動力学計算エンジン ODE 用グラフィクスライブラリの開発(第4報)

-上位互換ライブラリ irrDrawStuff のリリース-

Development of a graphic library for Open Dynamics Engine (4th report)

-irrDrawStuff: An upper compatible library with drawstuff-

出村 公成, 細川 和真 (金沢工大)

Kosei DEMURA, Kazuma HOSOKAWA, KIT, demura@his.kanazawa-it.ac.jp

This paper is the 4th report of the irrDrawStuff that is the 3D computer graphics library for the Open Dynamics Engine, and an upper compatible with the Drawstuff library. The first version of irrDrawStuff is officially released under dual licenses, the LGPL and the BSD license, on 12th January in 2010. The library has been developed continuously by authors. The current version is 0.0.4 and its compatible functions are almost finished. Functions about a menu bar, shadow rendering, single step, pause, measuring of simulation elapsed time are introduced from the former version. This paper describes those functions. Our goal is developing the robot development software platform that makes anyone be able to simulate robots easily.

Key Words: ODE, The irrDrawStuff library, The Drawstuff library, 3D graphic engine,

1. 緒言

オープンソースの動力学計算エンジンである Open Dynamics Engine (以下, ODE) [1, 2]は教育や研究におけるシミュレータを作成するために広く使われている。

ODE には, デモプログラムを可視化するために Drawstuff とよばれる3次元グラフィクスライブラリが付属している。Drawstuff は OpenGL をベースに書かれた簡易なライブラリであり, 比較的簡単に基本形状の表示が可能である。

しかしながら, 基本形状以外の複雑な形状を表示することは簡単ではなく, 描画ウィンドウに文字を表示することはできず, さらに描画スピードを一定にすることができないため, 高性能なグラフィクスチップを備えたコンピュータではディスプレイの垂直同期周波数以上に描画が更新され, CPU と GPU のパワーを有効に活用できないという問題がある。

そこで, 筆者らは 2008 年 6 月から Drawstuff の問題点を解決すべく, オープンソースの3次元グラフィクスライブラリ Irrlicht を使った上位互換ライブラリ irrDrawStuff (略称 iDS) の開発を進めてきており [3-5], 2010 年 1 月 12 日にバージョン 0.0.0 を公開した。本ライブラリは, 第3報までに以下の項目が開発されていた [5]。

- 照明, 空, 地面などの環境表示
- 基本形状である球, 直方体, 円柱, カプセル, 三角メッシュ, 線に関する API の実装。
- ウィンドウへの文字列表示機能
- 3次元オブジェクトファイルの読み込み
- ODE 付属デモプログラム (33 種類) での動作確認
- 描画速度は Drawstuff の 95%程度

本稿執筆時ではバージョン 0.0.4 を筆者のウェブサイトで公開している [6]。ODE と同時に配布することを考え, ライセンスは ODE と同じ LGPL と BSD のデュアルライセンスである。

第4報では, 公開するための配布パッケージの作成, メニューバー, 影, ステップ実行などの機能を追加した。さらに, 拡張機能としてシミュレータでの経過時間計測機能なども付与している。本稿ではその詳細について述べる。

2. 配布パッケージの作成

irrDrawStuff はマルチプラットフォームであるため, Windows, Linux, Mac OS X 環境のいずれでもソースからビルドまたはコンパイルできなければならない。各環境毎にソリューションファイル, プロジェクトファイルや make ファイルなどを作成することは効率的ではないので, ビルド構成ツールである Premake を利用した [7]。

Premake は Lua 言語を使ったプロジェクト設定ファイルを一度記述するだけで, Windows, Linux, Mac OS X 環境で利用可能な Visual Studio, GNU Make, Xcode, Code::Blocks 用のプロジェクトファイルを生成することができる。

なお, Premake の他にビルド構成ツールとして, Cmake, Scons などがあるが, ODE では Premake を使っているので Premake を採用することにした。Premake のバージョンは 4.2 を用いている。

Fig.1 A screenshot of irrDrawStuff

3. 互換機能の実装

3.1 メニューバー

Irrlicht は簡単な GUI (Graphic User Interface) が組み込まれているので、その機能を用いて Fig. 1 に示すメニューバーを実装した。メニューバーには 3 次元オブジェクトファイルの読み込み、一時停止、ステップ実行、影並びにテクスチャの On/OFF、プログラムの概要表示、終了のメニューがある。

3.2 影のレンダリング

影は 3 次元グラフィックスライブラリにとっては現実感を向上させ、各物体が 3 次元空間上でどのような位置関係にあるかを推定する情報となるので非常に重要である。影のレンダリング手法としてはテクスチャ法、地面へのポリゴン投影法、シャドウボリューム法、シャドウマッピング法などがある。Drawstuff はハードウェアの処理能力に依存せず比較的高速な地面へのポリゴン投影法を用いている。Irrlicht は低速であるが地面以外や自分自身にも投影可能なシャドウボリューム法を採用している。

IrrDrawStuff の現在のバージョン 0.0.4 では描画速度を優先するために DrawStuff と同じ地面へのポリゴン投影法を用いている。Fig. 1 に示す球の下に楕円の影がレンダリングされているのがわかる。

4. 拡張機能の実装

4.1 C++化

Drawstuff は OpenGL をベースとして書かれた簡易ライブラリであり C 言語で実装されている。主要部分は drawstuff.cpp (全 1616 行) に書かれ、マルチプラットフォームに対応するため GUI の部分は、windows.cpp, x11.cpp, osx.cpp にそれぞれの OS に対応したコードが書かれている。

Drawstuff 程度の規模なら 1 人で開発可能なので C 言語を用いて書かれても問題ないが、機能をさらに拡張するためにグループで開発することを想定すると C++ 言語を用いる方が、開発効率が良い。そこで、次のようにクラス化して C++ 言語で実装した。

- SimLoop クラス：描画ルーチン
- idsEventReceiver クラス：イベント処理
- ids***SceneNode クラス：シーンノード。***には Box, Capsule, Cylinder のいずれかが入る。
- DrawPrimitives クラス：基本形状の描画
- DrawMesh クラス：メッシュ (ポリゴン) の描画
- Vision クラス：カメラ
- Misc クラス：上記以外

この結果、グループによる開発が容易になり開発のスピードアップが期待できる。

4.2 描画速度の固定とシミュレーションの高速化

Drawstuff では描画速度の固定機能がないため、高速なグラフィックスチップを使ったコンピュータでシミュレーションを実行すると描画が速すぎるため雲の移動速度が高速で非常に不自然な感じを受ける。

描画速度は、ディスプレイの垂直同期周波数に合わせれば良く、通常の場合は 60fps 程度で十分である。それ以上描画速度が速い場合は CPU と GPU を無駄に使用している。そこで、irrDrawStuff では垂直同期周波数に描画速度を合わせる機能を追加した。

なお、ODE のデモプログラムでは動力学計算 1 ステップにつき描画が 1 フレーム更新されるためシミュレーションの実行効率が悪い。そこで、動力学計算のステップ数と描画フレームの数の割合を可変に設定できる機能を追加した。これにより Drawstuff と

Table 1 Comparison of drawing speed

	DrawStuff[fps]	irrDrawStuff[fps]	[%]
Sphere	317	302	95
Box	350	329	94
Cylinder	333	313	94
Capsule	359	358	90
Line	381	358	94
Trimesh	39	30	75
Texture	374	359	96
Bounce	531	523	98

比較した場合、シミュレーションを高速に実行できるようになった。また、ウインドウのタイトルバーにシミュレーションの計測時間をリアルタイムに表示する機能も追加している。

4.3 計測結果

irrDrawStuff-0.03 と Drawstuff との描画速度を比較した結果を Table 1 に示す。Sphere から Line までは基本形状であり、Trimesh は ODE 付属の兎データを用いた。なお、Bounce は本ライブラリ付属のデモプログラム bounce.cpp を実行した際の速度を計測した。

概ね 5% 程度の速度低下であるが、Trimesh は 25% と非常に遅くなっている。Trimesh は 3 角形の集合で構成されており、現在の実装では各 3 角形を描画する毎に diDrawTriangle() を呼び出しているためであり改善が必要である。

5. 結 言

本稿では、ODE を用いたシミュレータ開発を支援するために開発した drawstuff 上位互換ライブラリ irrDrawStuff について述べた。互換機能については概ね実装を終えた。さらに、拡張機能として、文字描画、3 次元オブジェクトファイルの読み込みと描画、描画速度の固定、シミュレーション経過時間の計測機能などを追加した。

今後の課題としては、Trimesh の高速化、GUI によるジョイントの設定などの機能拡張、並びに本ライブラリを使い、つくばチャレンジで使える実用的なシミュレータ開発などのキラーアプリケーションの作成などが挙げられる。なお、開発したライブラリは文献[6]で配布している。

文 献

- [1] Open Dynamics Engine: <http://www.ode.org/>, (2009 年 3 月 20 日アクセス).
- [2] 出村公成: “簡単! 実践! ロボットシミュレーション-Open Dynamics Engine によるロボットプログラミング-”, 森北出版 (2007)
- [3] 出村公成: “動力学計算エンジン ODE 用グラフィックスライブラリの開発”, 第 26 回日本ロボット学会学術講演会, 1I1-05 (2008)
- [4] 出村公成: “動力学計算エンジン ODE 用グラフィックスライブラリの開発 (第 2 報) -drawstuff 上位互換ライブラリ-”, 1A2-B03 (2009)
- [5] 細川 和貞, 出村 公成: “動力学エンジン ODE 用グラフィックスライブラリの開発 -drawstuff 上位互換ライブラリ- (第 3 報)”, 3M1-06 (2009)
- [6] demura.net: RoboCup・ODE・夢考房, <http://demura.net>, (2010 年 3 月 17 日アクセス)
- [7] Premake, <http://industriousone.com/premake>, (2010 年 3 月 17 日アクセス)