

Webとナビ

2013年度 オープンプロジェクト 演習
10D8104019I 山本 真史
12D8101003C 渡邊 千尋
12D8104033J 門間 雄亮

背景

最近では、自分が初めて行った場所でも、携帯などの端末機を便りに事前にその場所を調べなくなっている



問題提起

確かに、最近ではどこに居ても地図がすぐ表示できるが、平面上に出された地図ではまず自分の居るところの把握からはじまる。

次に、自分がどこにいるかがわかった後でも、そこに行くまでの経路を平面上の地図では正確に把握するのは難しい。

その場所までの経路しか表示されない。





先行事例

一般的な案内をしてくれるもの
(NAVITIMEなど)

今回自分達がイメージしたものに近いものでは



lifehacker

の「Map Fan eye」
(左図)が近いものになっている

提案

- 画面を分割し, 半分に現地の地図, 半分にストリートビューを表示して案内する.
- 経路施設を指定した上での道案内を行う.
ex) 後樂園工科大学からコンビニに立ち寄ってから水道橋駅に行きたい

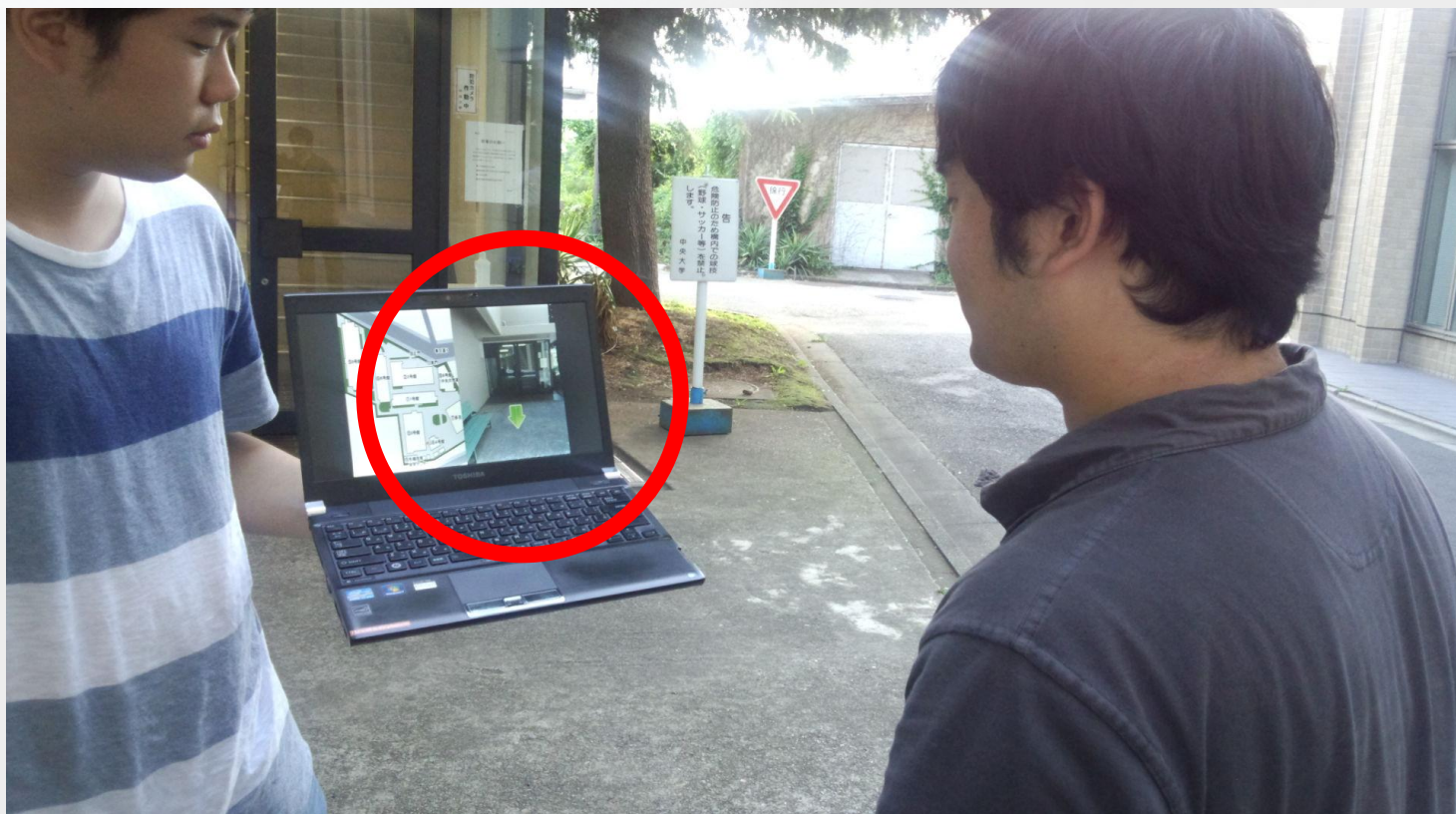


デモ

- 提案で作成したナビを実際に使ってみました。
- 4号館(サークル棟)から3号館の自販機に寄って、東門から帰るルートです。



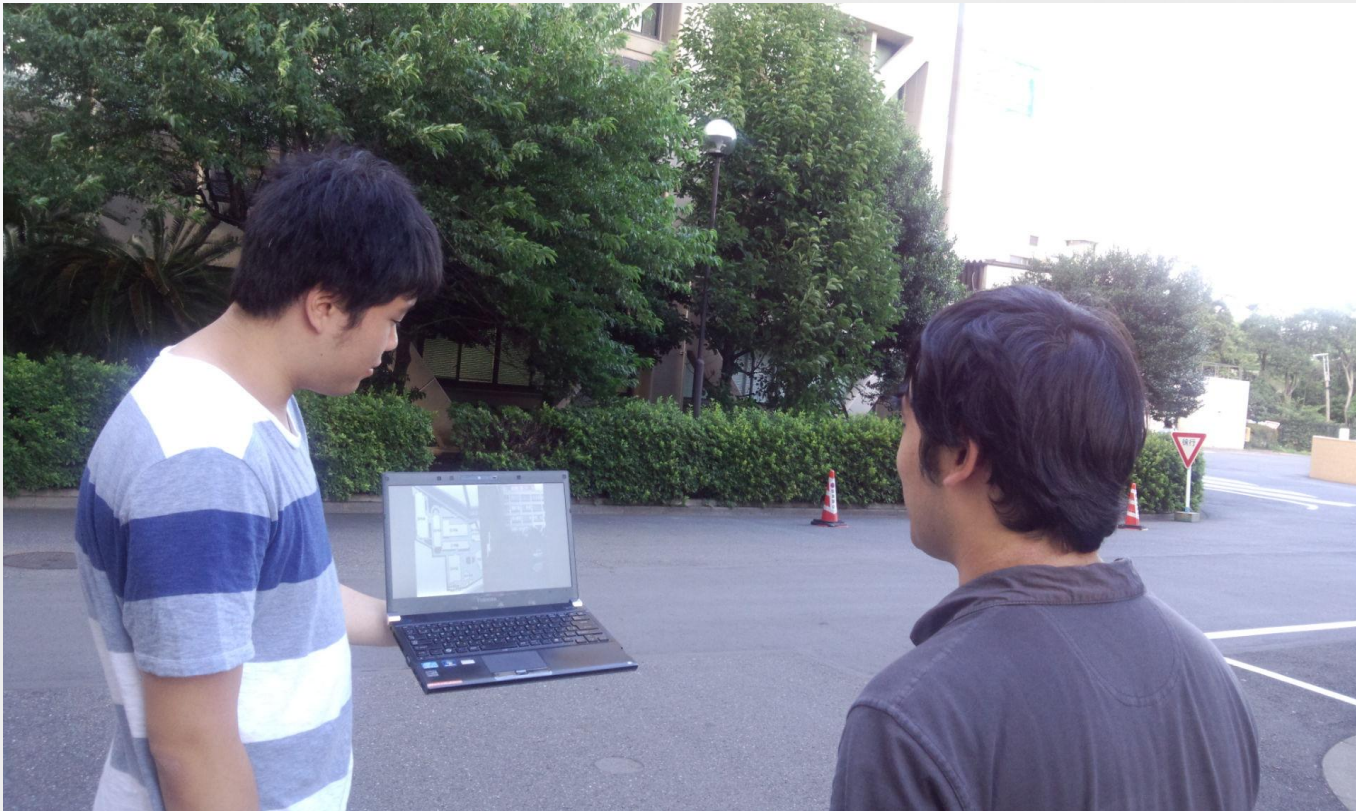
スタート地点



3号館入口



東門へのルート



東門



反省・改善点

- ◆建物内部の様子も取り組めばよかった。(3号館自販機
の位置など)
- ◆矢印と写真によって、道がわかりやすかった。
- ◆工事や季節により、背景が変わってしまうと、わかり
にくくなってしまうかもしれない。

デモ (.cpp)

- 各地点間の距離, 出発地, 経由地の種別, 目的地を入力すると, どこを経由して目的地に向かうのが最短かを出力してくれるものを作成.
- このようなものを計算の基幹部分として扱う...たぶん

```
masayama@masayama-virtual-machine:navi$ ./a.out
施設数 : 5
distance[0][1] = 1
distance[0][2] = 2
distance[0][3] = 3
distance[0][4] = 2
distance[1][2] = 1
distance[1][3] = 4
distance[1][4] = 3
distance[2][3] = 4
distance[2][4] = 3
distance[3][4] = 2
経由施設のある場所 (-1で終了) : 1 2 3
-1
開始位置 : 0
目的地 : 4
0 から 4 までは 1 経由が最短です.
```


デモ (.cpp) (一部)

```
#include <iostream>
#include <cstdio>
#include <set>

using namespace std;

const int ArraySize = 10;
// テニスコート, 1-6号館, 正門, 東門 = 9?

int main(void) {
    int n;
    double dist[ArraySize][ArraySize];
    int start=ArraySize*2, relay=ArraySize*2, end=ArraySize*2;
    set<int> relayS;

    // Input
    cout<<"施設数 : ";
    cin>>n;
    if(n>ArraySize) {
        cout<<"施設数は "<<ArraySize<<" 以下です."<<endl;
    }
    //cout<<n<<endl;

    for(int i=0; i<n; i++) {
        for(int j=i; j<n; j++) {
            if(i == j) {
                dist[i][j] = 0.0;
            }
            else {
                printf("distance[%d][%d] = ", i, j);
                cin>>dist[i][j];
                if(i!=j) {
                    dist[j][i] = dist[i][j];
                }
            }
        }
    }
}
```

```
// どこからどこまでどこ経由で
cout<<"経由施設のある場所 (-1で終了) : ";
int tmp;
cin>>tmp;
do {
    if(tmp < n) {
        relayS.insert(tmp);
    }
    else {
        cout<<"範囲外の施設番号です. 入力を無効とします."<<endl;
    }
    cin>>tmp;
} while(tmp>=0);
do {
    cout<<"開始位置 : ";
    cin>>start;
} while(start>n);
do {
    cout<<"目的地 : ";
    cin>>end;
} while(end>n);

// Calc
int cut = -10;
int distMin = 999999;
for(int i=0; i<ArraySize; i++) {
    if(relayS.find(i)!=relayS.end()) {
        if(dist[start][i]+dist[i][end] < distMin) {
            cut = i;
            distMin = dist[start][i]+dist[i][end];
        }
    }
}
```


展望

- GPSと連動して位置情報を自動取得し, 計算できるようになると実用性が高まる.
- 背景 (説明用) 写真をオープンで有志に更新して頂けると, 説明が行い易くなる.
- 寄り道をしない場合, すなわち “最短経路” も表示させるようにすると汎用性が高まる.



役割分担

- 山本cppコード, スライド (提案, デモ (.cpp), 展望, その他微調整)
- 渡邊 ... イメージppt (4号館→正門), スライド (デモ (ppt))
- 門間 ... イメージppt (4号館→裏門), スライド (背景, 問題提起, 先行事例)

終了

ご清聴ありがとうございました。

