

ビル街における歩行者のための3次元地図ビューア

A Three-dimensional Map Viewer for Walkers in High-rise Area

秦 祥彰†
Yoshiaki Hata

牧野 光則†
Mitsunori Makino

1 はじめに

地理情報システム (GIS) とコンピュータグラフィックス (CG) 技術の発展により、様々な地理情報を提供できるようになり、高精度な3次元地図や効率の良いナビゲーションシステムが開発されている。しかし、多数の建物が立ち並ぶ都市空間では、高い建物が後方の妨げとなり、3次元地図の通常表示では周囲の情報理解が難しい。また、現在、携帯電話などに搭載されている歩行者ナビゲーションシステムは、音声ガイダンスやGPS機能により目的地までのルート検索や現在地の把握は容易だが、買い物客など目的地が必ずしも明確でない歩行者の要求を満たせない。そこで、本稿では、ビル街における歩行者を対象とし、現在地の周囲の情報把握を容易にする3次元地図ビューアを提案する。提案システムでは、視点から可視な建物に対し、高さ調節、変形、半透明化などを行うことにより、建物表面上の情報を容易に認識させる。

2 定義

本稿では、ディスプレイ内に地図データを表示させる独立した小画面をウィンドウとし、地図情報として存在する個々の物体をオブジェクトとする。なお、ウィンドウ内に描画されているオブジェクトは、すべてユーザに認識されているものとみなし、周囲の情報把握の容易性をウィンドウ内のオブジェクトの個数で定義する。また、人が入れるオブジェクト (居宅、店舗、マンションなど) を建物とし、そうでないオブジェクト (木、電灯、ガードレールなど) を非建物とする。

3 提案手法

2で述べた周囲の情報把握の容易性を高めるために、建物の変形・消去を行う。このとき、出入口、窓ガラス、袖看板など建物の表面にある情報は、ただ名称を表すだけでなく、それぞれが持つイメージや特徴を我々に印象付け、歩行者が個々の建物を見分ける上で重要な役割を果たす。そこで、提案システムでは、建物表面の情報を維持するため、3.1で述べる手順に従って、スクリーン (座標空間内でウィンドウと対応する面) に近い建物から順に変形・消去を行う。また、非建物をこの操作を行う前に消去する機能を追加し、以下に述べるオブジェクトには含まない。

提案手法は、地図データの入力、ならびに視点・視線方向の決定の後に、処理対象となるオブジェクトの検出、描画処理を行う。以下、処理対象となるオブジェクトの検出と描画処理について説明する。

3.1 処理対象となるオブジェクトの検出

処理対象となるオブジェクトを以下の手順で検出する (図1参照)。

Step1: 地図データ内で接地面積最小の建物における接地面の対角線 (最遠の2点間) の距離を α [m] とし、ユーザが任意の値 n ($n \leq$ ウィンドウの横ピクセル数) を決める。

Step2: 現在地 (視点) からスクリーンを $n+1$ 等分する視線方向に n 本のベクトルを発生し、オブジェクトとの交点の有無を判定する。

Step3: Step2で発生したベクトルのうち、交点を持つ m 本 ($m \leq n$) 各々について、スクリーンに対する垂線の長さ (l_1, l_2, \dots, l_m) を求め、その最小値を $l_{\min} = \min_{1 \leq i \leq m} l_i$ とする。

Step4: $l_i \leq l_{\min} + \alpha$ を満たす交点を持つオブジェクトを有効オブジェクトとする。ただし、このとき、すでに有効オブジェクトとなっているものは、その性質を維持するものとする。

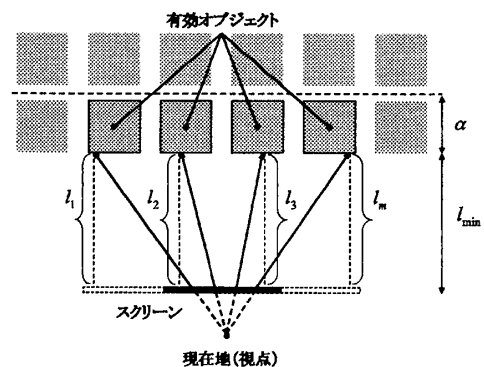


図1: 有効オブジェクト

3.2 描画処理

3.1で有効オブジェクトと判定されたすべての建物に対し (a) 半透明化, (b) 平面化を行なう。(a) は、建物を半透明化し、消去する操作, (b) は、建物を接地面と同じ平面に変形し、外枠を描く操作で、それぞれ任意の場所でのボタン操作によって行なわれる。なお、(a) はボタンを1度押すごとに半透明化と消去を交互に繰り返す。次に、上記の操作を行なう際、一部しか表示されない建物のうちユーザが選択する単一の建物に対して、(c) 縮小表示, (d) 下方移動の処理を行う。(c) は、建物を下方に一定の割合で縮小して表示する操作, (d) は、建物を下方に一定の割合で平行移動する操作で、それぞれ任意の場所でのボタン操作によって行なわれる。縮小率・移動量は、ユーザが指定できる。

† 中央大学 大学院 理工学研究科 情報工学専攻

4 シミュレーション

3次元GIS開発ツールキットGeo-Element [1]を用いて提案システムを構築し、有楽町付近を再現したWalk eye Mapのサンプル地図データ ($\alpha = 7.25$)でシミュレーションを行った。また、表1の環境で、3.2で述べた(a)～(d)の各々の処理時間を測定した。

表1: シミュレーション環境

| | |
|--------|-----------------------------------|
| CPU | Intel Pentium4 2.80GHz |
| メモリ | 1.5GB |
| ビデオカード | nVIDIA GeForceFX5700 |
| OS | Microsoft Windows XP Professional |

4.1 効果の検証

図2, 図3は、フライスルー時(高度40メートル)に半透明化と消去を行ったものである。また、図4, 図5は、ウォークスルー時に平面化を行い、その後、現在地を10メートル上昇させたものである。図2, 図3より、視界が拡大され、高位置にある情報を容易に把握できることがわかる。なお、ここでは、図2から図3に至るまでに15個の建物が新たに認識可能となった。同様に、図4, 図5では、視界が拡大されるだけでなく、その視界に含まれる建物への大まかな道順が理解でき、さらに、外枠を描画することで、ほぼ同色の建物同士の境界を明確にし、周囲の配置構造も理解できる。なお、ここでは、図4から図5に至るまでに、24個の建物が新たに認識可能となった。また、近距離にあり、ウィンドウ内に全表示できない階数の多い建物などは、ユーザがオブジェクトを自由に選択し、縮小表示・下方移動することによりウィンドウ内に収めることができる。

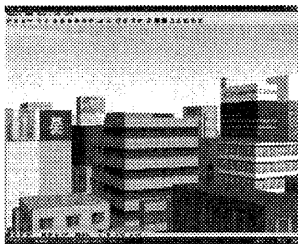


図2: 半透明化前



図3: 半透明化後

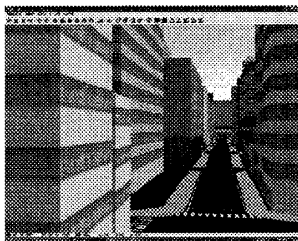


図4: 平面化前



図5: 平面化後

4.2 計算時間

表2は、各々の追加機能のボタンを1回押してからウィンドウに描画するまでの処理時間を10地点で10回(計100回)測定し、最小値と最大値を算出したものである。半透明化と平面化では、 n が大きいほど遠景の建物まで処理の対象とできるが処理時間が増大する。しかし、通常、遠景で著しく小さく見える建物を把握する場合、視点を動かさずに処理を続行する必要はない。したがって、 $9 \leq n \leq 15$ 程度で、周囲の把握は十分に可能である。また、縮小表示は、看板やガラス窓など形や文字が歪んでしまい、個々の情報は認識しにくくなるが、縮小する割合を調節することにより、1回のボタン操作でウィンドウ内に全表示可能である。一方、下方移動は、個々の情報を変形させることなく上から下まで把握できるが、階数の多い建物ほど全表示するまでの時間は長い。

表2: 処理時間の最小値, 最大値 [s]

| | $n = 3$ | $n = 9$ | $n = 15$ |
|------|---------------|---------|----------|
| 半透明化 | 0.000 | 0.016 | 0.032 |
| | 0.078 | 0.203 | 0.438 |
| 平面化 | 0.313 | 0.625 | 0.781 |
| | 1.500 | 2.359 | 3.997 |
| 縮小表示 | 0.156 ~ 1.078 | | |
| 下方移動 | 0.156 ~ 0.813 | | |

5 むすび

本稿では、ビル街における歩行者を対象として、3次元地図ビューアを提案し、Geo-Elementを用いてシステムの構築及びシミュレーションを行った。提案システムによって、3次元地図の通常表示では把握が難しい建物表面の情報をウィンドウ内に表示し、容易に周囲を把握できることを示した。同形式の地図データがあれば、他の地区でもシミュレーション可能である。また、将来、壁面テクスチャの高精度化に伴い、本システムの有用性がさらに高まると思われる。なお、今後の課題として、以下が挙げられる。

- 各オブジェクトに対して接地面データを追加する。
- 建物のデフォルトの形状から平面化を行うまで段階分けして表示させる。
- 歩道橋、信号などの歩行者に対する重要度と表示方法を決める。

謝辞

本研究の一部は、中央大学理工学研究所共同研究、中央大学特定課題研究の補助を受けた。

参考文献

- [1] 日本SGI株式会社: “3次元GIS開発ツールキットGeo-Elementプログラミングガイド”, 第1.1.1版, 2005.