街路の全方位画像に基づく テクスチャ付き 3 次元建物モデルの自動生成手法の検討

中村 遼斗*・佐藤 剛・小川 芳樹***・前田 紘弥****・関本 義秀*****

Automatic Generation Method of Textured 3D Building Models Using Omni-directional Images of Streets

Ryoto Nakamura*, Go Sato**, Yoshiki Ogawa***, Hiroya Maeda****, Yoshihide Sekimoto***

Abstract: The generation of a detailed 3D city model is an important topic. In general, a large-scale textured 3D city model is created by projecting aerial photographs onto a white box model launched from building footprints. However, in many cases, the textures are low resolution, contain some obstacles, and have no semantic information of building facades. In this paper, we propose a method for automatically creating realistic and noise free building models based on real-world building images. We first extract building parts from omni-directional images of an urban area and decompose them into facade segments like windows and walls. Then we generate texture images from façade crops using a texture synthesis model. By mapping these textures onto 3D models reconstructed from building footprints, we automatically generate a high-quality city model.

Keywords: 3 次元建物モデル（3D building model），全方位画像（omni-directional image），ファサードセグメンテーション（façade segmentation），テクスチャ合成（texture synthesis）

1. はじめに

都市のデジタルツインへの関心の高まりに伴い，3 次元都市モデルの整備が重要視されるようになった．近年は，日本国内でも国家規模のプロジェクトとして都市モデルの開発が進められており，3D モデルとともに様々な都市情報を一元的に蓄積することで，防災やまちづくり，商業サービス等の分野での活用が期待されている．

3 次元都市モデルのフォーマットとして採用されている CityGML では，Level of Detail（LOD）と呼ばれるモデルの詳細度が定義されており，高い LOD の建物モデルを広範囲で整備していくことが論点となっている．例えば，国土交通省が整備している PLATEAU [1] では一部の都市エリアでテクスチャ付きの LOD2 モデルが実装されており，今後その範囲の拡大を目指している．このような大規模の都市モデルの構築においては，航空写真から抽出した建物テクスチャを白箱型の LOD1 モデルに投影する手法が一般的である．しかし，この手法で生成されたテクスチャは解像度が低く，同時に建物以外の障害物の映り込みが懸念される．また、建物要素のセマンティック情報を持たないため、発展的な利用が難しい．

本研究では，車両に搭載した全方位街路画像から抽出した建物画像に基づいて合成されたテクスチャ画像を建物モデルにマッピングすることで，テクス

* 学生会員 東京大学工学系研究科社会基盤学専攻
（Department of Civil Engineering, the University of Tokyo）
〒153-8505 東京都目黒区駒場 4-6-1
E-mail：nakaryo@iis.u-tokyo.ac.jp
** 学生会員 東京大学工学系研究科社会基盤学専攻
（Department of Civil Engineering, the University of Tokyo）
*** 正会員 東京大学空間情報研究センター
（Center for Spatial Information Science, the University of Tokyo）
**** 正会員 東京大学生産技術研究所
（Institute of Industrial Science, the University of Tokyo）

チャ付き建物モデルを自動生成する手法を提案する．車載カメラが撮影した街路の全方位画像から抽出された建物部分を，窓や壁面等のファサード要素ごとに分解し，これらをもとに深層学習によって新たにテクスチャ画像を合成する．さらにフットプリントや階数等の属性情報から 3D モデルを再現し，各セグメントに画像をマッピングすることで，現実の風景に基づいた 3 次元建物モデルを自動生成する．

上記の手法を神戸市内の一部区画に適用し，建物ファサードのセグメンテーションやテクスチャ合成の精度検証及び最終的な出力モデルと Google Earth の比較を行った．

2. 提案手法の構築

本研究の概要は図 2 に示すとおりである．提案手法は主に「ファサードセグメンテーション」，「テクスチャ合成」，「モデリング・テクスチャリング」の 3 段階に分けられる．データに関してはいくつかの事前処理を行い，適切な形式に調節する．

2.1. データの事前処理

本研究では，建物画像として株式会社ゼンリンの全方位画像を使用した．これは図 1（a）のように正距円筒図法の画像となっているため，そのままだと画像の上端と下端ほど大きく歪んだ状態で使用することになる．そこで，photoshop の球パノラマ機能を利用して画像を読み込み，そこから建物の画像を切り出して使用する．今回は建物領域を手動で取り出したが，小川（2021）らの手法のように全方位画像から建物領域を自動で検出する研究もあり，今後このプロセスを自動化することが可能だと考えられる．その後，正確に建物ファサードを抽出するために，Affara（2016）らの手法を使って画像を整形処理する．上下に消失する線分を垂直方向に，左右に消失する線分を水平方向に整列させる変換を求める手法であり，これによって図 1（b）に示すような正面向きの建物画像を得ることができる．最後に，影の影響で建物全体が薄暗く写っている画像が一定数存在するため，手動で輝度の調整を行う．

またフットプリントに関しては，神戸市の建物データを QGIS で読み込み，Shape 形式の建物データや建物の属性情報を csv 形式で出力し利用する．この際，建物のドア位置は道路に最も近い辺にあると仮定し，建物ごとにその辺の両端の頂点情報を付与した．

2.2. ファサードセグメンテーション

建物テクスチャを生成するために，まず初めに建物ファサードの特徴を抽出することを考える．事前処理を行った建物画像から壁面や窓，ドア等のファサード要素ごとのセグメントに分類するために，セマンティックセグメンテーションを行う．この際，余分なオブジェクトを含まず，純粋な壁面や窓の領域を抽出できるよう，19 種類（wall，window，door，garage，stairs，roof，balcony，fence，sky，road，plant，bike，human，car，pole，signboard，vending machine，garbage，outdoor unit）のクラスを設定し細かく分類する．

モデルは Wang （2019）らが提案した HRNetV2 を採用する．既存の研究と異なり，このモデルは高解像度のネットワークと低解像度のネットワークを

（a）全方位画像　　（b）処理後

図 1：全方位画像の事前処理

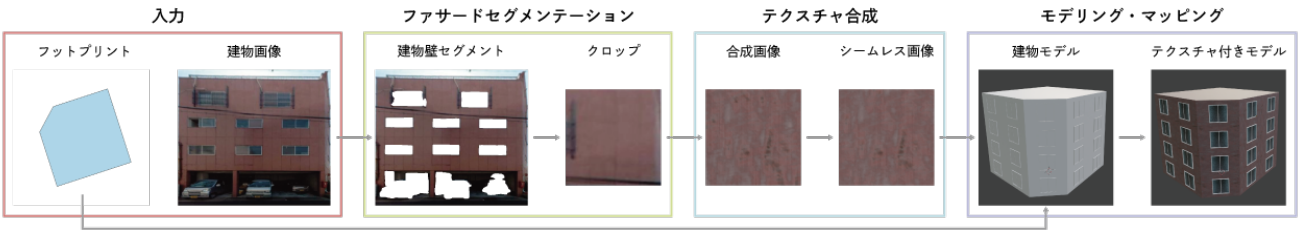

図 2：提案手法の概要，建物画像とフットプリントを入力としてテクスチャ付き建物モデルを出力する

並列につなぎ合わせ，より高い推定精度を発揮するようにしたものである．モデルの学習については，ピクセルごとに各クラスの確率を計算し，負の対数尤度損失を使って最適化する．

この手法によって，図 2 に示すような各クラスのセグメント画像を得ることができる．これらのうち，wall，window，door，stairs，roof 等の建物の基本構造に関わるクラスのセグメント画像について，クラスごとに領域内からランダムに 128 * 128 の画像を 10 枚ずつクロップし，当該クラスのテクスチャの候補とする．つまり，もし 1 枚の建物画像の中に上記クラスがすべて含まれていた場合，10 枚 * 5 クラス分のクロップを抽出する．もしセグメントの領域が狭く上記サイズのクロップを得ることができない場合には，64 * 64 の画像をクロップし，Bicubic 補間で 2 倍にリスケールしたものを代用する．

2.3. テクスチャ合成

前節で作成されたクロップはノイズを多く含み画質が粗いため，テクスチャ画像としてそのまま 3D モデルに適用することはできない．そこで，クロップを入力として新たなテクスチャ画像を生成することを考える．今回は Vidanapathirana（2021）らの提案したテクスチャ合成アプローチを参考にした．この手法では，まずクロップをエンコーダに入力し画像の特徴量をベクトル形式で埋め込む．そしてこの埋め込みをデコーダに通すことで，クロップの特徴を引き継いだテクスチャ画像を合成する．同時に，この埋め込みから対象画像の材質を推定する分類器を実装し，埋め込みの潜在空間を材質ごとに構造化する．今回は建物壁等の材質の候補として挙げられる 3 種類のクラス（wood，plastered，bricks）を設定し，分類器は埋め込み情報から画像がいずれのクラスなのかを推定する．

このエンコーダ・デコーダで構成されるモデルは，画像ピクセルに関する VGG ベースの損失関数とテクスチャの材質に関するクロスエントロピー損失で学習させる．前者については，クロップと生成画像をそれぞれ VGG に入力した際に出力される特徴マップの MSE によって計算する．後者については，トレーニング画像にラベル付けされた材質クラスと分類器が推定したクラスとの間のクロスエントロピー誤差によって計算する．

訓練済みのモデルに対象の建物のクロップを入力し，合成画像を得る．ただし，各建物セグメントにつき 10 枚のクロップが存在するため，このうちクロップと合成画像の間の VGG 損失が最も小さいものを代表とし，建物モデリング時のテクスチャとして採用する．

最後に，このテクスチャ画像をそのまま張り付けると，画像のつなぎ目が不連続になり境目が浮き上がってしまう．そこで Embark Studios [2)] の手法を使い，事後処理として画像のシームレス加工を行う．

2.4. 建物のモデリングとテクスチャリング

建物の 3D モデルの作成には Blender を使用する．Blender では GUI の操作プロセスを Python で記述することが可能であり，今回は全モデリングプロセスを自動化する．まず，QGIS から出力した建物の頂点座標のリスト及び建物属性情報から，白箱型のモデルを立ち上げる．この際，建物の高さは 3 m * 階数として計算する．簡単のため，今回の実験では屋根やバルコニー等の建物要素は考慮せず，箱形の建物本体に窓とドアを附属させたモデルを作成する．

今回行うファサードセグメンテーションでは，各建物要素の個数や壁面上での相対的な位置等を推測するわけではないため，現実の建物の各要素の位置を 3D モデル上で厳密に再現することはできない．したがって，一定のアルゴリズムに従って各要素を配置する．窓については，建物の属性情報のうち，用途が独立住宅ならば窓サイズが 2 m * 2 m で窓間隔が 1 m，それ以外（つまり集合住宅等の大規模なものや商業利用の建物）ならば窓サイズが 3 m * 2 m で窓間隔が 1.5 m になるようにする．ドアについては，頂点情報から読み込んだドアのある面上に 1 m * 2.5 m のドアを設置する．

最後に 3D モデルにテクスチャ画像をマッピングする．今回の実験では，壁面テクスチャには 2.3 節で合成した画像を採用するが，窓とドアに関しては事前に用意したテクスチャ画像を使用する．これは，

窓やドアには映りこみが生じていたり，セグメントの面積が小さいために十分なクロップを得ることができなかったりしており，うまくテクスチャ画像を合成できないためである．

3. 使用データ

3.1. 対象領域と建物画像

神戸市内の街区領域から図 3 に示す約 100 m 四方の区画を設定し，本実験の対象とした．当該エリアの全方位画像のうち，各建物を最も正面に近い位置から撮影したものを選定し，2.1 節で述べた事前処理を施すことで、各建物に対して一枚ずつの画像を得た．この領域には 81 棟の建物が存在するが，うち図 3 中の灰色の建物が示す 19 棟からは画像を得ることができず，建物カバー率は 77%となった．これは，道路に面しておらず他の建物の裏側にある，もしくは屏などで隠れてしまっていることが原因で，全方位画像中に写っていなかったためである．

建物のフットプリントには，神戸市の都市計画基礎調査のデータを使用した．各建物のフットプリントに対して，階数や用途等の詳細な属性情報が紐づいており，建物の白箱型モデルの作成時に利用した．

3.2. セグメンテーションモデルの訓練画像

ファサードセグメンテーション用の訓練データとしては，神戸市の建物画像 192 枚とオープンデータである eTRIMS Image Database 60 枚を組み合わせて使用した。神戸市の画像に関しては、前節で述べたものとは別のエリアの道路画像を用意し，同様の事前処理を施して建物画像を用意した。各画像について，2.2 節で挙げた 19 クラスでラベリングしたアノテーションデータを作成した．

図 3：対象領域の建物フットプリントおよび画像撮影位置

3.3. テクスチャ合成モデルの訓練画像

テクスチャ合成モデルのトレーニングにはオープンソースのテクスチャ画像を使用した．建物画像から切り出したクロップを直接学習に使わない理由は，画像内に影や反射によるノイズが含まれているため学習が安定せず，非定常なテクスチャを生成しやすくなるからである．

Pixels 等のオンライン上の画像ライブラリ [3] から 2.3 節で述べた 3 種類のクラスに分類できる画像を 206 枚ダウンロードし， 512 * 512 のサイズに加工してトレーニング画像とした．

4．実験結果と考察

4.1. セグメンテーション精度の検証

ファサードセグメンテーションの結果，ピクセルベースの Accuracy が 86.82 %となった．また，すべてのクラスの結果を平均した Mean IoU が 0.340 となった．

代表的な要素についてクラス別の IoU を表 1 に示した．wall が高い値になっているが，これはひとつのセグメント内で大きな色の変化がなく検出しやすいのに加え，最も画像中に占める割合が大きく，学習材料が豊富だったからだと考えられる．一方，window や door の精度は，ほぼいずれの画像にも含まれているメジャーな要素にもかかわらず，50 %程度にとどまった．窓に関しては，ガラスへの映り込みが激しいケースが数多くあり，セグメント内でピクセル値の大きな変動が生じてしまっていたことが原因だと考えられる．ドアに関しては，ガラス製

表 1：クラス別の IoU

クラス	wall	window	door	roof
IoU	0.864	0.521	0.509	0.173

（a）元画像　　（b）真値　　（c）推定結果

図 4：建物画像のセグメンテーション結果

（a）成功例　　（b）失敗例

図 5：テクスチャ合成の結果，成功例（左）と失敗例（右）

のものが一定数あったため，窓への誤認識が生じてしまっていた．さらに， roof 等のトレーニング画像に含まれている数が少なかったマイナーな要素についてはかなり値が低くなった．

また，図 4 は視覚的な結果の例である．建物壁や上部のエッジが高精度で検出されていることがわかり，IoU の数値と一致する．また，雨戸がしてある窓はきちんと検出できているにもかかわらず，映り込みや張り紙がしてある窓は正確に識別できていないことが確認された．

本論文では 252 枚の画像でモデルを学習させたが，これはセマンティックセグメンテーションに用いられる学習画像の枚数としては少ない．したがって，今後より大量の画像で学習を行うことで各要素の抽出精度を上げることができると考えられる．

4.2. テクスチャ合成の結果の検証

図 5 に壁面のテクスチャ合成の結果を示した．(a) が成功例を示しており，左からオリジナルの建物画像，抽出したクロップ，合成されたテクスチャ画像である．比較的シンプルな構造の建物のため正しく壁面をクロップできており，その結果実際の壁面の色や垂直方向のアーキテクチャが再現され，かなり実物に近いテクスチャ画像が生成された．

一方（b）に失敗例を示した．建物の壁面は白であるにもかかわらず，合成された画像は青灰色になった．これは，建物の手前にある歩道橋の領域を壁面と認識し，そこからクロップを取ってしまったためである．このように，障害物が原因でセグメンテーションがうまくいかず，その結果全く異なる画像を生成してしまうケースや，壁面の色味は継承されたものの，オリジナル画像の解像度が足りないために模様が再現されないケースなどが確認された．前者に関しては，ファサードセグメンテーションの精度の向上や，障害物を含まない建物画像の選択等で発生を抑制できると考えられる．

4.3. 建物モデルの可視化

図 6 に対象領域の Google Earth と生成された 3D モデルの比較を示した．3D モデルは CityEngine のマップ上に表示したものである．建物画像を得ることができなかった内側の建物はテクスチャなしの白色で表示した．全体的に明度が異なってはいるが，現実と似た色彩を再現できている建物が多数存在していることがわかる．また，窓等の数や位置を比べても大幅な差異は見られず，本研究で採用した配置アルゴリズムに妥当性があることが確認された．以上の点から，提案したテクスチャ付き建物モデルの

(a) 生成された 3 次元建物モデル　　(b) 同一地域の Google Earth

図 6：3 次元建物モデルの生成結果と Google Earth の比較

自動生成手法には一定の有効性があると考えられる．

一方で，側面によって壁の色が異なる建物がいくつか存在しているが，本手法では建物の各ファサード要素とテクスチャ画像が一対一対応なので，面による違いは再現できていない．今後は，建物ごとに複数のテクスチャ画像を用意することで，面による建物の見え方の違いを考慮する必要がある．また，屋根の形状や材質を無視しているため，真上から見た際の色は現実の建物と大きく乖離してしまっている．衛星画像を使って屋根のテクスチャを合成することができれば，より複雑なモデリング・テクスチャリングが可能になると考えられる．さらに，本研究では建物要素に抽出にセマンティックセグメンテーションを利用したが，インスタンスセグメンテーション等を行うことによって，壁面における窓やドア一つ一つの位置を特定ができ，より正確な要素配置ができるようになると考えられる．

5. おわりに

本研究では，建物画像をファサード要素ごとにセグメント化し，各セグメントのクロップからテクスチャ画像を合成し，3 次元建物モデルにマッピングすることで，テクスチャ付き建物モデルを自動生成する手法を提案した．

今後は，訓練画像の枚数増加によるセグメンテーション及びテクスチャ合成の精度向上や屋根等の複雑な要素のモデリング手法の開発，各要素の個別の位置を考慮した建物オブジェクトの配置に取り組むことで，より現実に近い 3 次元建物モデルの生成を目指す．

謝辞

株式会社ゼンリンには本研究で用いた全方位画像データを提供して頂いた．また，都市計画基礎調査は G 空間情報センターより提供して頂いた．関係各位に謝意を表する．

注釈

1）https://www.mlit.go.jp/plateau/

2）https://github.com/EmbarkStudios/texture-synthesis/

3）https://www.freepik.com/，https://www.pexels.com/

参考文献

Affara, L., Nan, L., Ghanem, B., and Wonka, P. (2016) Large Scale Asset Extraction for Urban Images. ECCV 2016, 437–452.

Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., and Xiao, B. (2019) Deep high-resolution representation learning for visual recognition. TPAMI 2019.

Vidanapathirana, M., Wu, Q., Furukawa, Y., Chang, A. X. , and Savva, M. (2021) Plan2scene: Converting Floorplans to 3D Scenes. CVPR 2021.

Korč, F., Förstner, W. (2009) eTRIMS Image Database for Interpreting Images of Man-Made Scenes. Dept. of Photogrammetry, University of Bonn, Tech. Rep.

小川芳樹・沖拓弥・陳聖隆・関本義秀（2021）街路の全方位画像と建物 GIS データの結合手法．第 30 回地理情報システム学会講演論文．