

令和4年度 卒業研究

西部地区の元町教会群を3Dモデルで再現

3D model of a group of churches in Motomachi, Western District

函館工業高等専門学校

生産システム工学科 情報コース

5年 小濱悠心

指導教員 東海林智也

目次

1. はじめに	2
1.1 研究概要.....	2
1.1.1 和文	2
1.1.2 英文	2
1.2 背景・目的.....	2
2. 研究方法.....	3
2.1 研究の流れ.....	3
2.2 開発環境.....	3
2.3 Blender の操作.....	3
2.3.1 モデリング.....	3
2.3.2 テクスチャマッピング.....	7
2.3.3 ライティング.....	8
2.3.4 レンダリング.....	11
2.4 モデルの作成.....	12
2.4.1 ベースとなるモデル.....	13
2.4.2 ハリストス正教会.....	14
2.4.3 ヨハネ教会.....	15
2.4.4 カトリック元町教会.....	17
3. 研究結果	18
3.1 結果.....	18
3.2 考察.....	18
3.3 展望.....	18
4. おわりに.....	19
4.1 まとめ.....	19
4.2 参考文献.....	19
4.3 謝辞	19

1. はじめに

1.1 研究概要

1.1.1 和文

この研究の目的は、西部地区を舞台としたメタバース空間を開発することである。しかし、西部地区全体を再現するには1年という期間は短く、現実的ではないため、元町の教会群に焦点を当て、3Dモデルで再現することにした。教会群を選んだのは、狭い地域に様々な宗派の教会が集まっていることが、本研究に最も適した場所だからである。なお、本研究で作成した3Dモデルは、CGTrader上にて無償で公開する予定となっている[1]。

1.1.2 英文

The goal of this research is to develop a metaverse space set in West District. However, one year is too short and impractical to reproduce the entire West District, so we decided to focus on a group of churches in Motomachi and reproduce them in a 3D model. We chose the church complex because it is the most suitable location for this study, where churches of various denominations are gathered in a small area. The 3D model created in this study will be made available free of charge on CGTrader[1].

1.2 背景・目的

私たちが住む函館市は観光都市として親しまれ、毎年500万人前後の観光客が訪れている[2]。様々な観光エリアがある函館だが、函館山のふもとに位置する西部地区はその中でも特に人気が高いエリアである。

異国情緒あふれる街並みが広がる西部地区では、歴史的建造物が多く存在しており、どれも函館らしさを兼ね備えている。旧函館区公会堂や赤レンガ倉庫がその代表例である。また開港とともに栄えた西部地区では、1階が和風、2階が洋風といった上下和洋折衷建築や、教会群など西洋の技術を用いた建造物も多く存在している。

そんな魅力あふれる函館市だが、近年の新型コロナウイルスの影響で1年あたりの観光客数が約200万人減少してしまっている[2]。

そこで本研究では、函館市の観光支援につなげることを目的とし、まずは西部地区の元町教会群を3Dモデルで再現する。最終的には、西部地区全体を舞台としたメタバース空間の開発につなげる。

2. 研究方法

2.1 研究の流れ

本研究は、西部地区全体を舞台としたメタバース空間の開発を最終目標としている。これは、生まれ育った西部地区の歴史的雰囲気の良いを知ってもらいたいという思いからと、そのために自分の技術で貢献できる方法を探した結果である。しかし、1年という期間では西部地区全体を再現するにはあまりに短く現実的ではなかった。

以上の点から、まず範囲を元町教会群に絞り3Dモデルで再現することにした。西部地区の歴史的建造物は点在している場合が多く、単純に範囲を狭めても作成できる歴史的建造物は多くて2つほどである。ゆえに、狭い範囲に異なる宗派の教会が密集しているこの場所は本研究の舞台として最適であると判断した。

また、今回作成した3DモデルはCGTraderにて無償で公開することとする[1]。数ある3Dモデル素材サイトからCGTraderを選んだ理由としてはアカウントの新規取得が必要ないことと、公開時のファイル形式に制約が少ないことが挙げられる。

上記にもある通り、今回の3DモデルはBlenderという統合型3DCGソフトウェアで制作している。このBlenderで作成したプロジェクトファイルをそのまま公開できるという点もCGTraderを選んだ大きな理由になっている。

2.2 開発環境

開発環境はSurface Pro7を使用した。また統合型3DCGソフトウェアとしてBlender3.2.0を使用した。Blenderは、3次元のCGアニメーションを作成するために必要な機能が一通り搭載されており、全て無料で利用することができる。しかし、機能が豊富すぎることから、習得が難しく初心者にはハードルが高いというデメリットも存在する。

したがって今回は数ある機能の中から、モデリング・テクスチャマッピング・ライティング・レンダリングの4つに絞った。3Dモデルを作成したあと、テクスチャなどで質感付けをし、ライトでモデルに陰影をつけ、最後に画像として書き出すといった流れになる。

2.3 Blender の操作

2.3.1 モデリング

本研究ではポリゴンモデリングという手法を用いている。建造物をモデリングする際は点、辺、面をしっかりと表現することが重要である。そのため、ポリゴンと呼ばれる多角形を編集しながらモデリングする方法を選択した。まずはこのポリゴンモデリングの基本的な操作方法を解説していく。

また、本研究ではショートカットを使用した操作が主であるため、解説の際もショートカットで解説する。

はじめに画面操作だが、左クリックで選択、右クリックでオプションと、PC操作と同じ感覚で作業できる。ズームイン、ズームアウトはスクロールボタンをスクロールさせる。視点移動は少し特殊で、スクロールボタンを押し込みながらマウスを動かすと視点が回転する。さらに Shift キーを押しながらこの操作を行うと、視点を上下左右に平行移動することができる。また、Ctrl+Z キーで Undo、Shift+Ctrl+Z キーで Redo である。Undo は PC 操作と同じだが、Redo のショートカットはあまり馴染みがないため注意が必要である。

Blender にはデフォルトで平面、立方体、円、UV 球、ICO 球、円柱、円錐、トーラスの 8 つのメッシュが用意されている。円は線のみで構成されており、UV 球はきれいな球、ICO 球はごつごつとした球、トーラスはドーナツのような形状をしている。Shift+A キーでこれらのメッシュを追加することができ、基本的にはこの 8 つのメッシュを形状変化させながら追加し、目的のモデルに近づけていく。本研究では主に、立方体、円、UV 球を使用した。

メッシュの操作は移動、拡大縮小、回転の 3 つが基本である。移動は G キー、拡大縮小は S キー、回転は R キーを押し、マウスを動かすとメッシュを操作できる。また、Blender は X、Y、Z を軸とした 3 次元空間で構成されており、メッシュの操作の際もこの軸を参考にできる。例を挙げると、G キーの後に X キーを押すとメッシュは X 軸上のみを移動する(図 1)。

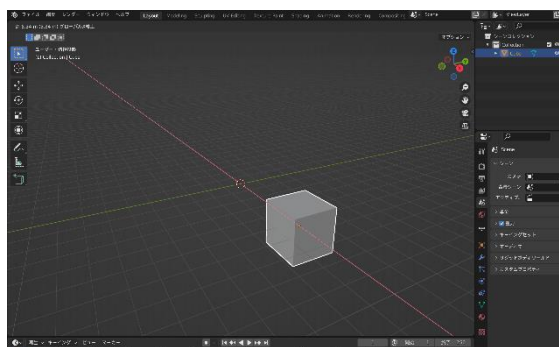


図 1 X 軸上を移動するメッシュ

S キーの後に Y キーを押すとメッシュは Y 軸上のみで拡大縮小する(図2)。

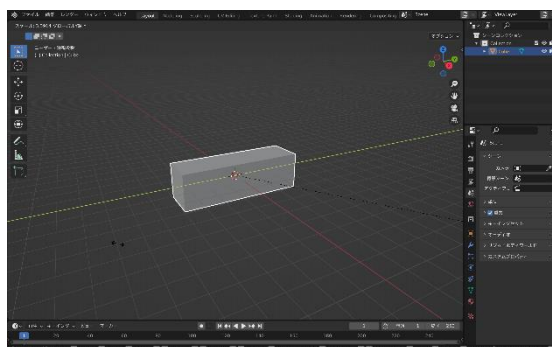


図2 Y 軸上で拡大するメッシュ

R キーの後に Z キーを押すとメッシュは Z 軸を中心として回転する(図3)。

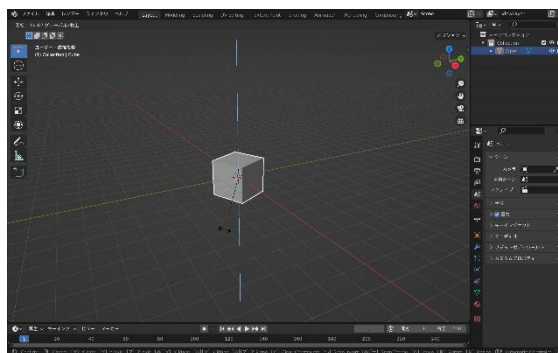


図3 Z 軸を中心として回転するメッシュ

このように操作の種類を選択し、どの軸を基準にその操作を行うか、といった流れでメッシュを変形させていく。また、よく使う操作として複製と削除がある。複製は Shift + D キーで行うことができ、同様に最後に 3 軸のどれかを選択すると、複製されたメッシュを選択した軸上の好きな箇所配置することができる。削除は X キー、もしくは Del キーである。

この基本的な操作のほかに、複雑な変形を行う操作がある。押し出し、インセット、面貼り、ループカット、ベベルの 5 つである。それぞれ順に解説する。

押し出しは選択した面を法線方向に押し出す操作で、E キーで行う(図4)。

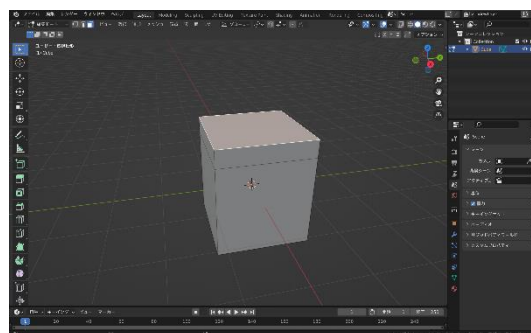


図4 法線方向に面を押し出したメッシュ

インセットは選択した面の中に新たな面を差し込む操作で、I キーで行う(図5)。

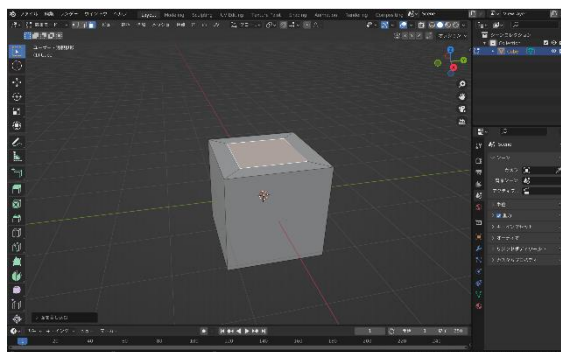


図5 新たな面を差し込んだメッシュ

面貼りは選択した複数の点や線を埋めるように面を作る操作で、F キーで行う(図6)。

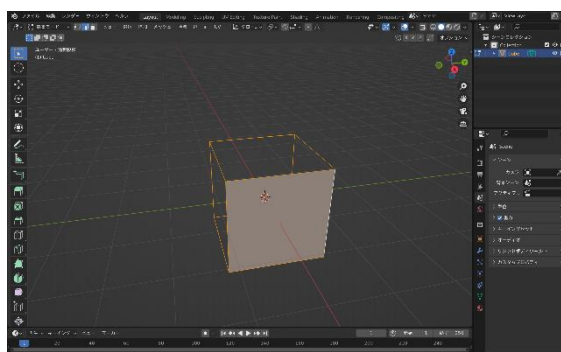


図6 手前の面のみ面貼りを行ったメッシュ

ループカットは面や線を分割するように線を入れる操作で、Ctrl + R キーで行う(図7)。スクロールボタンをスクロールすることで分割数を決めることができ、カットを入れた後にマウスを動かすと、カット位置をスライドさせることができる。

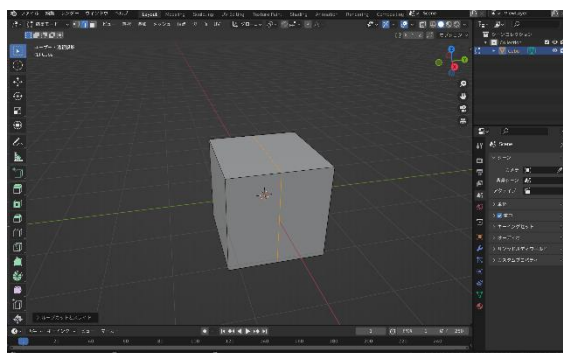


図7 Y軸と垂直にカットしたメッシュ

ベベルは選択した辺に丸みを帯びさせる操作でCtrl+B キーで行う(図8)。スクロールボタンをスクロールすることでセグメント数を決めることができ、セグメント数が多いほどスムーズな丸みになる。

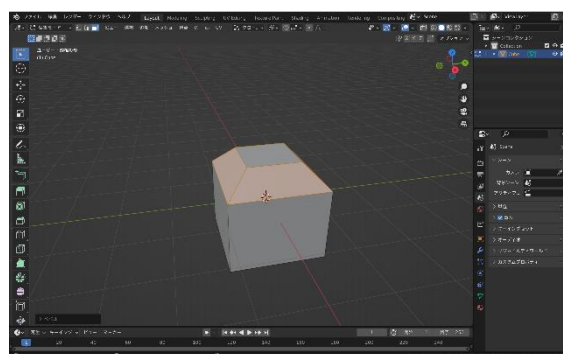


図8 上面にベベルをかけたメッシュ

これらの操作を駆使して、より複雑なモデリングを行っていく。

2.3.2 テクスチャマッピング

モデリングの次はテクスチャマッピングを行う(図9)。テクスチャマッピングとはメッシュの各面に対してテクスチャを貼る作業で、色付けのようなイメージである。メッシュの形状を変えずに見た目を変えることができる為、モデリングで表現できない細かい形状をテクスチャで補う形をとる場合が多い。

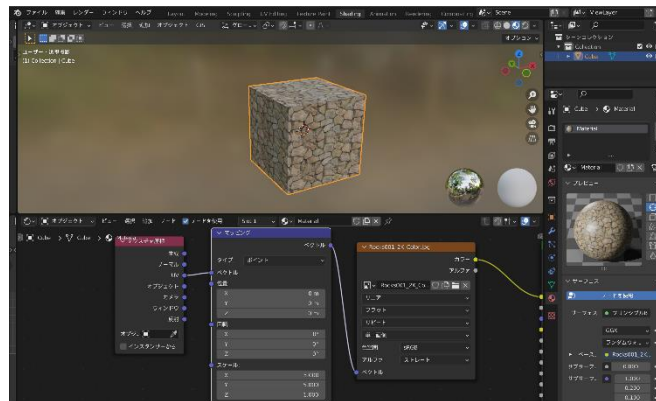


図9 岩のテクスチャを貼ったメッシュ

このテクスチャマッピングを行う際は、ノードと呼ばれるレンダリング実行時の各処理を構成する機能ブロックを使う。テクスチャマッピングの場合、画像テクスチャノード、マッピングノード、テクスチャ座標ノードの3つを使うことが多い。

画像テクスチャノードでは貼りたいテクスチャの画像を選択し、投影方法等を決定する。マッピングノードでは選択したテクスチャのサイズや位置、角度等を調整することができる。テクスチャ座標ノードではテクスチャをどのようにメッシュに投影するかを決めることができる。主に使うのはUVである。

2.3.3 ライティング

モデリングで形を作り、テクスチャマッピングで色や質感付けが終了したら、ライティングを行う。ライティングとは、モデルにライトを当てて陰影をつけることで、よりリアルに表現する作業である。ライティングを適当に行くと現実味のないものになってしまうため、重要な工程である。

Blenderのライトはデフォルトで用意されているため、メッシュと同様にShift + Aキーで追加することができる。また、移動、拡大縮小、回転など、メッシュの基本操作と同様の方法でライトも操作することができる。色や光源の強さなどはオブジェクトデータプロパティと呼ばれるパネルから調整する。

Blenderのライトには4つの種類がある。ポイント、サン、スポット、エリアの4つである。それぞれ特性が大きく違うため、1つずつ解説していく。

ポイントは電球のように、光源から全方向に光を放つ(図10)。光の強さは比較的大きくしないと変化を認識することができない。全方向に光を放つため、Rキーによる回転操作は必要なく、Gキーによる移動操作がメインとなる。

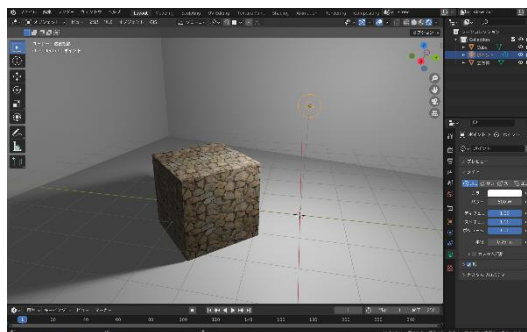


図10 500Wのポイントライト

サンは太陽光のように一律の指向性を持つ(図11)。光の強さは1Wでも十分なほどである。光の届く距離に差異が無いいため、Gキーによる移動操作は必要なく、Rキーによる回転操作がメインとなる。

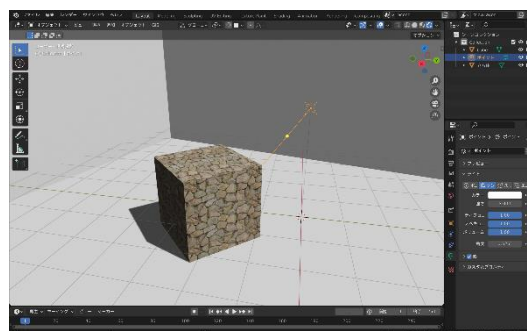


図11 5Wのサンライト

スポットはスポットライトのように円錐状の指向性を持つ(図 12)。光の強さはポイント並みで、G キーによる移動操作も R キーによる回転操作も必要になってくる。また、円錐のサイズ、つまりスポットの開き具合も 1° ~ 180° の範囲で設定する必要がある。デフォルトは 45° である。

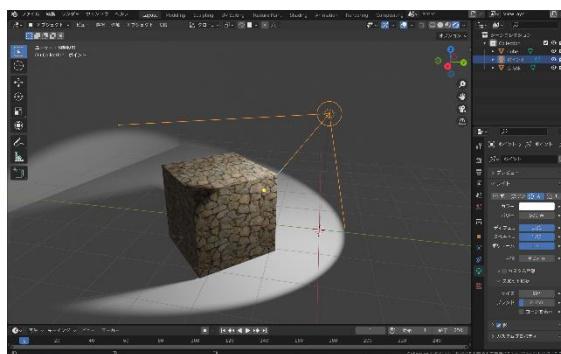


図 12 500W のスポットライト

エリアはテレビや窓のように、光る平面のようなライトである(図 13)。光の強さはポイント並みで、G キーによる移動操作も R キーによる回転操作も必要になってくる。また、光源自体の形状を正方形、長方形、ディスク、楕円の 4 つから選択することができる。デフォルトは正方形である。

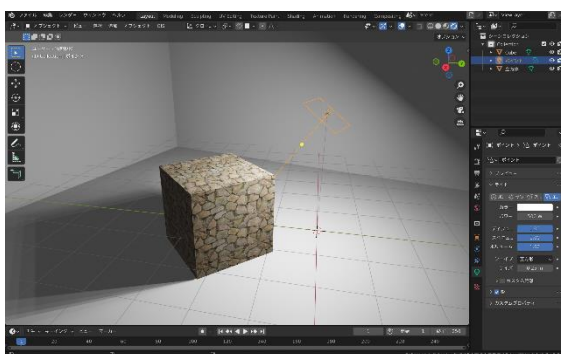


図 13 500W のエリアライト

以上の 4 つのライトを用途に合わせて設定配置していく。

また、全ての面を等しく照らすワールド自体の環境光も設定が必要である。ワールドの明るさはテクスチャマッピングのようにノードを使用して設定していく。使用するノードもテクスチャマッピングと同様に画像テクスチャノード、マッピングノード、テクスチャ座標ノードの 3 つを使うことが多い。テクスチャマッピングと違う所は、テクスチャ座標ノードにつなぐソケットが UV ではなく生成であることと、画像テクスチャノードで使用する画像が単なるテクスチャではなく HDR 画像という全周囲画像をテクスチャのように使用することである。

HDR 画像は光源が再現されているかのように、画像の位置によって明るさが変わる。そのため、マッピングノードで Z 軸上の位置を変えたり、Z 軸を中心として回転させたりすることで、ライティングすることができる。

図 14 では公園のような場所の HDR 画像をワールドに設定している。ワールドを設定する前と比べ、周囲全体が明るくなっていることがわかる。

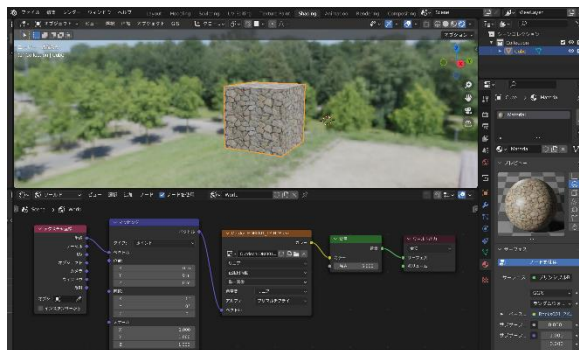


図 14 HDR を設定したワールド

また、背景ノードの数値でワールドの明るさが決まり、画像テクスチャノードと背景ノードの接続を切ると、単色のワールドカラーを設定することができる。

2.3.4 レンダリング

ライティングも終わったら最後にレンダリングをし、画像や動画として書き出す作業がある。レンダリングの際に必要な設定はまず、画像として書き出すのか、動画として書き出すのか、である。ファイルのフォーマットを決め、画像と動画どちらの場合も解像度を設定する。デフォルトは PNG 画像で、解像度が 1920px×1080px となっている。

また、レンダリングの際はカメラを設置する必要がある。カメラはメッシュやライト同様 Shift + A キーで追加することができ、G キーによる移動操作や R キーによる回転操作も同様に行える。

実際にレンダリングされる画像や動画はこのカメラから見た映像になるためカメラ枠は設定した解像度と同じ比率で表示される。出力されるメッシュはこの枠内に収まっているメッシュのみだが、ライトやワールドの明るさ等はカメラの枠外に設置されていてもレンダリング時に影響を及ぼす。

Blender には Eevee、Workbench、Cycles というレンダラーエンジンが用意されており、デフォルトは Eevee である。この 3 つは特徴が大きく異なるためそれぞれ解説する。

Eevee は高速でレンダリングすることができるが、リアリティに欠けるという特徴がある。基本的な質感であれば問題なくレンダリングされるが、デフォルトの設定のままであると光の反射、透過、拡散をうまく表現することができず、CG さがぬぐい切れない画像として書き出されてしまう。

しかし、これは複雑な光の処理を行う際の懸念点であり、フォトリアルを再現したい場合以外は Eevee を使うことが多い。アニメーションなどの動画を作る際も 1 フレームずつレンダリングすることになるため、高速にレンダリングできる Eevee を使うことが多い。

図 15 は Eevee でレンダリングした画像である。これがデフォルトの設定だが、複雑な処理が無い
ため十分リアルに見える。



図 15 Eevee でのレンダリング結果

Workbench はライトやテクスチャが反映されない。つまり、最終レンダリングではなく作業する
際の 3D ビューポート用のレンダーエンジンである。作業中はこのレンダーエンジンを使い、画像
として書き出す際は Eevee か Cycles を使うのが一般的である。

Cycles はレンダリングに時間がかかるが、リアルに表現することができる。動画には向いておら
ず、画像 1 枚を書き出す際は Cycles を使うことが多い。

図 16 は Cycles でのレンダリング結果である。Eevee でも十分リアルな表現ができていたが、
Cycles はよりリアルな表現ができていることがわかる。



図 16 Cycles でのレンダリング結果

2.4 モデルの作成

前述の通り、本研究の 3D モデルは全て Blender で作成している。本研究で使用したテクスチャ
や全周囲画像は ambientCG という著作権フリーの素材サイトのものを利用している[3]。また、無
料で公開されているアドオンと呼ばれる拡張機能をいくつか使用しているが、詳細は次に示す。

2.4.1 ベースとなるモデル

今回 3D モデルで再現する舞台を元町教会群というかなり狭い範囲に絞ったが、それでも一から地形をモデリングするのは気が遠くなる作業である。

そこで Blender-OSM というアドオンを使う(図 17)[4]。このアドオンはオープンデータの地理情報から地形データを取り込むことでメッシュを自動生成するものである。坂や建物の高低差なども再現することができ、指定した範囲の建物も簡易的に生成される。



図 17 Blender-OSM で生成した地形

範囲を指定する際はマップが開かれるため、そのマップ上で四角形を拡大縮小させて範囲を絞る。すると、その範囲を示す座標が表示される。この座標をコピーし、Blender-OSM の範囲パネルにペーストすることで、Blender に指定した範囲が読み込まれる。

範囲を指定できたら、生成するメッシュの種類を選択する。水路や線路など様々あるが、今回は建物、道路、地面を選択した。

ここで簡易生成した民家についてはメッシュ変形することなく、テクスチャのみで壁と屋根を再現する。建物は壁と屋根が統一されたまま生成されるため、真上からの視点で建物すべての面を選択し P キーで分離する。そうすることで建物の壁と、屋根となる上面を別メッシュに分けることができる。

地面、道路に関しても民家同様草やアスファルト、コンクリートのテクスチャで表現し、背景は雲を有する空の全周囲画像を使用した。図 18 は Blender-OSM で自動生成したモデルに各テクスチャを設定しレンダリングした画像である。この段階で既に教会を表す簡易モデルは削除してある。後に各教会をモデリングし、削除した簡易モデルがあった箇所に合成する。



図 18 テクスチャを設定したベースモデル

2.4.2 ハリストス正教会

教会は一見複雑な形状をしているが、部分的に見ると対称であったり同じ形状が何度も使われていたりする。例えば、ハリストス正教会の入口上に位置する円柱部分は八角形になっており、各面に同じ形状の窓がついている。したがって一面だけモデリングし、ミラーモディファイアーでコピーすることにより作業を効率化できる(図19)。これはShift + Dキーで行う複製でも代用できる。しかし、この場合7つ複製しなくてはならない為、ミラーモディファイアーを使用した方が効率良く作業できる。このように、同じ結果になるとしてもそれまでの過程がより効率よくるように知見を増やすことも大切である。

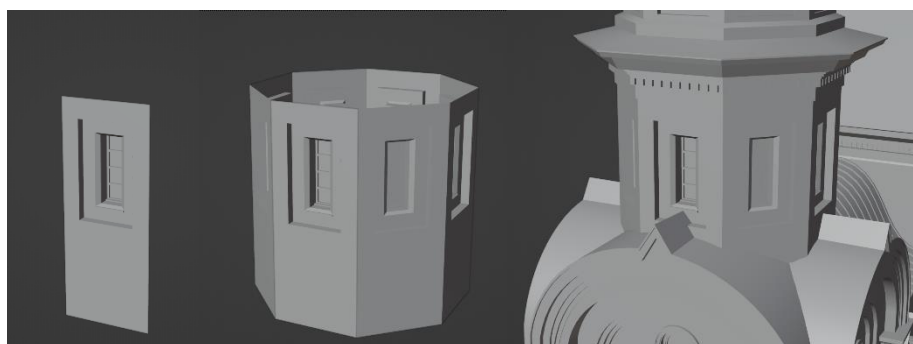


図19 ミラーモディファイアーを使用したモデリングの流れ

また、複雑なモデリングをする際はナイフ投影という操作が重宝される。これは対象のメッシュに別のメッシュのシルエットを投影し、新たな面を形作る操作である。

まず、対称のメッシュを選択し編集モードにする。次にCtrlを押したまま投影したい別メッシュを選択する。その状態でメッシュ→ナイフ投影を選択すると、現在の視点の向きから投影され新たな面ができる。図20はX軸+方向から円を投影した図である。

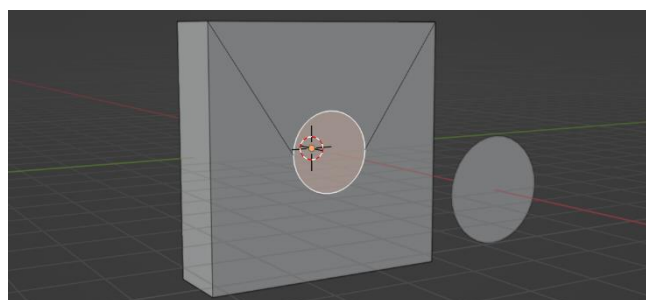


図20 円をナイフ投影した直方体

ナイフ投影を使うときは平行投影に変更することが多い。ほとんどの作業は、透視投影という現実と同じようなビューで行うが、シルエットをそのまま投影するナイフ投影の場合、透視投影のままであると面が歪んでしまうことがある。

図 21 はモデリングが終了した段階の全ての辺を表示した画像である。ミラーモディファイアーを繰り返した際に、同じ面が重なってしまうことがあった。その際はこのワイヤーフレームで表示すると、裏側に隠れた面も選択することができる。

最終的なポリゴン数は 56,173、頂点数は 64,699 であった。

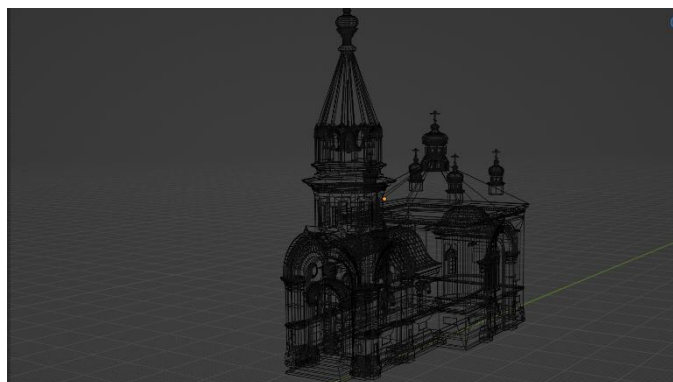


図 21 ハリストス正教会のフレーム画像

作成したモデルを Blender-OSM で生成したベースモデルと合成する。ハリストス正教会はテクスチャを使用せず、質感はマテリアルのみで表現している。太陽光を表現するライトと、反射を表現するライトを配置し、レンダリングした画像が図 22 である。今回は建物の中は再現していない為、窓の中が見える必要はない。したがって窓のマテリアルの透明度は 1 に設定してある。



図 22 ハリストス正教会のレンダリング画像

2.4.3 ヨハネ教会

ヨハネ教会もハリストス正教会と同様にモデリングした。この教会は勾配が急な坂の途中に位置しているため、場所によって高さが変わってしまう。

そのためモデルの底を通常より深くし、地面に埋まるように設置することで高さの差をカバーしている(図23)。

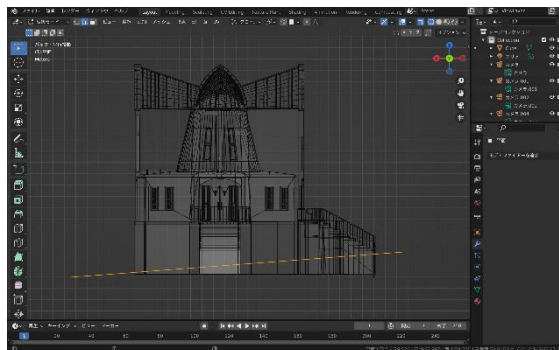


図23 モデルの底と斜面の位置関係

また、ハリストス正教会は中心を軸とし正面からみた線対称であったが、ヨハネ教会は中心を軸とし上空から見た点対称であるため、全て1からモデリングした場合と比べると作業量が1/4で済む。したがってより効率よく作成が完了した。

ハリストス正教会をモデリングした際も使用したが、周辺の植物はbotaniqというアドオンを使用して生成した[5]。このアドオンは無料版と有料版があり、用意された複数のモデルの一部を使用することができる。このアドオンで生成した植物のモデルは、マテリアルが既に設定されているため新たにテクスチャを貼る必要はなかった。

レンダリングした画像が図24である。最終的なポリゴン数は24,724、頂点数は29,522であった。



図24 ヨハネ教会のレンダリング画像

2.4.4 カトリック元町教会

この教会は全体が柵で覆われている。したがって周囲を参考にすることができず、正面のみの作成となった。ただし、見えない部分は想像で補うことで壁や屋根、扉などを違和感なく作成している。

また、この教会はステンドグラスが特徴的であったため、窓の素材の透明度をノイズが無いアルファクリップで0.3に設定することで、うまく表現している。

手前右の建造物はレンガのような壁になっているが、フリー素材のテクスチャは使用しておらずBlenderにデフォルトで用意されているレンガテクスチャというものを使用した。この、Blenderにデフォルトで用意されているテクスチャの中にはノイズテクスチャや、ボロノイテクスチャといったものもあり、これらは汚れなどを再現する際に有効となる。

この土地は斜面を切り崩して平面に整えたような土地であるため、自動生成したベースとなるモデルでは、この箇所がすべて斜面のまま生成されてしまっていた。したがって、斜面を切り崩し平らにする修正は手作業で行った。

レンダリングした画像が図25である。ポリゴン数は28,105、頂点数は38,812であった。



図 25 カトリック元町教会のレンダリング画像

3. 研究結果

3.1 結果

今回作成したモデルはポリゴン数が20,000から60,000に収まっている。ただし、レンダリングエンジンはEeveeを使用したため、植物等の再現が現実味のないものになってしまった。

3.2 考察

今後メタバース空間を作成していく過程で今よりモデリングの範囲を広げていくことになる。そうした場合、現状のままモデリングを続けていくと今のようなポリゴン数の建物が膨大に増えることになり、ファイルサイズがとて大きくなってしまふことが懸念される。

3.3 展望

この課題を解決するためにポリゴン数をより小さく収めていく必要がある。まずモデリングを大まかな概形が捉えられた時点で終了させることとする。その後の細かい箇所はテクスチャなどを使い、モデリングした場合に比べて劣らないよう再現する。そうすることで、クオリティを下げることなくポリゴン数は減らすことができ、より快適で没入感のあるメタバース空間を作り上げることができる。

4. おわりに

4.1 まとめ

本研究では元町教会群のモデリングを行った。各教会のモデルはポリゴン数が 20,000 から 60,000 となり、今後はよりポリゴン数を減らす必要がある。そのため、モデリングを大まかな時点で終了させ細かい部分はテクスチャで再現することとする。そうすることで、より快適で没入感のあるメタバース空間を作成できると考える。

4.2 参考文献

- [1] CGTrader: *Top Page*, <https://www.cgtrader.com/> (2023).
- [2] 函館市: *来函観光入込客数推計*, <https://www.city.hakodate.hokkaido.jp/docs/2015062500021/> (2022).
- [3] ambientCG: *Top Page*, <https://ambientcg.com/> (2023).
- [4] prochitecture: *Blender-OSM: OpenStreetMap and Terrain for Blender*, <https://prochitecture.gumroad.com/l/blender-osm> (2023).
- [5] polygoniq: *Tree Library Botaniq Trees+Plants*, <https://polygoniq.gumroad.com/l/botaniq-trees> (2023).

4.3 謝辞

本研究を進めるにあたり、ご指導くださいました指導教員の東海林智也准教授に感謝いたします。