,a

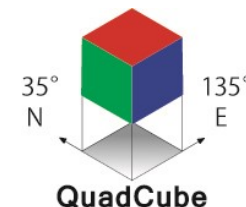
S6,135.00001,35.00015,0,0,0,a

S6,135.00000,35.00013,0,0,0,a

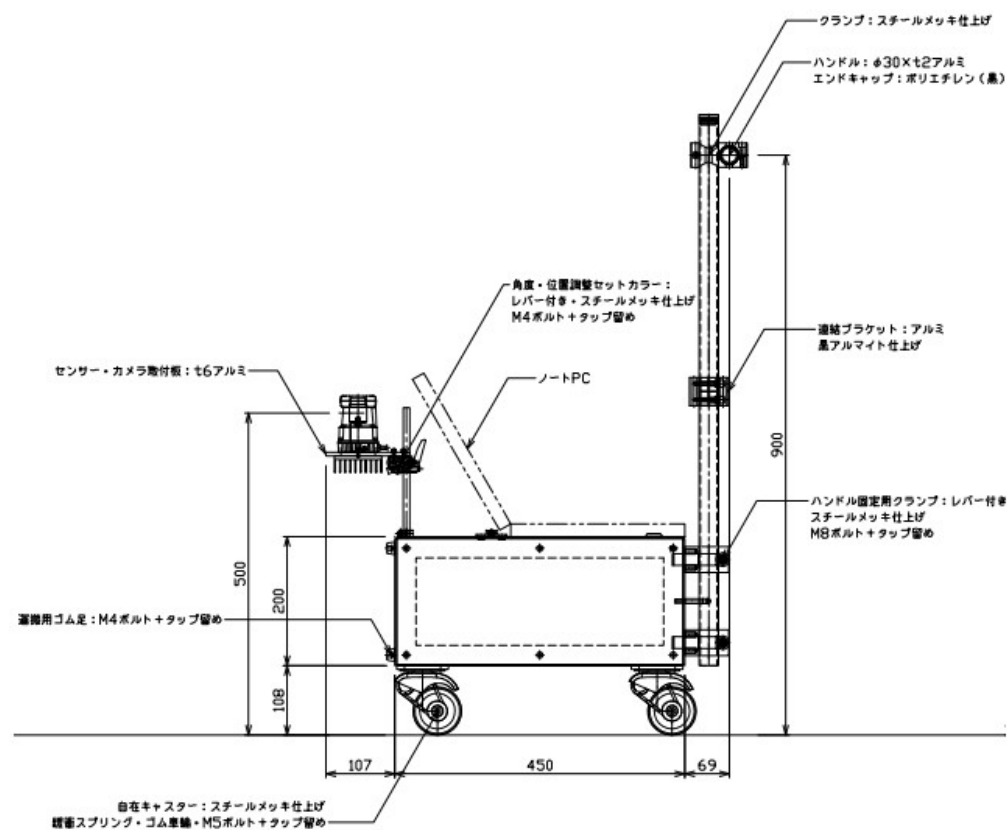
SL



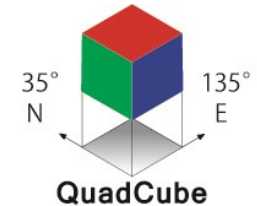
空間計測のための、走行車両



Derimo U-1



測域センサ



UTM-30LX-EW 外観



UTM-30LX-EW 特長

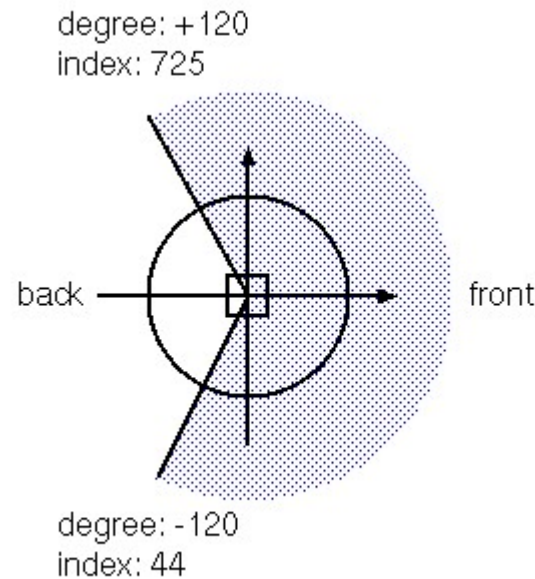
北陽電機 スキャナ式レンジセンサ

30m、270° の広範囲スキャンを実現。長距離、高速応答で、サービスロボットの高速走行に効果的です。

<特長>

- 長距離！ 検出距離: 30m
- 広視野！ 270° (±135° 1080点)
- 屋外対応！ (距離が短くなる場合があります)
- 小型軽量！ W60×D60×H87mm、370g

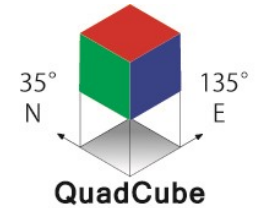
Classic-URGのデータ取得範囲



Classic-URG のデータ取得範囲

UTM-30LX-EW 仕様

電源電圧	DC12V±10%
光源	半導体レーザー λ=905nm
測距範囲	定格: 0.1~30m、270°
測距精度	0.1~10m: ±30mm 10~30m: ±50mm1
角度分解能	ステップ角: 0.25° (360° /1,440分割)
走査時間	25ms/scan
騒音	25dB以下
インターフェース	USB2.0(Full Speed)
専用コマンド	SCIP Ver.2.0
温度・湿度	-10~+50°C、85% RH以下
耐振動	複振幅1.5mm、10~55Hz
耐衝撃	196m/s ² 、X・Y・Z各方向
質量	約370g(ケーブル含む)



SCIP2.0 (スキップ)プロトコル

送信データ
GD0044072501

受信データ例
GD0044072501
00P

ODK0>

00i00i00i00i00k00k00n01101101101101101
101100o00m00o00o0130130140]

14012012014015017017017016017016016
015015015014014014014015010



測行スキャナは、

PCからURGに、空間スキャン指示を送ります。

URGは、受信データを、PCに返します。

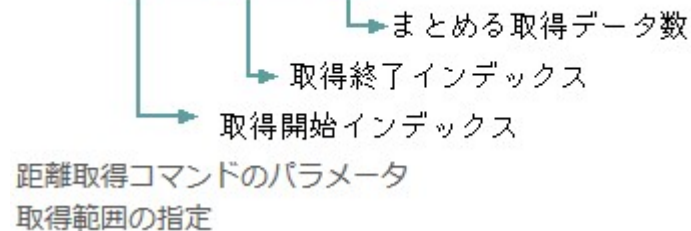
270° にわたり、
0.25° おきに計測された
1080個の距離データは、
デ変研QCライブラリのデータハンドラが処理し、

Q4 の緯度経度情報にすることができます。

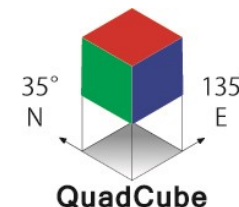
Q5 の緯度経度情報にすることができます。

これらを、**測域地図**とし、
クアッドキューブの風景地図データと、
することができます。(建物の高さは手入力で)

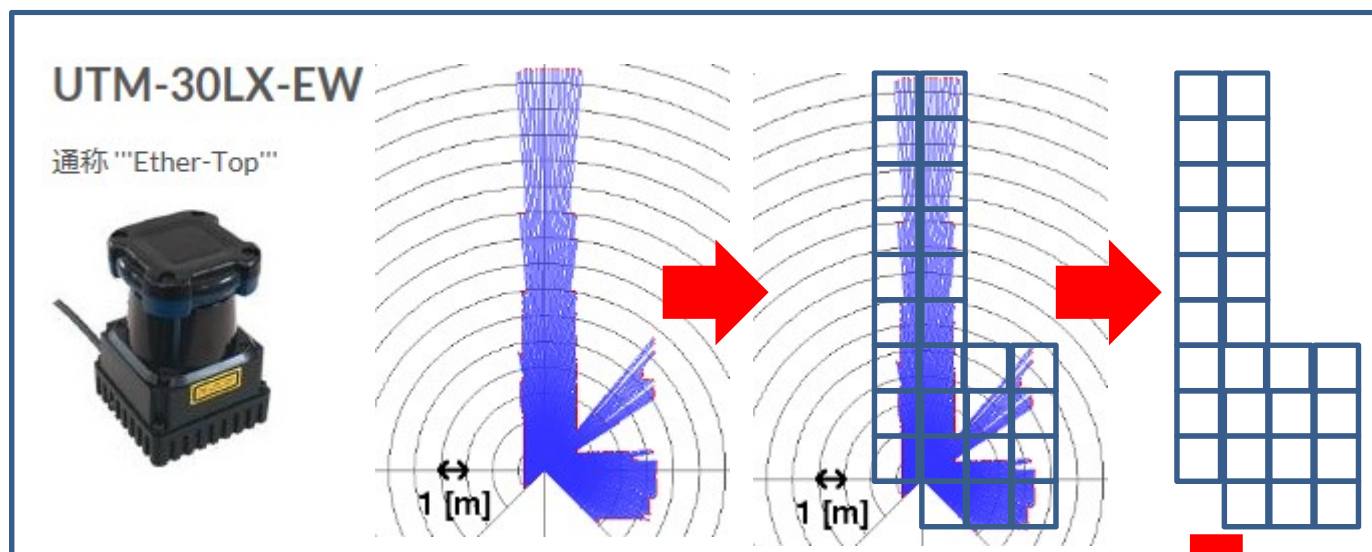
GD 0044 0725 01



本データの引用元 https://sourceforge.net/p/urgnetwork/wiki/scip_capture_jp/



測域センサによる 壁データの作成(QC化)

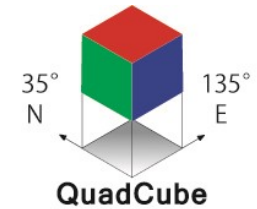


「デ変研QCライブラリ」で用いる地図には、地図からの生成方法と、測域センサを搭載した機器を用いる方法があります。

現地で空間を走行しながら測域し、QCデータで地図を作成することができます。

Q6,135.759331,34.987561,0,0,1,6
Q6,135.759331,34.987560,0,0,1,6
Q6,135.759334,34.987556,0,0,1,6
Q6,135.759335,34.987556,0,0,1,6
Q6,135.759334,34.987557,0,0,1,6

Sodaデータの粒度



データの粒度

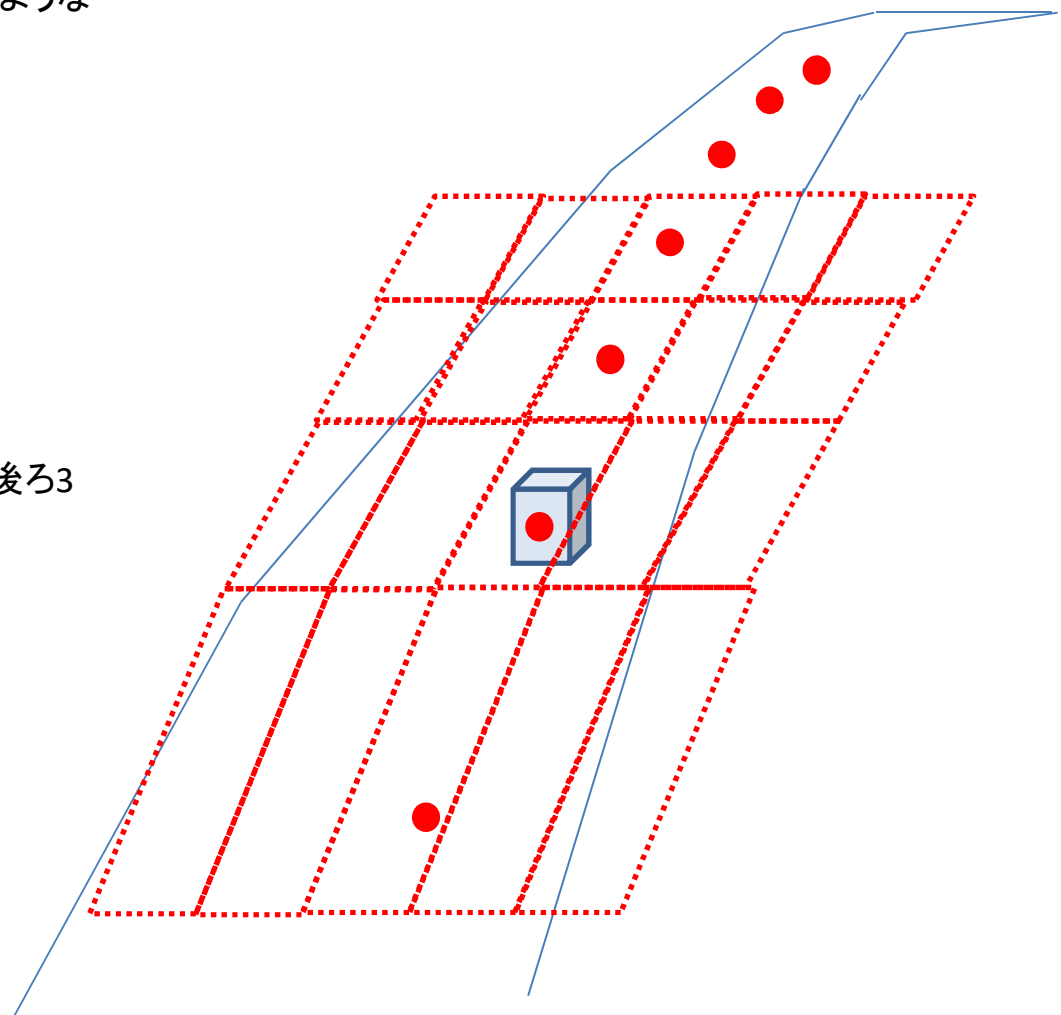
という言葉进行分析の現場ではよく使用しますが、ここでは、Sodaデータの粒度という表現で、キューブデータの個数を最低限必要な情報が得られ、計算に負荷を与えないようなリソースの保持について考えてみましょう。

東名高速道路の走行データの粒度

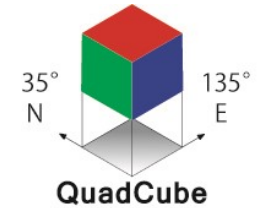
東京IC - 小牧IC間の道路名、346.8 km。
時速100kmの移動体が1秒間に進む距離の間隔を、27.78mとすると、単純計算で、12,484点のSoda点で片道の走行経路を描けます。

全位置検索時のヒット数

前後3秒の長さの範囲で全位置検索すると、前3個、後ろ3個程度のSodaデータが、常に検索ヒットします。



自己位置補正



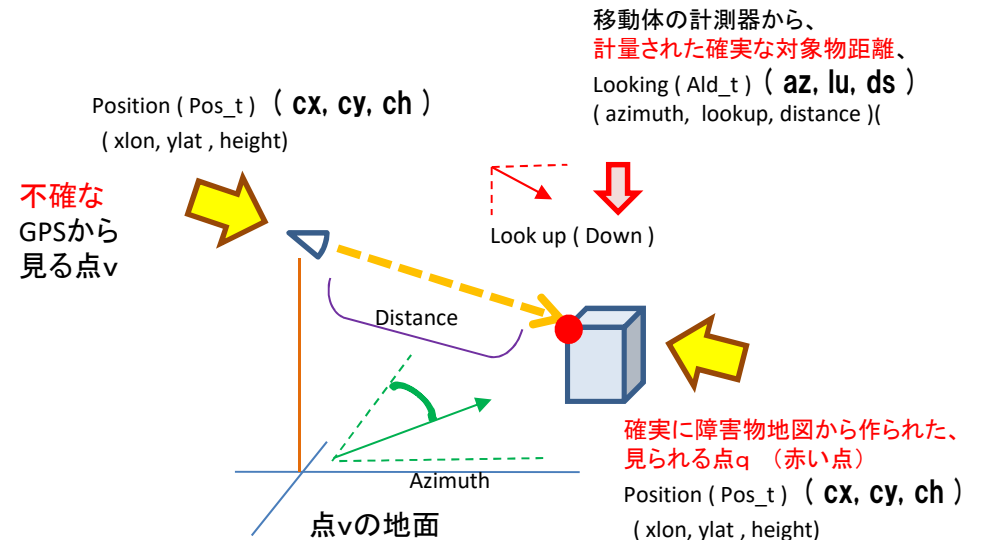
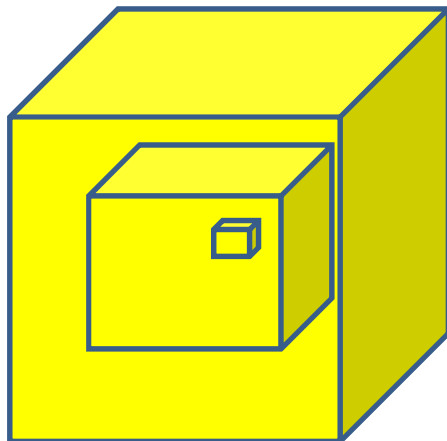
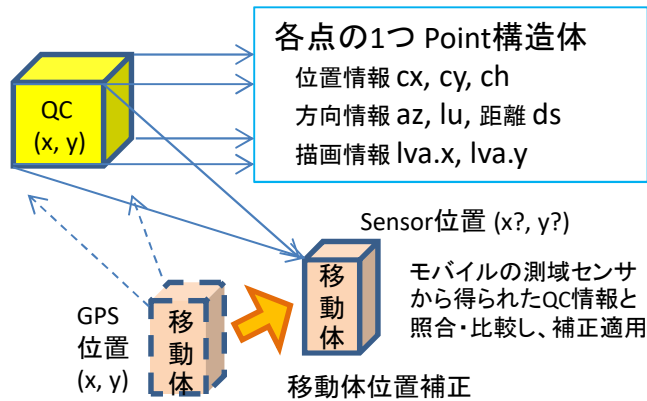
デ変研QCライブラリの

ソフトウェア環境が、ようやく、群化してきて、

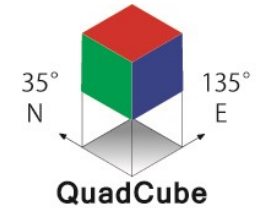
ソフトウェア群が、互いに関係が結び付き始めました。
最終的には、このQCのデータをつかって、
自己位置補正に役に立つことが、
2018年5月10日にDefenseのお客さんで出会い、
証明できました。

位置補正

QCの8つの頂点から得られる情報



QCデータの粒度



データの粒度

Sodaデータの粒度につづいて、
QCの構造物(壁)のデータにも触れます。

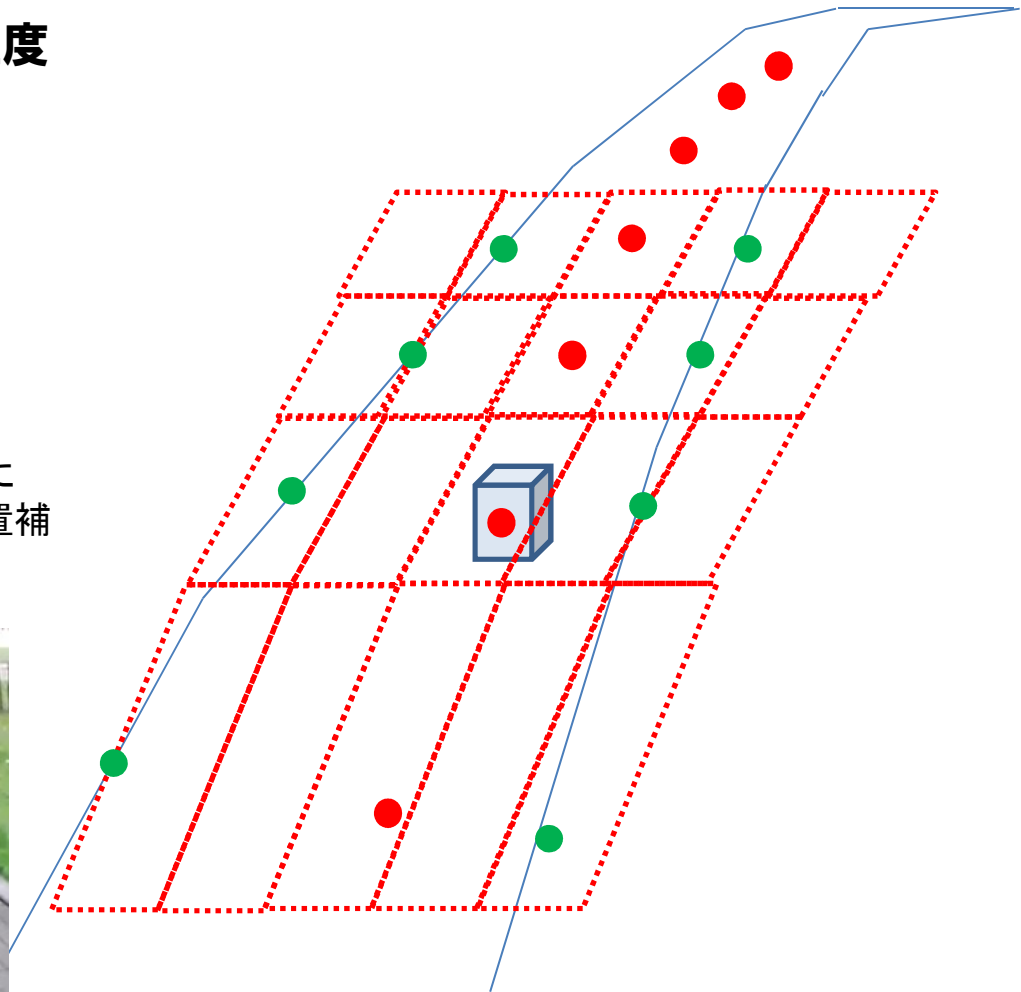
つくばチャレンジの走行時の壁面データの粒度

つくばチャレンジの2kmのコースをすべて、
測域センサを用いて、壁データをQCデータとして、
作成します。

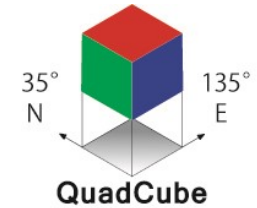
これを、1回の測域センサ作動に、
およそ8個のQCが得られるような、
空間内での検索粒度を設計します。

全位置検索時のヒット数

前後3秒の長さの範囲で全位置検索すると、前後左右に
8個のQCデータがヒットします。うち、3個のデータで位置補
正をします。



道路面の検出



最終的には

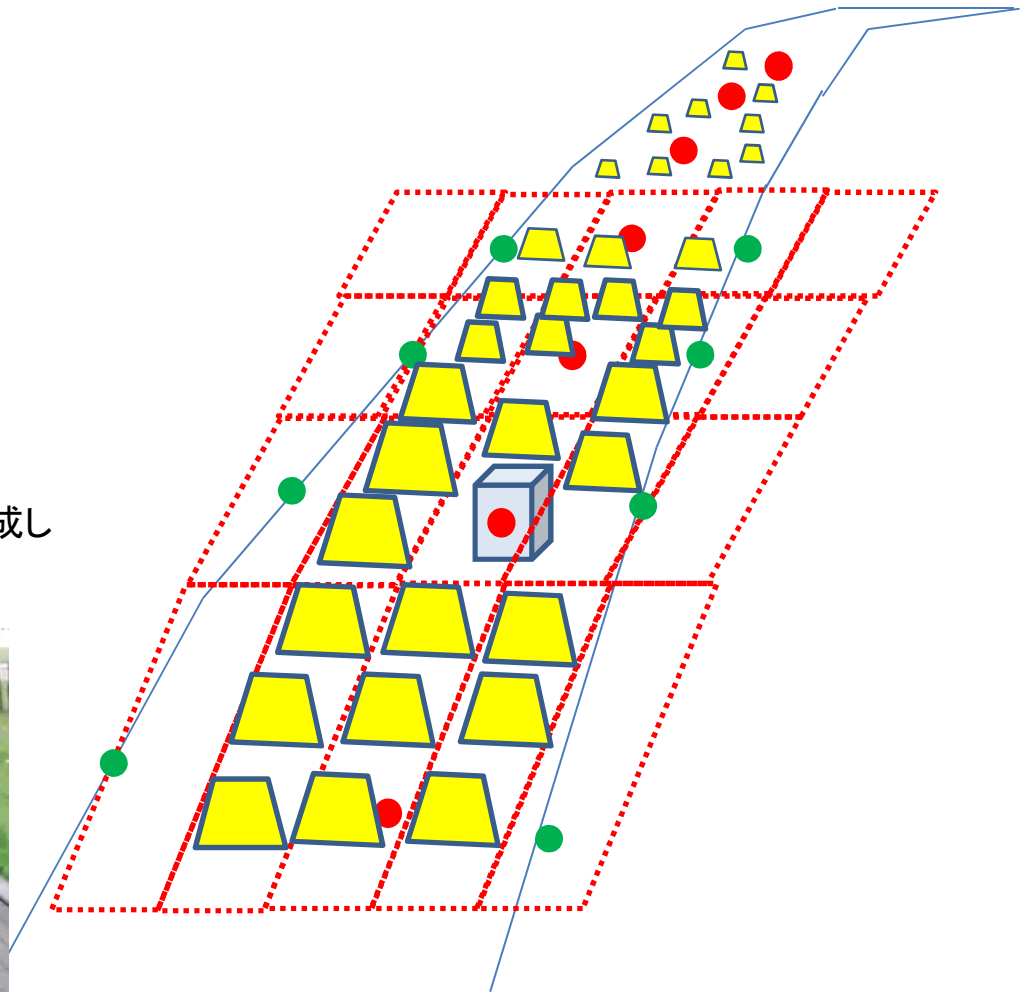
測域マシンを使って、緯度経度のついた、QCデータを用意すると、位置情報の補正ができる技術が確認されても、日本全国、計測してまわるわけには、いきません。

道路面を検出すること

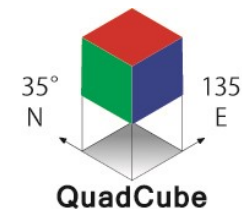
障害物があるということを検索して見つけることができますが、そこに至るまで空間は、障害物がないという空間です。

道路面の検出は空間検出

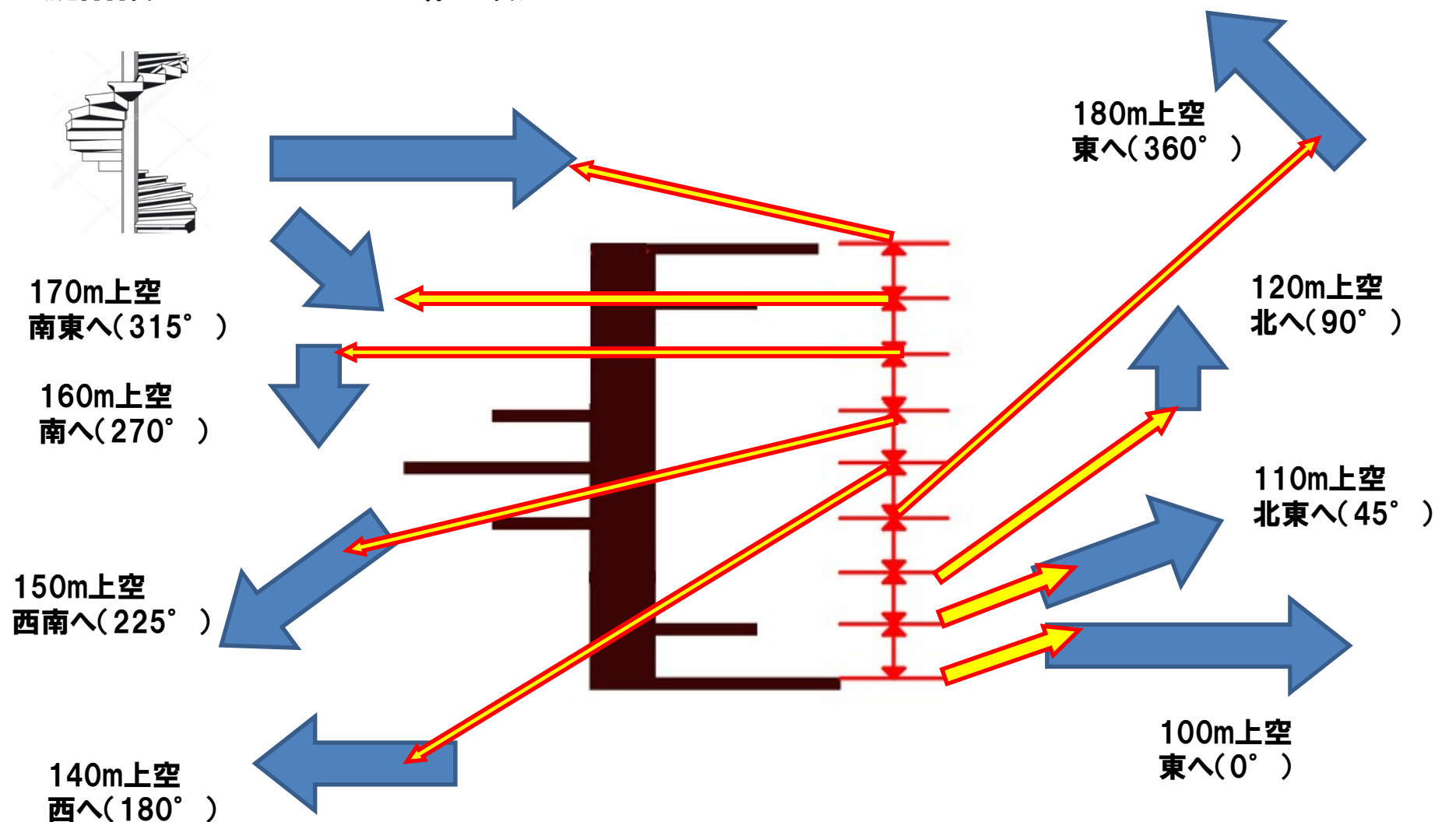
ある、ということから、ないということに着目することで、道路を走れる空間内を検出する技術により、地図を使った自動空間検出が完成します。



ドローンの航空域

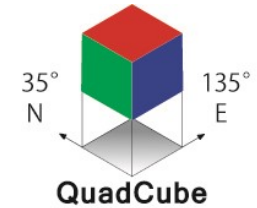


複数のドローンが安全に飛行できる螺旋階段をイメージした空域の設定



ご清聴ありがとうございました

Thank you for your listening.



教育サブスクリプション/OEMキット

ヤトロとは

▼ ハンバイ ▼ カイケツ ▼ テイアン

「ハンバイ」「カイケツ」「テイアン」の3本柱で、
お客さまの多様なニーズにお答えいたします。



販売

教育サブスクリプション/OEMキット PC用販売

ヤトロ電子株式会社

〒300-3257

茨城県つくば市筑穂1-11-8

TEL 029-864-4484 / FAX 029-864-4485

E-mail: info[at]yatoro.co.jp

[at] は@に置き換え

OEMキット販売



株式会社データ変換研究所（以下、Dehenken）は、
「テキスト処理技術に強みをもつソフトウェアOEM会社」です。

OEMキット販売

株式会社データ変換研究所

〒604-8155

京都市中京区錦小路通室町東入占出山町308 ヤマチュウビル1F

TEL 075-254-8780 / FAX 075-254-8790

E-mail: info_ml[at]dehenken.co.jp

[at] は@に置き換え

企画・市場開拓

クアッドキューブ株式会社

開発元

株式会社データ変換研究所