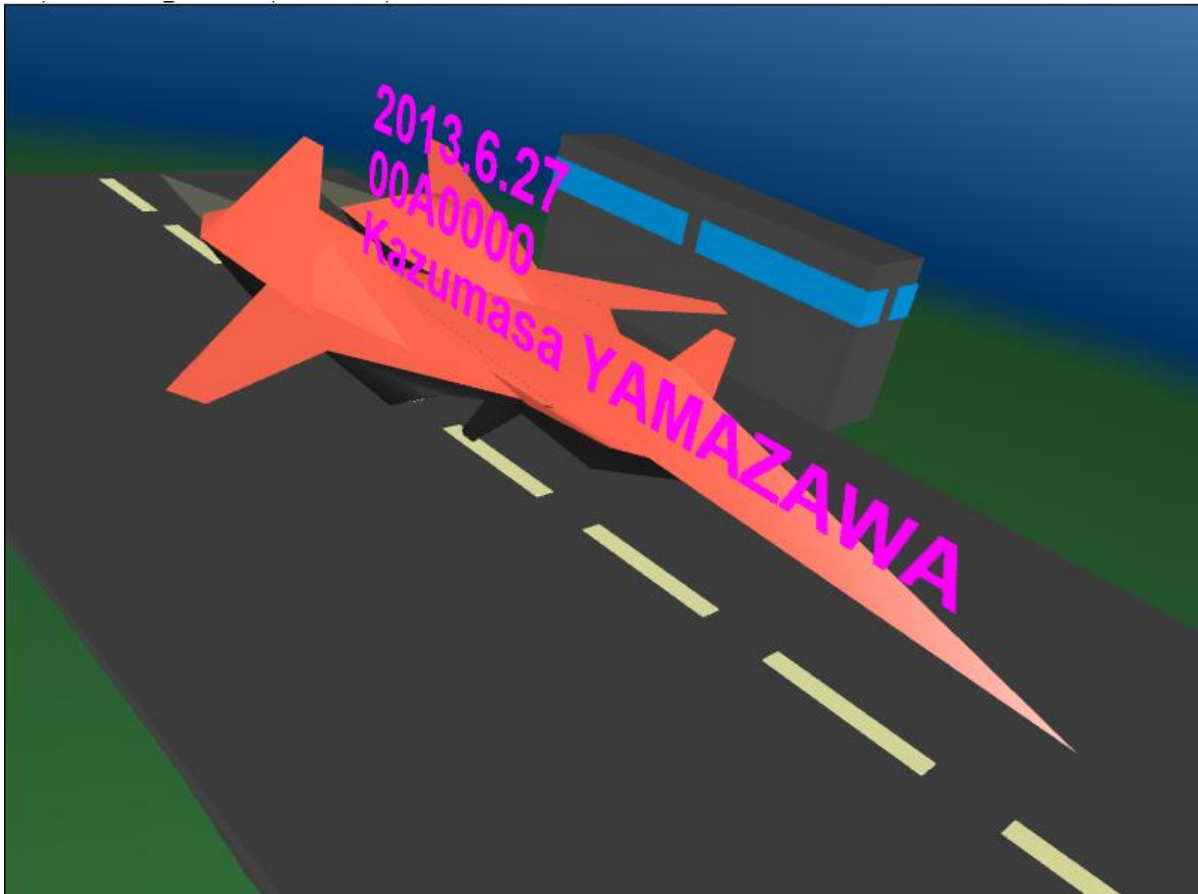


情報工学実験Ⅲ

実験テーマ：Web 対応 3 次元グラフィックスの制作
(情報工学科 3 年前期)



福岡工業大学 情報工学部 情報工学科
山澤 一誠
(yamazawa@fit.ac.jp)

本テキストは次のホームページで閲覧・ダウンロード可能です。
<http://www.fit.ac.jp/~yamazawa/jikken3/>

Web 対応 3 次元グラフィックスの制作

1. 実験の目的

最新の Web3D 技術である X3DOM を用いて 3 次元仮想世界を制作することを通し、Web 対応インタラクティブ 3 次元グラフィックスの基礎を習得する。

2. 達成目標

- ① 基本的 3 次元形状（直方体，球，円柱，円錐）の記述法を理解する。
- ② 3 次元形状の幾何変換（平行移動，スケール変換，回転）の記述法を理解する。
- ③ 光の 3 原色 RGB の加法混合と透明度 α チャンネルを理解する。
- ④ 光源，視点，背景の記述法を理解する。
- ⑤ 3 次元空間内に文字を記述する方法を理解する。

3. 作品が満たすべき条件

X3DOM を用いて自分の好きな 3 次元仮想世界を制作する。ただし、次の条件を満足している必要がある。

条件 1：球や円柱など基本的 3 次元形状を **5 個以上** 組み合わせて使用する。

条件 2：3 次元形状の平行移動，スケール変換，回転のすべて使う。

条件 3：背景，視点，光源を設定する。

自分の作品にあった背景に変更してください。

ヘッドライトを消して(headlight="false")もらっても結構です。

条件 4：視点には必ず異なる名前（mae, kao, zentai など）をつけて **新たに三つ以上** 設定する。

視点の追加方法は 7 節最後のほうを参照してください。

条件 5：視点はブラウザに表示される **Viewpoint タグ** をコピーして最低一つ設定する。（再提出になる一番の原因）

自分の作品がかっこよく見えるアングルをお願いします。

条件 6：制作年月日，学籍番号，氏名は **初期視点からはっきり大きく見えるところに** 配置する。

最初の（一番上の）Viewpoint が初期視点です。

ステータス表示などは邪魔なら消して(showStat="false")もらって結構です。

条件 7：HTML のタイトル(2 箇所)として **作品名** と本文に作品の説明や感想を HTML に記入する。

例えば，ミスタードーナツのボン・デ・ライオン。

4. 作品の提出

(1) myfit のクラスプロファイルから課題提出。

(2) 添付ファイルとして作成したファイルをアップロードする。

(3) 実験時間中に提出した人は山澤の **チェックを受け**，OK が出たら退室しても構わない。

(4) 実験時間中に完成させることができなかったときは，USB メモリなどにコピーをとり，

実験後 1 週間以内にクラスプロファイルから提出する。

後ほど，山澤がコメントを書くので **必ずチェック**しておくこと。

(5) 送られてきたファイルにエラーがあるとき，ファイルが添付されていないときなどにはその旨をクラスプロファイルのコメントに書くので，修正後，クラスプロファイルより **再提出**すること。

5. 制作の流れ

3次元仮想世界制作の大まかな流れは次の通りです.

【ステップ1】 ログアウト時に消去されない場所にフォルダ “Web3D” を作る.

【ステップ2】 次のホームページにアクセスし, ユニットプログラム (ファイル名 “unit.htm”) をダウンロードしてフォルダ “Web3D” に保存する.

URL : <http://www.fit.ac.jp/~yamazawa/jikken3/>

【ステップ3】 フォルダ “Web3D” 内のファイル “unit.htm” をコピーし, “適当な名前.htm” というファイル名で保存する. 適当な名前には自分の学籍番号などを直接打ち込む.

【ステップ4】 ファイル “適当な名前.htm” を Firefox 等の WebGL 対応のブラウザで開くと 3次元世界が表示される. いろいろな角度からオブジェクトを観察し, 図 1 のソースコードと照らし合わせて理解する.

【ステップ5】 基本的 3次元形状 (直方体, 球, 円柱, 円錐) を組み合わせてできそうなものを決め, その概略図を描いてみる. 想像力を働かせば, 基本的 3次元形状のスケール変換でかなりいろいろな形状を作ることができる. 例えば, 線や板は直方体から, 卵のような形は球から, 円盤は円柱から, 三角形は円錐から簡単に作ることができる.

【ステップ6】 ファイル “適当な名前.htm” をテキストエディタ (フリーソフトのサクラエディタなど) や HTML エディタで開き, そのコードを, 「コピー」, 「貼り付け」, 「修正」, 「上書き保存」, 「画像の確認 (更新ボタンを押す)」を繰り返すことによって少しずつ作り込んでいく.

(使いこなせるならば Blender 等の CG ソフトを利用して, X3DOM ファイルを作成しても良い)

```

<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8">
    <script src="http://www.x3dom.org/download/dev/x3dom.js"></script>
    <link rel="stylesheet" href="http://www.x3dom.org/download/dev/x3dom.css">
    <title>タイトル</title>
  </head>
  <body>
    <h1>タイトル</h1>
    <p>
      作品の説明<br>
      . . 省略 . .
    </p>
    <div class="buttons group">
      . . 省略 . .
      <input type="button" value="前から (+Z から)"
        onclick="document.getElementById('front_view').setAttribute('set_bind','true');"/>
      A' { <input type="button" value="後から (-Z から)"
        onclick="document.getElementById('back_view').setAttribute('set_bind','true');"/>
      . . 省略 . .
    </div>
    <x3d id="x3d" width="800px" height="600px" showLog="true" showStat="true">
      <scene>
        <!-- ヘッドライトの設定 -->
        <navigationInfo id="navi" headlight="true"></navigationInfo>

        <!-- 太陽光源の設定 (複数可) -->
        <!-- direction 光の方向 -->
        <!-- ambientIntensity 環境光の明るさ -->
        <directionalLight direction="-1 -1 -1" ambientIntensity="0.5"></directionalLight>

        <!-- 背景の指定 -->
        <background skyColor="0.2 0.2 0.4"></background>

        <!-- 視点の設定 (複数可) -->
        <viewpoint id="front_view" description="front" position="0 0 10" orientation="0 1 0 0"></viewpoint>
        <viewpoint id="back_view" description="top" position="0 0 -10" orientation="0 1 0 3.14"></viewpoint>
        . . 省略 . .
        B { <transform scale="0.2 1 1" rotation="0 0 1 1.57" translation="-1 0 0">
          <shape>
            <sphere radius="2"></sphere>
            <appearance>
              <material diffuseColor="1 0 0" specularColor=".5 .5 .5" shininess="0.2" transparency="0"></material>
            </appearance>
          </shape>
        </transform>

        <transform scale="1 1 1" rotation="0 0 1 0" translation="-1 -2 0">
          <shape>
            <box size="10 0.1 10"></box>
            <appearance>
              <material diffuseColor="1 1 1" specularColor=".5 .5 .5" shininess="0.2" transparency="0"></material>
            </appearance>
          </shape>
        </transform>

        <transform scale="0.5 0.5 0.5" translation="0 2 0">
          <shape>
            <text string="2015.4.1 00A0000" "Kazumasa YAMAZAWA">
              <fontstyle family="TYPEWRITER" size="1" style="BOLD"></fontstyle>
            </text>
            <appearance>
              <material diffuseColor="1 0 1"></material>
            </appearance>
          </shape>
        </transform>

      </scene>
    </x3d>
    . . 省略 . .
  </body>
</html>

```

図1 ユニットプログラム

6. ユニットプログラムの説明

前述の【ステップ2】でダウンロードしたユニットプログラム（ファイル名“unit.htm”）は、図1のようなコードであり、これが作品制作の出発点となるコードである。長さの単位はメートル、角度の単位はラジアンである。

7. 編集方法

●背景の色を変えるには

skyColor の値を変える。

1 0 0 赤, 0 1 0 緑, 0 0 1 青, 1 1 0 黄, 1 0 1 紫, 0 1 1 空色,
0 0 0 黒, 0.3 0.3 0.3 濃い灰色, 0.7 0.7 0.7 薄い灰色, 1 1 1 白

●物体の形状を変形するには

図1のBのscaleの値を変更する。例えば,

scale="0.2 1 1" # x 軸方向を 0.2 倍にする

scale="0.1 10 0.1" # x 軸方向を 0.1 倍, y 軸方向を 10 倍, z 軸方向を 0.1 倍

scale="1 0.01 1" # y 軸方向を 0.01 倍にする

●物体を回転させるには

図1のBのrotationの値を変更する。例えば,

rotation="1 0 0 -0.78" # x 軸回りに -0.78 ラジアン (-45 度) 回転させる。

rotation="0 1 0 3.14" # y 軸回りに 3.14 ラジアン (180 度) 回転させる。

rotation="0 0 1 1.57" # z 軸回りに 1.57 ラジアン (90 度) 回転させる。

注) 任意のベクトル回りの回転については<付録1>を参照。

●物体の位置を変えるには

図1のBのtranslationの値を変更する。例えば,

translation="1.5 0 0" # x 軸方向に 1.5m 平行移動

translation="-1.5 0 2" # x 軸方向に -1.5m, z 軸方向に 2m 平行移動

translation="3 3 3" # x 軸方向, y 軸方向, z 軸方向にそれぞれ 3m 平行移動

●幾何変換の順序

物体はまず scale によって「スケール変換」され、次に rotation によって「回転」され、最後に translation によって「平行移動」される。

●物体の色を変えるには

diffuseColor の値を変える。値は背景の色を参照。

●基本的3次元形状を変えるには

図1では<sphere radius="2"></sphere>となっているが、これを次と入れ替えることにより、直方体、円錐、円柱に変えることができる。

#直方体. 三つの数字はそれぞれ x, y, z 方向の長さ (メートル)

<box size="2 2 2"></box>

#円錐 パラメータはそれぞれ底面の半径, 高さ, 底面を描画するかしないか

<cone bottomRadius="1" height="2" bottom="TRUE" ></cone>

#円柱 パラメータはそれぞれ底面の半径, 高さ, 底面および上面を描画するかしないか

<cylinder radius="1" height="2" bottom="TRUE" top="TRUE"></cylinder>

●新しい3次元形状を追加するには

図1のBの部分をコピーし、その下に貼り付ける。そしてその中身を少しずつ書き換えては上書き保存し、Webブラウザで画像のどこが変化したかを確認する。

●新しい視点を追加するには

図1のAの部分をコピーし、その下に貼り付ける。そして、ID(変数名)、視点の名前、位置、方向(図4参照)の値を書き換える。同様にA'の部分をコピーし、その下に貼り付ける。そして値(ボタンの表示文字列)、変数名を書き換える。書き換えたら上書き保存し、Webブラウザで確認する。

●物体をまとめて回転や移動するには

＜付録2＞を参照してください

8. 作品例の説明

図2は基本的3次元形状の幾何変換だけで作った飛行船である。個々の部品をスケール変換と回転によって作り、平行移動によって適切な場所に配置すればよい。

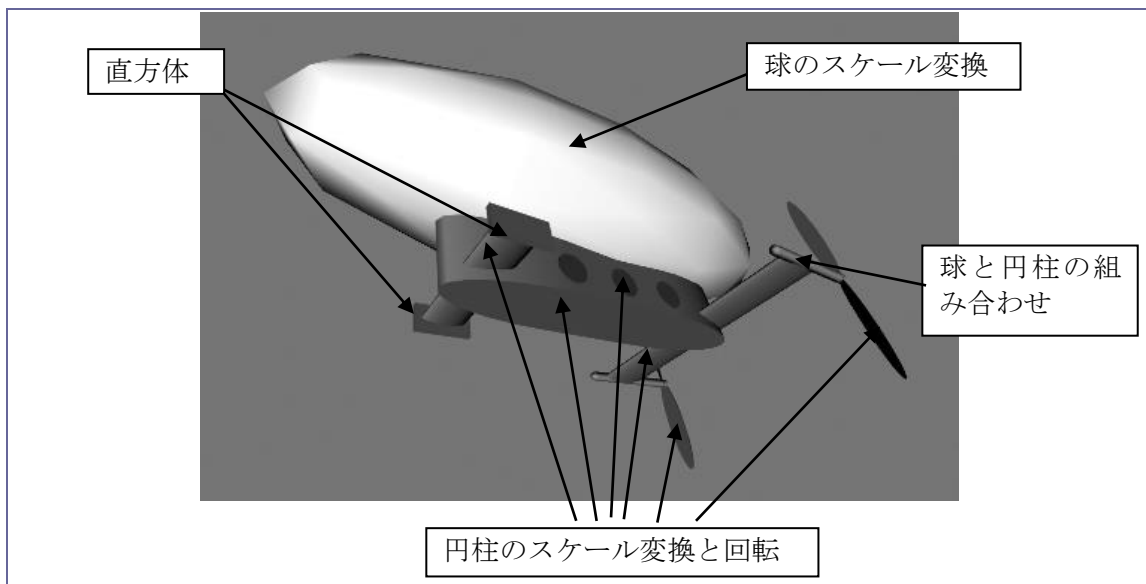


図2 基本的3次元形状の幾何変換だけで作った飛行船

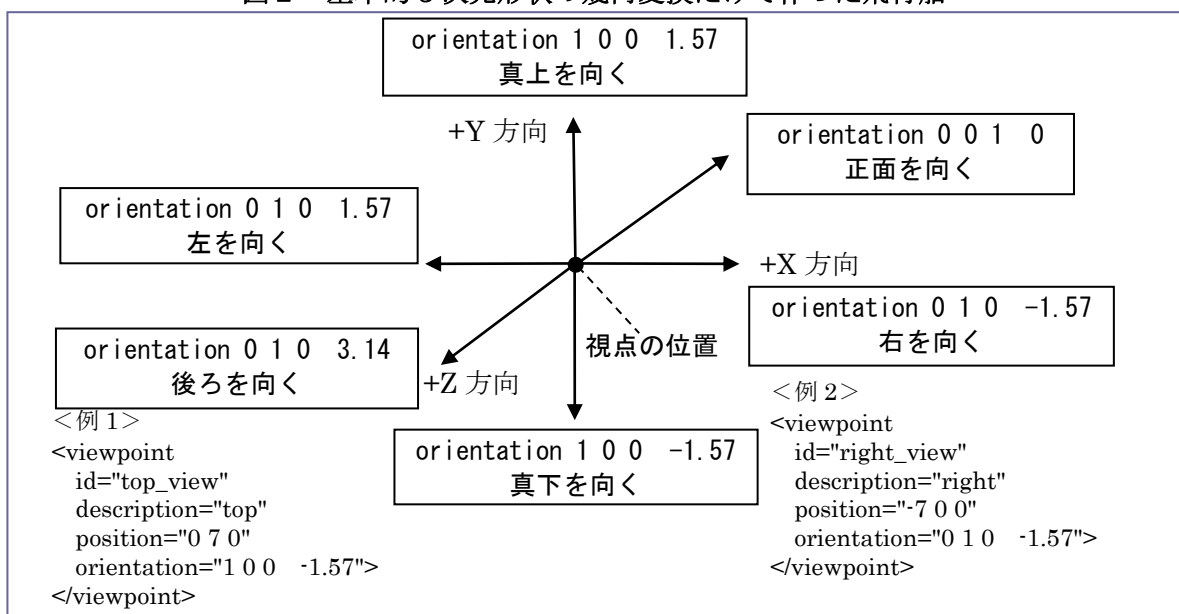


図3 Viewpoint ノードにおける視線方向の記述法

<付録 1> 物体の複雑な回転

任意のベクトル回りの回転は2年後期コンピュータグラフィックスのテキストにも書いてある通り、かなり複雑な式になる。実際、rotation="1 0 -1 1.57" のような回転はなかなか想像しにくいものである。ゆえに、通常は次のように二つの回転を組み合わせて記述する。

```
<transform rotation="0 1 0 0.78">      <!-- 2. 次にy軸周りに45度回転 -->
  <transform rotation="1 0 0 1.57">      <!-- 1. まずx軸周りに90度回転 -->
    <shape>
      <cone></cone>
    </shape>
  </transform>
</transform>
```

<付録 2> 作った物体をまとめて移動したいとき

```
<transform translation="5 0 0">          <!-- まとめてx方向に5m移動 -->
```

```
<transform rotation=".....">
  <shape>
    .....
  </shape>
</transform>

<transform rotation=".....">
  <shape>
    .....
  </shape>
</transform>

.....
```

```
</transform>
```