

メディア情報学実験・メディア生成・制作実験

最終課題 作品タイトル：吊り橋

2010335 佐野遵平

s2010335@edu.cc.uec.ac.jp

1 仮想世界の説明

1.1 仮想世界について

このモデルは、吊り橋を再現したものである。具体的には、東京のレインボーブリッジに似た見た目になるように制作した。また、昼と夜の景色を切り替えられるようにした。

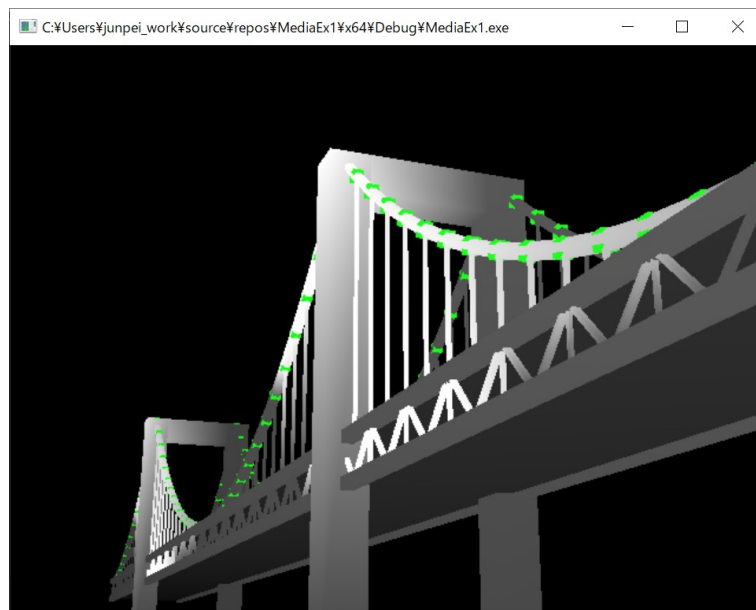


図 1: 製作物のスクリーンショット



図 2: 実際のレインボーブリッジ

1.2 操作方法

マウスのボタンを押しながらマウスを動かすことで操作ができる。マウスの左ボタンで回転、中央ボタンで移動、右ボタンで拡大・縮小ができる。また、スペースキーを押すことで、昼（空が青色）と夜（空が真っ暗）を切り替えることができる。

2 アピールポイント

2.1 橋の構造

できるだけ本物のつり橋に近く見せるために、次のような点をこだわって制作を行った。

一つ目は、メインケーブルの形である。実際のつり橋は、支柱間のケーブルの形が放物線になっている。そこで、本課題のモデルでも実際に $y = ax^2$ の式から放物線を求め、それに合わせてケーブルを配置している。

二つ目は、橋の構造である。モデルとしたレインボーブリッジは、上が高速道路、下が一般道とゆりかもめというように2階建て構造となっている。そこで、本課題のモデルにおいても2階建て構造を採用した。また、1階部分と2階部分をつなぐ柱の構造も本物のレインボーブリッジに合わせ、三角形状に配置された柱を再現した。

三つ目は、橋の設計におけるプログラムである。今回のプログラムでは、橋の長さや幅などのパラメータを少し変更するだけで、容易に橋の大きさや形を変えることができる仕組みになっている。したがって、人手で頂点を一つずつずらすことなく橋の形を変えることができる。これは、上述の放物線の関数を利用したり、プログラムで自動的に頂点の場所を計算したりしているためである。

四つ目は、計算量を減らす工夫である。今回のプログラムは、先述のとおり頂点座標などの多くをプログラ

ムの実行時に計算している。よって、これらの値を毎フレーム計算していると、1 フレーム当たりの計算量が増えてしまうことが想定される。従って、上述の放物線の計算や各頂点の座標の計算はプログラム起動時に一度だけ行い、それらの値を変数に格納することで、その後のフレームでの計算量を減らす工夫をしている。

2.2 カメラの移動

今回のプログラムは、マウスの左ボタンを押した状態でマウスをドラッグすることで回転、中央ボタンを押した状態でドラッグすることで視点位置の移動、右クリックをした状態でマウスを上下に動かすことで拡大・縮小を行うことができる。これにより、仮想世界の中をほとんど制約なく自由に動いたり回転したりできる。これは、3D を扱うソフトで実際に使われている操作性をできるだけ再現しようとしたためである。

具体的には、左ボタンが押された状態では視点中心からの角度を変更している。マウスの x 座標の変位を 3D ワールド内の y 軸回りの角度に、マウスの y 座標の変位を 3D ワールドの地面とのなす角に反映している。

また、中央ボタンを押した状態では、視点中心とカメラとの距離を変えないまま視点中心の位置を変更している。マウスの x 座標の変位は、カメラと視点中心を左右方向に移動している。このとき左右方向は、カメラから視点の中心へのベクトルと鉛直上向きのベクトルの外積を計算することで得ている。マウスの y 座標の変位は、鉛直上向き方向への移動に反映している。さらに、視点の中心との距離が近いときはゆっくり、離れているときは素早く動くようにすることで、できるだけ自然な操作性を実現しようと試みている。

さらに右クリックでは、カメラが視点中心の周りを回転する際の半径の大きさを変更している。

2.3 陰影処理

橋の見た目をできるだけ本物に近づけること以外にも、次の点を意識して製作した。

一つ目は、光の当たり方が変わっても自然に見えるようにすることである。今回の製作は、昼と夜を変更できるようにするなど、光の当て方を変更しても問題が起きないような仕様で制作を行った。従って、特に橋の白い部分は、光が当たっている面は白く輝き、光の当たっていない面は薄暗くなるようにマテリアルの設定を行った。

二つ目は、ライトアップである。スポットライトとエミッションのマテリアルを使用することで、実際のレインボーブリッジに近いライトアップを再現しようと試みた。

2.4 テクスチャマッピング

海面を再現するために、水面の画像を用いてテクスチャマッピングを行った。1 枚のテクスチャを大きく広げた場合、解像度が悪くなってしまうことが想定されたため、複数枚をタイルのように並べて配置するプログラムを制作した。

3 実験の考察

今回のプログラムにおいて、次のような点が反省点であったと考えている。

一つ目は、グローバル変数を多用しすぎたことである。今回のプログラムでは、計算量削減のために一度計算した頂点座標などを変数や配列に格納していた。しかし、これらの値をグローバル変数として記録してしまったため、グローバル変数の値が増加する結果となってしまった。一方で計算量についても、コンピュータの性能によってはほとんど削減した意味は無かったのではないかと考えられる。従って、計算に大きな時間がかかりそうな値だけを記録するなどの対応をすることで、グローバル変数をもっと減らす検討をすべきだったと考えられる。

二つ目は、クラスを中途半端に導入してしまったことで、プログラムが分かりにくくなってしまったことである。先述のようにグローバル変数が増えてしまったため、それを減らすためにクラスを導入した。また、複数枚のテクスチャマッピングを行う際に、共通の処理をまとめられるようにするためにもクラスを作成した。しかし、制作終盤で導入したために、制作の一部でしか利用することができなかった。また、テクスチャも結局 1 枚しか貼らなかったため、クラスにしたことが無意味となってしまった。その結果、C 言語と C++ 言語の混在する分かりにくいプログラム構造となってしまった。従って、クラスを使うのであれば製作初期からオブジェクト指向を意識してプログラムすべきだったと考えられる。

4 感想

座標変換やマテリアルの設定などがなかなか掴めず苦労してしまい、想像以上に製作に時間がかかってしまった。しかし、3D ソフトが内部でどのような処理をすることで仮想世界を再現しているのかについて理解を深めることができた。

5 スクリーンショット

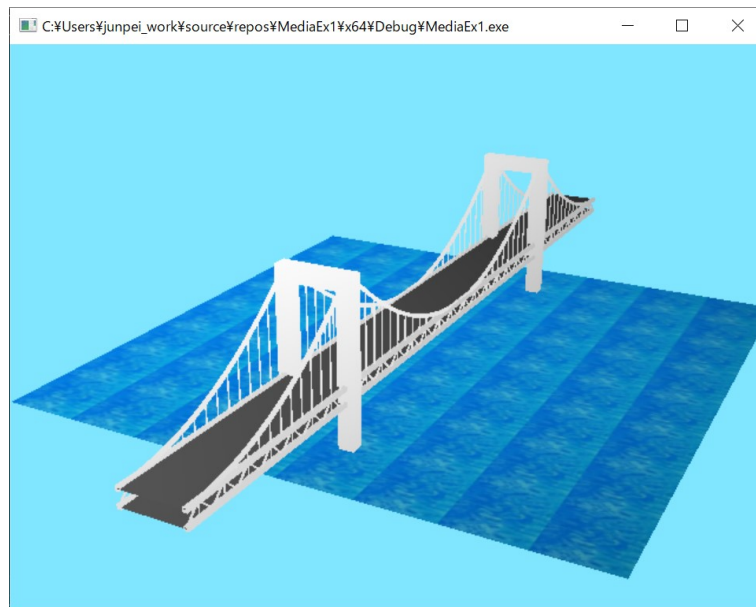


図 3: スクリーンショット (昼)

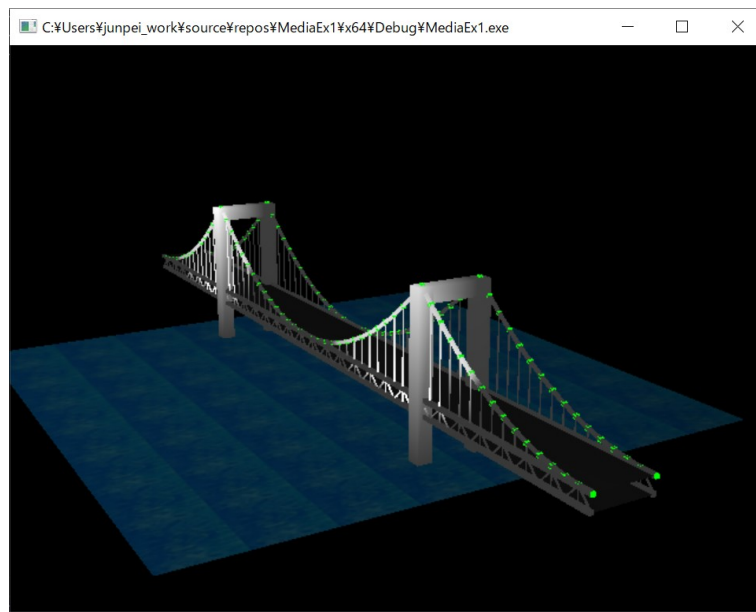


図 4: スクリーンショット（夜）

6 プログラムの置き場所

ディレクトリ

/home0/y2020/s2010335/CG_2010335/

コンパイル方法

```
g++ main.cpp -lglut -lGL -lGLU -lm
```

実行ファイル

./a.exe