

避難シミュレータの実用化・普及のための ソフトウェア開発

佐々木 聡之 朝比奈 勇揮 喜多 康 横井 利彰

これまで、通常の避難訓練を補完することを目的として、Java3Dで仮想空間を構築した避難シミュレータの開発を進めている。本稿では、火災時の避難の際に必要な防御姿勢への対応機能の追加と、プログラミング経験が少ない利用者でも容易に仮想空間の構築・管理が可能となるJava3Dモデリングソフトウェアの開発、そして本格的な避難シミュレータに興味をもってもらきかけとしてのスマートフォン版避難シミュレータアプリの開発についての成果を報告する。

キーワード：防災、シミュレーション、Java3D、モデリング、スマートフォン

1 まえがき

東京消防庁管内での2011年中の火災件数は5088件あり、この火災による死者は105人となっており、火災件数・死者数ともに減少傾向にはあるが、依然として社会にとっての日常的脅威となっている[1][2]。未然に火災の発生を防止することが最善ではあるが、現実起きた際の減災の視点から、多くの人が災害場面でのるべき姿勢・行動を事前に学ぶことは、意義が大きいと考える。

筆者らの所属する研究室では、通常の避難訓練を補うことを考え、コンピュータ上に仮想空間を構築して、その中で災害の疑似体験をするための避難シミュレータを3つのモデルに分けて開発・研究を行っており、それぞれの内容の充実を並行して進めている状況にある。

- ・現実感重視モデル：
没入感を重視したモデル
- ・学校・地域向けモデル：
同時参加型モデルを基本とし、ネットワークを介し
多人数での避難行動を防災教育に役立てるモデル
- ・インターネット配信モデル：
不特定多数の人に避難シミュレータ用の仮想空間を
各自で構築し体験してもらうモデル

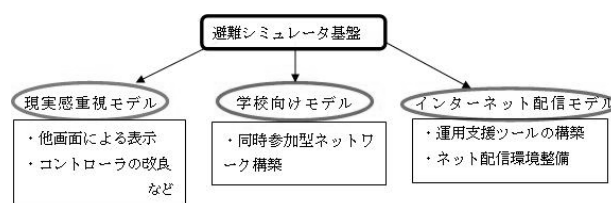


図1 避難シミュレータ3つの開発モデル

2 実用化・普及のためのソフトウェア開発

本稿では、まず、疑似体験の中で避難時の防御姿勢を学習するための、しゃがむ動作の検出センサー追加と視点との連携、それに合わせた煙描写の改良について述べる。次に、プログラミングの知識を持たない利用者でも、避難シミュレータ用の仮想空間モデルの作成が可能となるモデリングソフトウェアの開発について述べる。最後に、避難シミュレータに興味を持ってもらうことを目的として、スマートフォン向けの簡易版避難シミュレータの開発を行った結果についてまとめる。

2.1 煙に対する防御姿勢の学習への対応

火災現場においては、時間経過とともに段々と煙が濃くなっていき、またその中で避難を行わなければならない。その際に、煙が立ち込めた中でも避難できるようにするために防御姿勢を採る必要がある。防御姿勢は、比較的新鮮な空気が残っている床の方向にしゃがみながら壁をたよりに避難していく姿勢である[3]。そこで防御姿勢に対応するしゃがむ動作のセンサーを追加して画面に反映することにより、防御姿勢を体得し減災に役立つものとする。なお、防御姿勢に合わせて煙の視覚的描写を時間経過とともに徐々に濃くなるような改良も行った。

SASAKI Satoshi

東京都市大学大学院 環境情報学研究所1年生

ASAHINA Yuuki

東京都市大学環境情報学部情報メディア学科2011年度卒業生

KITA Yasushi

東京都市大学環境情報学部情報メディア学科2011年度卒業生

YOKOI Toshiaki

東京都市大学環境情報学部情報メディア学科教授

2. 1. 1 防御姿勢状態の取得とシミュレータへの反映

しゃがむ姿勢で前に進む防御姿勢での避難をシミュレータに反映させるために、傾きを検出し無線で情報を送信できるデバイスである SunSPOT (図2) を用いる。このデバイスは、3軸(X, Y, Z)加速度情報や光・温度センサーを有している。すでに現実感重視モデルとして研究を進めている避難シミュレータでは、移動の指示用のデバイスとしてこのSunSPOTを用い、その前後の傾斜で前進・後進、左右への傾斜で旋回動作を行うように、避難シミュレータで利用している[4]。

今回、しゃがむ姿勢の検出に用いるために、腿にSunSPOT 入れこんだ器具を取り付け (図3)、しゃがむときの足の動作に合わせてシミュレータの視点を変更できるようにプログラムに機能を追加した。しゃがんだ角度の情報は、ネットワーク通信を介して避難シミュレータ本体のプログラムに送信されている。



図2 無線センサーデバイス SunSPOT



図3 しゃがむ姿勢の検出用 SunSPOT の取り付け具



図4 しゃがんだ時の避難シミュレータ画面

2. 1. 2 煙描写についての改良

実際の火災現場では、有害でかつ濃い煙が時間経過とともに発生し、それが視界を遮り避難を難しくする。そこで本シミュレータでは防御姿勢追加に合わせて視覚的な煙の描写を実際の火災現場のように、時間経過とともに徐々に濃くなり、視界を遮るように改良を施すこととした。煙はJava3Dの部品である fog を用いて発生させ、時間経過とともに煙の濃さが段々と濃くなるような仕掛けを組み込んだ。図5は煙が全くない状態、図6は開始から30秒後の状態、図7は完全に充満した状態をそれぞれ示している。



図5 煙がない状態

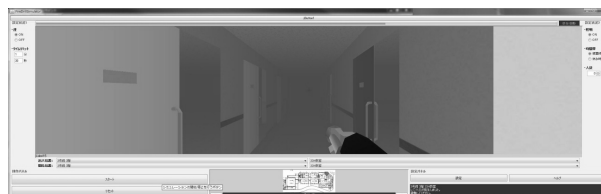


図6 煙発生から30秒経過時の状態



図7 煙が充満した状態

2. 2 java3Dによる仮想空間作成支援のためのモデリングソフトウェア

本研究室で開発している避難シミュレータでは、リアルタイムグラフィック Java3D を用いている。今後、避難シミュレータを広く配信し本学外の人間を対象としたサービスを行う場合、その利用者に合わせたより多くの施設モデルが必要になると考えられる。しかし、利用者自身が java3D でのプログラミングを行い、施設モデルを一から作成することは、プログラミング技術や3次元CGの知識が必要であり、容易ではないと考えられる。そこで、利用者がプログラミングに関する特別な知識を持たなくても施設モデルを作成・管理出来るソフトウェアが必要である。その解決策として、昨年度に3Dモデル作成を容易にするためのモデリングソフトウェアの開発が行われた ([5], 図8)。しかし、まだ発展の余地や未実装

の機能がかったため本年度追加開発を行うこととした。

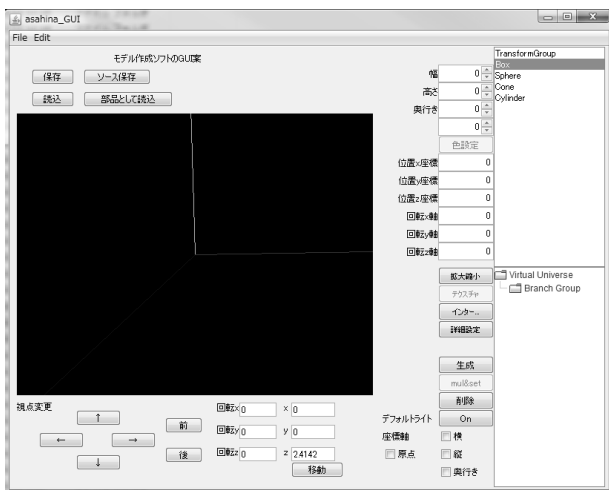


図8 2010年度卒業研究でのJava3Dモデリングソフトウェア主画面 [5]

ジェットのグループ間の移動が出来るようにした。

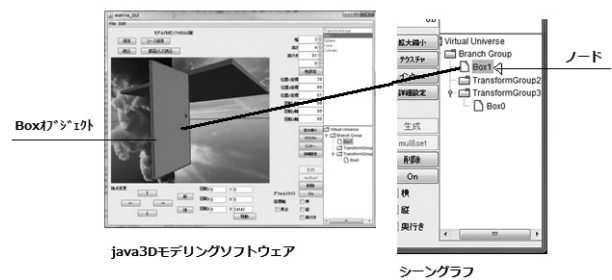


図9 シーングラフノードとオブジェクトの関連図例

2. 2. 1 モデリングソフトウェアの要求事項

このソフトウェアで求められることを以下に掲げる。

- ・プログラミング未経験者でも簡単な操作で3D作成が出来るように、GUIを利用した対話型インターフェースを用意する
- ・利用対象者は、それぞれの施設の防災担当者や、学校の先生などの防災訓練責任者を想定する
- ・GUIパーツ (テキストフィールド, ボタン, リスト, ツリー) へのキーボードによる数値入力やマウス操作によって3Dモデルの作成を可能とする

2. 2. 2 本年度に開発したシステムの機能

昨年度ソフトウェアの問題点として、以下の点があった。

- (1) 作成したオブジェクトの編集が出来ないこと
- (2) 避難シミュレータとの連携システムの未実装
- (3) 操作を戻すことが出来ないこと
- (4) モデリング機能の不足

そこで、今年度は以降に示す機能を開発して実装した。また昨年度との比較表を表1に示す。

(1) オブジェクト管理システム

本ソフトウェアで生成したオブジェクトを管理しやすくするために、swing.jTree を使用し、シーングラフを可視化したガイドを作成した。生成したオブジェクトとシーングラフノードを関連付け、ノードをクリックすることで対応する TransformGroup や、Primitive, Interpolator, Appearance の各オブジェクトの編集や、ノードのドラッグ・ドロップを行うことで対応したオブ

(2) Java3D モデル転送システム

本ソフトウェアが生成した java3D モデルの情報と衝突判定コードをファイルとして出力する機能を作成した。避難シミュレータ本体側で、本ソフトウェアが出力したモデルの java ファイルを読み込むことで、仮想空間上に衝突判定処理が適応された java3D モデルの表示を行うことが出来るようになった。



図10 モデルの保存・出力と読み込みの仕組み

(3) リドゥ・アンドゥ機能

直前の操作と直前に変更されたオブジェクトに関するログを利用し、本ソフトウェアの操作の「元に戻す」、「再実行」を行うことが出来る。

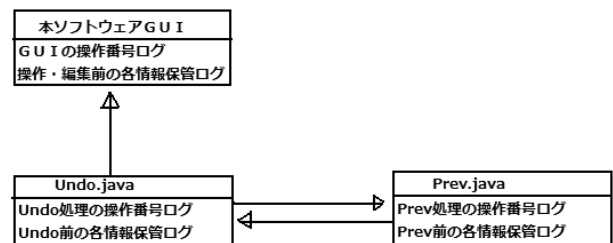


図11 本ソフトウェア GUI のログ、Undo 処理のログ、Prev 処理のログの参照イメージ

(4) モデリングの機能追加

- ・アニメーション設定 (Interpolator) : TransformGroup に対して回転, 平行移動, 拡大縮小, 色変化, スプライン統合, 透明度変化のアニメーションを設定する。
- ・テキストチャマッピング : Appearance オブジェクトを編集し, 3D モデルに指定した画像を貼り付ける。

表1 本年度に開発したシステムの機能の
昨年度との比較

昨年度	本年度
<ul style="list-style-type: none"> 主画面GUI開発・オブジェクト作成削除 オブジェクトの位置座標、回転軸設定 色設定 ・拡大縮小 ・保存、読み込み 操作補助・・・3Dグリッド線表示 視点変更 	<ul style="list-style-type: none"> 避難シミュレータへの転送処理 ・アニメーション設定 マウス操作の最適化 ・オブジェクトとアニメーション編集 色、画像(Appearance)編集 操作補助・・・元に戻す、再実行 ショックマフットのトランプ

上記の(1),(3),(4)により使用性・操作性が向上し,(2)により本ソフトウェアで生成した3Dモデルを避難シミュレータの仮想空間上で表示する事が可能になった。以上により,本ソフトウェアを利用することで,CG知識やプログラミング経験をほとんど持たない利用者でも施設モデルの作成が出来るようになったものと考えられる。しかし3Dモデル作成は,キーボードによる数値入力による方法しかないため,ポリゴンなど複雑な3Dモデルを生成する際,現在のキーボードによる数値入力方式のみでは操作が難しい。直観的な操作を可能にするため,GUIパネル部分へのマウス操作方式の導入が今後求められると考える。

2.3 スマートフォン向け避難シミュレータアプリケーションの開発

パソコンを前提とした没入感重視モデルでは,体験できる場所や時間に制限がかかってしまう。今後避難シミュレータへの理解を高め,普及・促進を行なっていくにあたり,簡易版で体験できるものを用意することがその助けになると考えた。そこで,現在多くの人が所有しているスマートフォン向けに簡易版の避難シミュレータを開発することとした。その際,避難における重要な学習ポイントを楽しみながら体験して学ぶことが可能ではないかと考えた。開発プラットフォームは,PC版避難シミュレータとの連携も考え,Javaで開発が行えるAndroidでの開発を行った。構成は大きく分けて,3Dモデルの構築,GUIの設計の2つとした。

2.3.1 3Dモデルの構築

先に述べたとおり,まずPC版避難シミュレータの利用促進を行うためには3Dを用いた現実感のあるアプリケーションが必要となる。しかし,携帯端末での3Dモデルの描画や動作などはCPU性能が問題となる。そこで,3D空間をより高速に描画する為にグラフィック用ライブラリであるOpenGL(Open Graphics Library)[6]のAPI群の中から一部のAPIを削除し携帯端末などの組み込み用途に特化したOpenGL ES(OpenGL for Embedded Systems)[7]を用いて3D描画を行うこととした。また,より簡単に3Dモデルが作成できるように,このOpenGL ESを用いて,オープンソースとして公開されているmin3D[8]を使用した。これにより,従来のJavaのように四角形や球体等の大きさを指定することで3D描画を行うことができ,OpenGL ESを土台としているため,ス

マートフォン上でも高速に3Dモデルの描画が行えるようになった。今回はこのフレームワークを用いて図12のアプリケーション部分の開発を行った。

アプリケーション ※Java

アプリケーションフレームワーク (min3d) ※Java
ライブラリ(OpenGL ES 等)
OS
ハードウェア

図12 Androidアプリケーションの階層関連図

仮想空間の構築は,壁や天井などの簡単な物体はプリミティブのBoxで作成し,椅子や扉など複雑なモデルはプリミティブだけでは作成することが難しいため,PC版避難シミュレータでも使用しているVRML(Virtual Reality Modeling Language)形式のモデルデータを使用した。そのデータを本アプリケーションで読み込めるようにOBJ形式にコンバートして使用し,3号館をイメージした3D仮想空間を構築した。

2.3.2 GUIの設計,空間内移動

当初,アプリの開始画面には,PC版避難シミュレータと同様にGUIを用いて煙設定や照度設定等と動作を1つの画面で行う予定だったが,スマートフォンの画面は小さく見づらくなってしまったため,別途パラメータ設定と動作の起動が行える画面を作成した(図13)。なお,現在ここで出来る設定は開始地点の変更である。



図13 スマートフォン向け避難シミュレータの開始画面

避難シミュレータの開始地点の設定は,誰でも簡単に設定が行えるように画面を触ることで変更が行えるようにした。画面左上のボタンをタッチすることで表示されるリストから場所を選択し(図14),スタートボタンをタッチすることで選択した場所から動作が開始する仕様とした(図15)。

また,仮想空間内の移動指示に関しては,まずPC版避難シミュレータと同等にするために前後移動を前後への傾斜角,旋回動作を左右への回転角を用いて行う仕様と

した。Android 端末に搭載されている傾きセンサーから取得した角度値に加速度を掛け、傾きの大きさに移動量が変化することとした。また、旋回動作によって向いている方向に前後移動を行うために、現在の向いている角度を取得し、その値を利用し縦横への移動を行うこととした。



図 14 開始地点の設定リスト



図 15 スマートフォン向け避難シミュレータ動作画面

以上により、スマートフォン向けの簡易版避難シミュレータを開発できた。これは、場所や時間に拘束されずに利用でき、またタッチパネルでの操作など使い方も容易なため、誰でも簡単に体験が行えるものと考え。今後は、より詳細な仮想空間の構築及び煙機能などの拡張を行い、アプリケーションの配信を行なっていくことで利用促進として多くの人に避難シミュレータを体験してもらうことが可能になると考える。また、ビデオコンテンツとの連携や学習要素を取り入れた開発も行なっていくことで、簡易版としてだけでなく、防災学習や避難の意義を多くの人に理解してもらえるようなアプリケーションに発展できるものと考え。

3 まとめ

本稿では、通常の避難訓練を補完することを目的とした避難シミュレータの開発において、避難時の防御姿勢を体得してもらうための機能追加とそれに伴う煙動作の改良、プログラミングの知識がなくても簡単にモデルの作成・管理を行えるソフトウェアの開発、スマートフォン向けの簡易版避難シミュレータの開発を行った。本研究により、体験型避難シミュレータを用いた避難時行動の学習機能が向上し、地域での仮想モデル構築と管理も行いやすい形に近づき、また身近に体験できる環境を用意することができた。今後、さらに避難シミュレータへ

機能の追加・改良を行い実用化に向けて開発を行いたいと考える。

参考文献

- [1] 横井利彰, 防災・減災のための避難シミュレータ開発, 東京都市大学環境情報学部情報メディアジャーナル第 12 号, 2011.
- [2] 東京消防庁 災害情報・災害統計
<http://www.tfd.metro.tokyo.jp/saigai/toukei/index.html>
- [3] 京丹後市消防局 生活安全情報一火災から身を守る(煙編)
<http://www.city.kyotango.lg.jp/kcfd/seikatu/kaji01/kemuri/kemuri.html>
- [4] 松川洪典, SunSPOT による避難シミュレータの操作向上の研究, 横井研究室 2009 年度卒業研究論文, 2010.
- [5] 高宮正一, Java3D モデル簡易作成ソフトウェアの開発, 横井研究室 2010 年度卒業研究論文, 2011.
- [6] OpenGL, <http://www.khronos.org/opengl>
- [7] OpenGL ES, <http://www.khronos.org/opengles/>
- [8] min3D, <http://code.google.com/p/min3d/>