Delft University of Technology
Faculty of Architecture and The Built Environment
Department of Urbanism

Master of Science Graduation Program

Enriching the 3DBAG with Roofing Material Types

P2 Assessment Erratum

Dimitrios Mantas
5836670

19 February 2024
Delft, The Netherlands

1 Introduction

The spatial distribution of roofing materials is becoming increasingly relevant in the context of urban policy- and decision-making (Wyard et al., 2023). The motivation behind this phenomenon can be attributed to four main driving forces: the protection of public health [e.g., by identifying asbestos-containing materials (Abbasi et al., 2022)], the improvement of the overall quality of life of citizens [e.g., by exploring the potential correlation of material type and condition with economic indicators for target social welfare policy implementation, mitigating the urban heat island effect (Phelan et al., 2015), etc.], the reduction of the urban environmental footprint (e.g., by mapping materials which are technically difficult to down- or re-cycle), and the optimisation of urban planning and development activities (e.g., disaster response and management operations, identification of cultural heritage sites, etc.). In this context, this thesis aims to enrich the 3DBAG, an open-source three-dimensional (3D) building model dataset of the Dutch building stock, by extending its existing data attribute catalogue with the material type of the corresponding semantic surfaces.

Despite the above-established significance of roofing material classification, the main body of related literature is generally limited in number, scope, and impact, primarily due to the intertemporal absence of reference datasets to train state-of-the art models, which are capable of achieving relatively greater class coverage and performance. This obsercation is supported by Abriha et al. (2018) as well as numerous expert interviews conducted by the author.

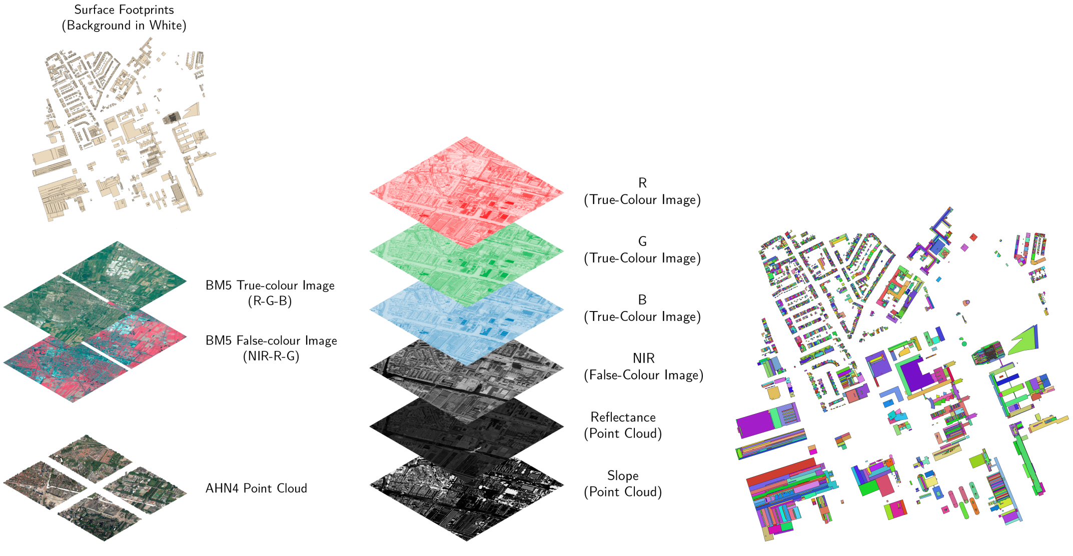
This considered, the main research question to be answered is:

How can the roof surfaces of the 3DBAG models be classified according to their material type such to produce the most added value for policy- and decision-makers, as well as practicioners regardless of sector?

In combination with an extensive literature review of this field, this question has guided the design of a pertinent methodology (Section 2). The proposed approach to the problem is based on the semantic segmentation of appropriately fused aerial imagery and light-detection-and-ranging (LiDAR) derived data to produce high quality roofing material maps for a variety of material types.

2 Methodology

The proposed methodology is designed to automatically generate roofing material maps for individual 3DBAG tiles (i.e., collections of neighboring building models). In general, given the roof surfaces of a tile, relevant data is extracted from nationwide true- and false-color orthophotographs (BM5), as well as the AHN4 point cloud. Subsequently, this data is appropriately fused and passed as input to a machine learning model for which finally produces the maps.(Figure 1).



(a) (b) (c)

Figure 1: Example input (a), intermediate (b), and final (c) output of the proposed pipeline.

This considered, the methodology is logically divided into two separate but semantically interconnected pipelines, depending on the operational status of the model. In particular, the *inference pipeline* (Section 2.1), describes the generation of material maps assuming a trained model, whereas the appropriately named *training pipeline* (Section 2.2), entails the actual training process.

2.1 Inference Pipeline

The inference pipeline entails the generation of the roofing material map for a given 3DBAG tile, assuming the downstream existence of a trained machine learning model which is capable of performing this task. In general, given the identification code (ID) of this tile (e.g., 9-284-556), it is first downloaded and its roof surfaces are extracted (Section 2.1.1). Next, the corresponding AHN4 and BM5 data is downloaded, appropriately parsed (Sections 2.1.2 and 2.1.3), and subsequently merged into a *raster stack*, a multi-band raster dataset (Section 2.1.4). The stack is then appropriately postprocessed and finally passed as input to the model for inference (Section 2.1.5). The main steps of this pipeline are outlined in Figure 2.

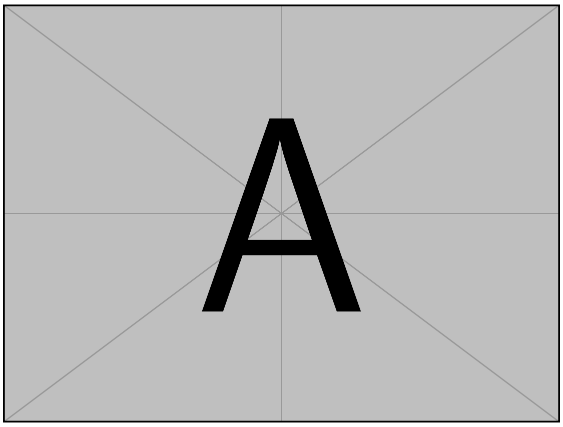


Figure 2: Flowchart of the inference pipeline of the proposed methodology.

2.1.1 3DBAG Data Downloading & Parsing

Given the ID of the tile, it is first downloaded in CityJSON format. Subsequently, the LoD 2.2 representations of its roof surfaces are extracted, projected to two-dimensions (2D), and saved to disk as a tile-specific GeoPackage file (Figure 3).

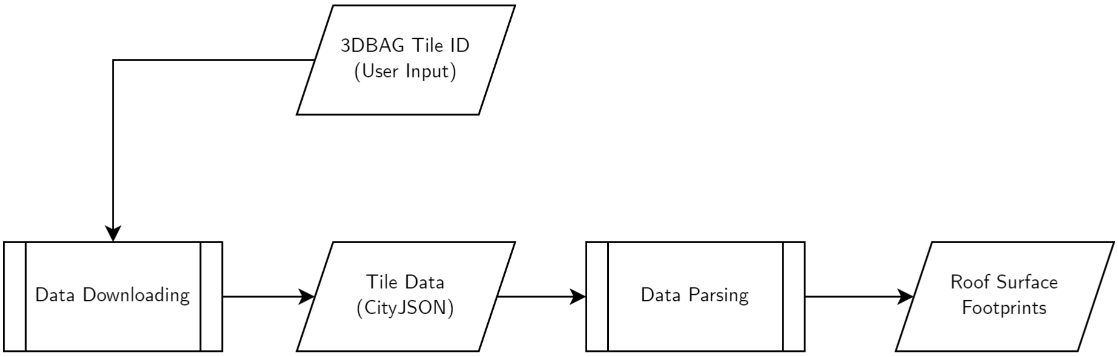
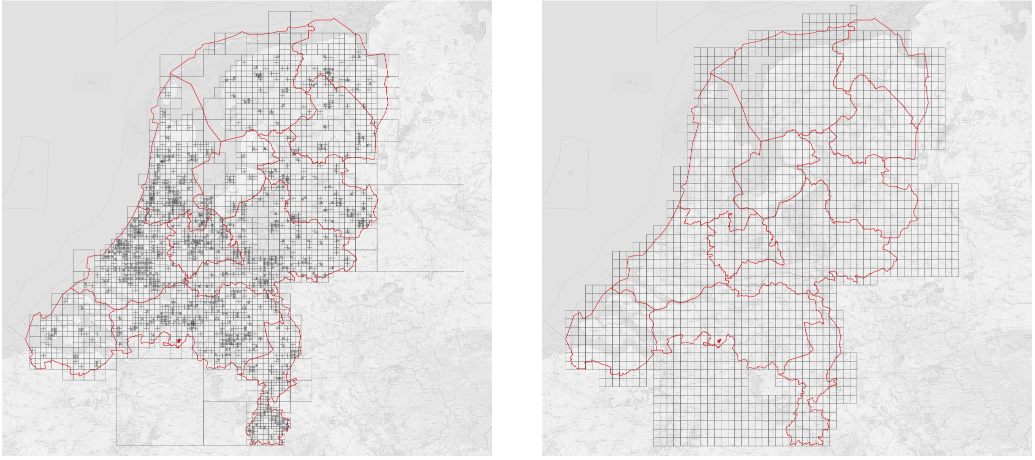


Figure 3: Flowchart of the 3DBAG data downloading and parsing stage of the inference pipeline.

The choice of the particular LoD is due to the underlying assumption that it is the most aligned with the corresponding AHN4 and BM5 data, both geometrically and semantically. In particular, empirical evidence collected by the author in the context of this thesis has shown that the discrepancy between the LoD 2.2 3DBAG models, defined as the root mean squared error between them and the AHN data used to generate them, is minimal in comparison to the other available geometric representations, namely LoD 1.1 and 1.3. Furthermore, it should be noted that maps for any given LoD can be trivially transformed into those representing a lower one, as is the case with geometry. However, the reverse operation is impossible.

2.1.2 BM5 Data Downloading & Parsing

Given the roof surface footprints of the tile, the corresponding BM5 data is first downloaded in GeoTIFF format. This data includes true- and false-color summer orthoimagery with a spatial resolution of 0.25 m and is served as individual 5 × 6.25 km tiles as part of the GeoTiles project, courtesy of the Optical and Laser Remote Sensing group of the Department of Geoscience and Remote Sensing, Faculty of Civil Engineering and the Geosciences, Delft University of Technology. At this point it should be noted that it is possible for the roof surfaces to spatially intersect multiple tiles, because the 3DBAG and GeoTiles sheet indices are not aligned with each other (Figure 4).



(a)

(b)

Figure 4: The 3DBAG (a) and GeoTiles (b) sheet indices. Administrative boundaries are delineated by red polygons.

Once all relevant BM5 tiles have been downloaded, they are cropped to the bounds of the surfaces and merged. The red and green bands of the compound false-color imagery are discarded. Finally, the resulting data is saved to disk as tile-specific GeoTIFF files (Figure 5).

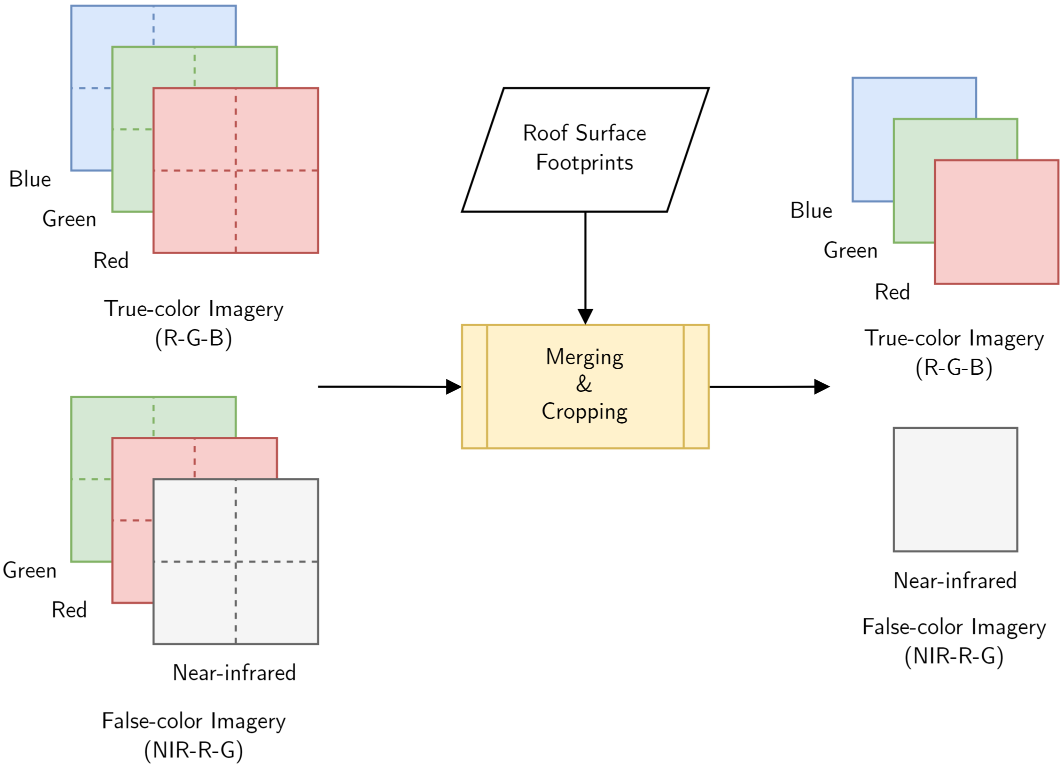
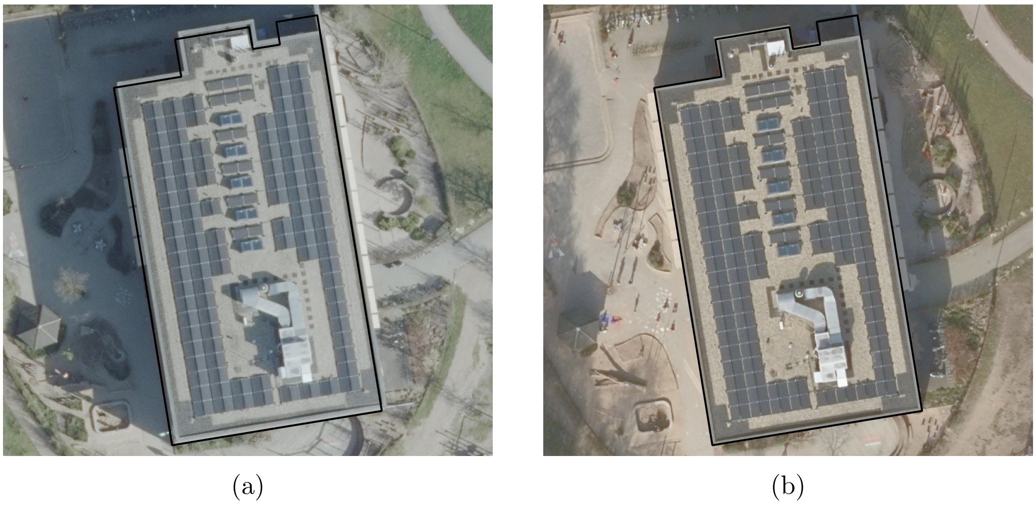


Figure 5: Flowchart of the BM5 data parsing process of the corresponding stage of the inference pipeline. Individual tiles are delineated by dashed lines.

The cropping operation is performed in such a way that all cells intersected by the surface geometry are included in the output dataset. Furthermore, appropriate steps are taken during the merging operation to ensure that this dataset is aligned with its constituents at a pixel level.

At this point it should be noted that although the publicly available BM5 data is corrected for ground deformities, meaning that objects appear as if the ground surface were flat, it still suffers from parallax distortion artifacts (Figure 6). These effects result in objects appearing increasingly tilted with respect to the focal point of camera sensor the further they are from it.


(a)

(b)

Figure 6: Example building footrpint (i.e., the black polygon) superimposed on the corresponding 8 cm, true-color BM5 ortho- (a) and true orthoimagery[1] (b). There is significant data misalignment along the north side of the footprint.

Because the location of this point in space is not constant, as the camera is mounted on a moving airplane, and since the object tilt direction depends on both its distance and height from the aforementioned point, a significant number of surfaces may not be aligned with the corresponding AHN4 data. Eliminating these artifacts requires dense stereo matching using the BM5 stereo imagery, which are not open-source data. Therefore, this issue is an inherent limitation of the methodology.

2.1.3 AHN4 Data Downloading & Parsing

Given the roof surface footprints of the tile, the corresponding AHN4 data is first downloaded in LASzip format. This data represents a LAS 1.4 point cloud with an average point density of ten points per square meter. The point records of this cloud include the fields specified by Point Data Record Format 8 of the ASPRS LAS specification, as well as three variable length records, namely intensity, amplitude, and reflectance, with the latter being a range-normalised measure of the titular physical material property. This data is served as individual 1×1.25 km tiles as part of the GeoTiles project but, in contrast to the BM5 imagery, the corresponding tiles overlap with their adjacent neighbors by 20 m to allow for massive parallel processing.

Once all relevant AHN4 tiles have been downloaded, they are cropped to the bounds of the roof surfaces and merged, with duplicate points due to the above-mentioned overlap removed. This process entails the sorting of all point records of the cloud in descending elevation order and the subsequent elimination of identical entries according to their planimetric records, preserving the last one in each duplicate set. This approach ensures

[1] The true orthoimage is courtesy of the 3D Geoinformation group of the Department of Urbanism, Faculty of Architecture and the Built Environment, Delft University of Technology.

that the locally highest points, which are naturally the most relevant in the context of roofing material classification, are preserved. The use of point records instead of coordinates ensures that only fixed-point comparisons are performed, hence resulting in increased computational efficiency and numerical stability. In addition, the output of this operation is guaranteed to be the same regardless of input, because the transformation from records to coordinates is linear. The main steps of this process are outlined in Figure 7.

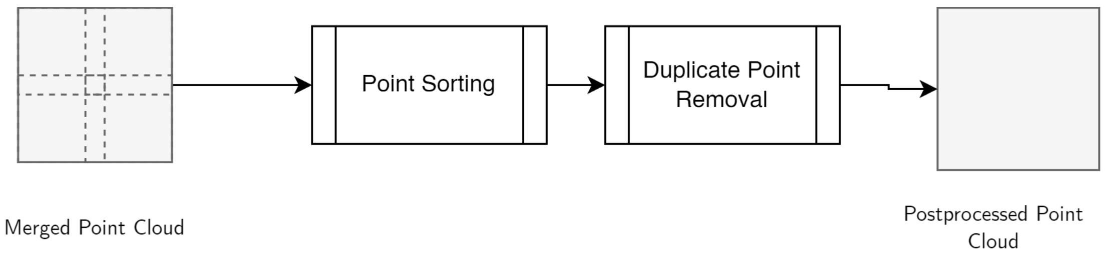


Figure 7: Flowchart of the duplicate point removal process of the AHN4 data parsing stage of the inference pipeline. Individual tiles are delineated by dashed lines.

At this point it should be noted that this process disturbs the spatial coherence of the underlying point, and is thus an inherent limitation of the proposed methodology.

Next, the elevation and reflectance fields of the postprocessed point cloud are rasterised to a 0.25 m cell grid of size equal to that of the parsed BM5 data[2] using inverse distance weighting interpolation (IDW) with a radius and power of $0.25\sqrt{2}$ m and 2 m, respectively. Subsequently, empty cells are filled with an additional, targeted IDW pass over the resulting rasters with a radius of 100 m. The slope of the elevation raster, which is commonly known as the digital surface model of the scene, is then extracted with a second-order central differencing scheme, except along its edges, where first-order forward and backward differences are used (Figure 8).

[2] The BM5 data downloading and parsing stage is a prerequisite of this process.

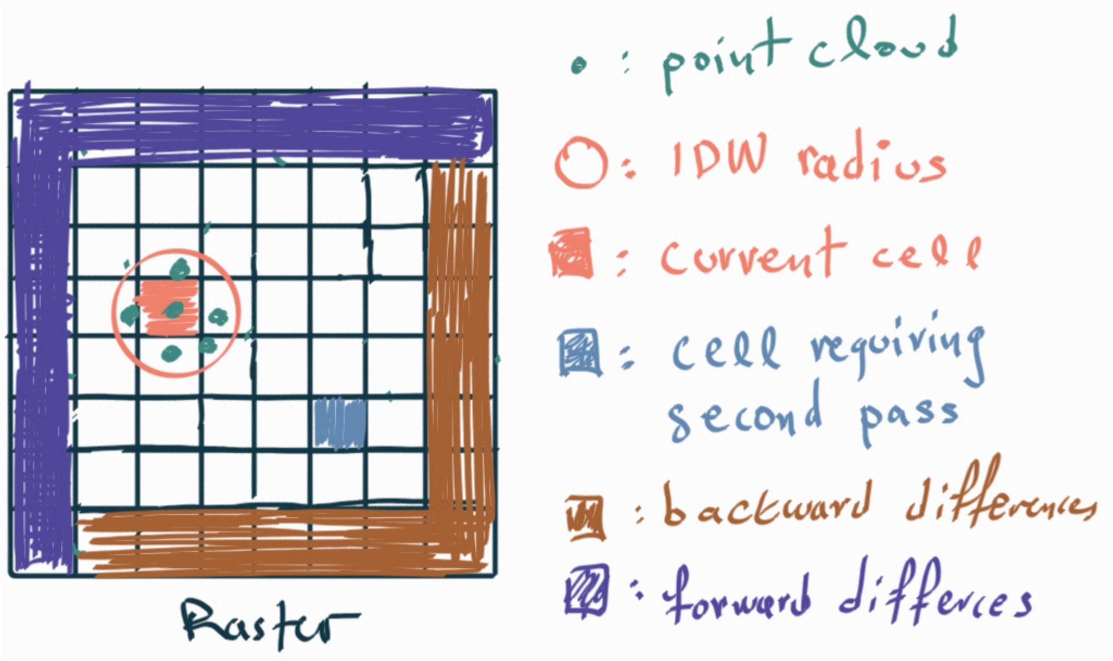


Figure 8: Flowchart of the rasterisation process of the AHN4 data parsing stage of the inference pipeline.

Finally, the output data is saved to disk as a tile-specific LASzip file. The main steps of this stage are outlined in Figure 9.

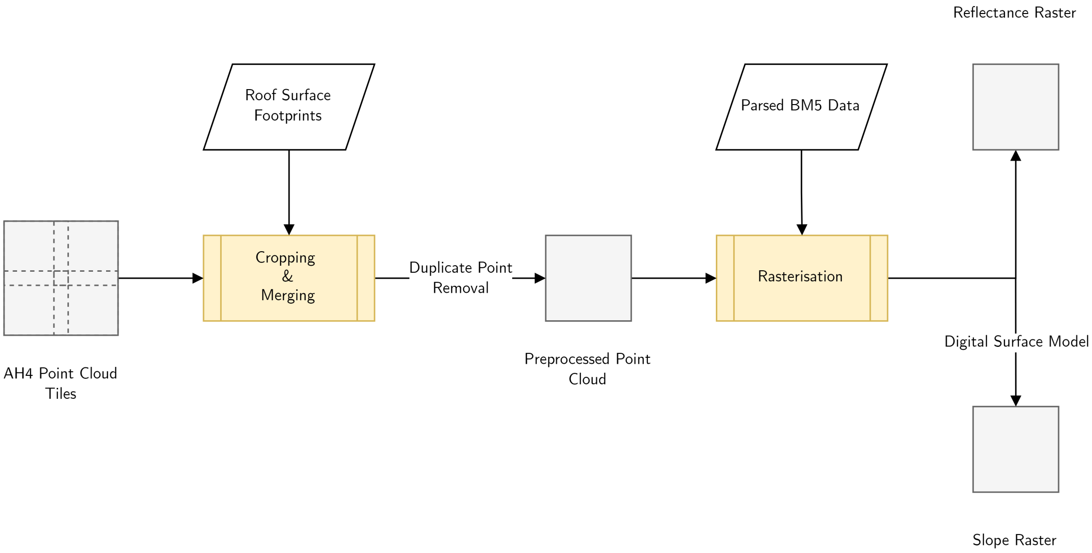


Figure 9: Flowchart of the AHN4 data parsing process of the corresponding stage of the inference pipeline.

2.1.4 Raster Stack Generation

Once the BM5 and AHN4 data has been downloaded and parsed, the resulting data is concatenated along the spectral axis to create a raster stack with the following channel configuration: red, green, blue, near-infrared, reflectance, slope
At this point it should be noted that reflectance data, which in the context of LiDAR is measured in decibels relative to a reference target of specific appearance, texture, and orientation, is converted to the corresponding optical amplitude ratio during this operation. This approach ensures that the corresponding band interval contains exclusively positive numerical values, and thus allows for zero-valued cells to be ignored by the model during training. The relevance of this technique is discussed in Section 2.2.1.2. In general, these pixels represent contextually irrelevant areas, i.e., regions on the exterior of the surface footprints, henceforth collectively referred to as the *background*, which are ignored by the model during training to increase its overall performance.

2.1.5 Roofing Material Map Generation

Once the raster stack has been generated, it is first masked using the roof surface footprints such that the background cells across all its bands are identified and assigned a value of zero. The mask used for this process is composed by dissolving and buffering the footprints by one percent of the minimum planar dimension (i.e., width or height) of the rasters the underlying model is trained on. Next, the masked stack is passed as input to the model for inference, which in turn produces the corresponding material map in chips of size equal to that of its training data. Subsequently, these patches are merged, remasked to remove background predictions, and then denoised using a single-pass median filter with a kernel size of three. Finally, the resulting data is saved to disk as a tile-specific GeoTIFF file.

2.2 Model Training Pipeline

This section describes the process of training a machine learning model to generate the roofing material map for a given 3DBAG tile. In general, the reference dataset to be used for model training, validation, and testing is first generated, given its desired size. This stage entails iteratively sampling 3DBAG tiles, producing, appropriately postprocessing, and splitting the corresponding raster stacks into square chips of a particular size, content, and structure (Section 2.2.1). These patches comprise the *reference dataset*. Next, these patches are manually annotated with a representative set of material classes to produce the corresponding segmentation masks, which serve as example material maps for the model (Section 2.2.2). These masks are single-band, 8-bit rasters of the same size as the relevant chips whose cell values represent material classes using one-hot decimal encoding. Subsequently, these masks are duly postprocessed to increase their contextual value (Section 2.2.3), and passed as input to the model for training (Section 2.2.4). Finally, the performance of the model is evaluated according to various statistical performance indicators and expert opinion.(Section 2.2.5).

2.2.1 Reference Dataset Generation

2.2.1.1 3DBAG Tile Sampling

To ensure that the reference dataset is representative of all geographic regions and material classes the model will be trained on, and thus minimize the probability of potential statistical bias issues, a multistage random sampling approach is used to generate it. This approach is a variation of that proposed by Stewart et al. (2023) and entails the following iterative process:

1. Randomly sample one of the Dutch cities with a population of at least one hundred thousand people, henceforth referred to as *seed cities*, from a uniform distribution.

2. Randomly sample a 3DBAG tile within 15 km from the characteristic geographic coordinates (e.g., the geographical or population center) of this city from a normal distribution.

3. Assuming that the tile has not been selected before, generate the corresponding raster stack, postprocessed, and split it into chips.

4. Add these patches to the reference data pool.

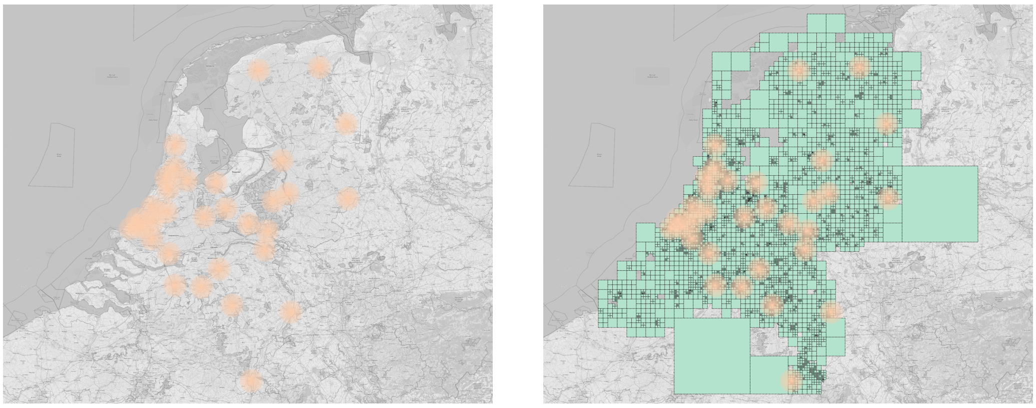
5. Repeat until the pool reaches the required size.

The main steps of this stage are outlined in Algorithm 1.

Algorithm 1: GenerateReferenceData($\mathcal{I}, C, N$)

```
Input  : The 3DBAG sheet index, $\mathcal{I}_{\text{3DBAG}}$, a collection, $S$, of the
         characteristic geographic coordinates of seed cities, the required
         reference data pool size, $N$.
Output: The reference dataset, $\mathcal{D}$.
$\mathcal{D} \leftarrow \emptyset$
$T \leftarrow \emptyset$
while $|\mathcal{D}| < N$ do
    $\mathsf{s} \leftarrow$ Draw a uniformly distributed sample from $S$.
    // Generate a offset vector from the seed.
    // NOTE: This covariance matrix ensures that only ca. 0.3% of
    //       sampled vectors require normalisation.
    $\mathbf{o} \leftarrow \mathcal{N}\left(\mathbf{0}, \operatorname{diag}\left(\frac{1}{9}, \frac{1}{9}\right)\right)$
    if $\|\mathbf{o}\| > 1$ then
        $\mathbf{o} \leftarrow \frac{\mathbf{o}}{\|\mathbf{o}\|}$
    end
    $\mathbf{t} \leftarrow \mathsf{s} + 15000\mathbf{o}$              // Scale the offset to a 15 km radius.
    $\mathsf{Tile} \leftarrow$ Compute the spatial intersection of $\mathcal{I}_{\text{3DBAG}}$ and $\mathbf{t}$.
    if $\mathsf{Tile} \notin T$ then
        $\mathsf{Chips} \leftarrow$ Postprocess and split the corresponding raster stack.
        $\mathcal{D} \leftarrow \mathcal{D} \cup \{\mathsf{Chips}\}$
        $T \leftarrow \mathcal{D} \cup \{\mathsf{Tile}\}$
    end
end
```

At this point it should be noted that material variability is assumed to be most prominent near urban centers. This hypothesis is assumed to be valid given the fact that building use in increasingly rural areas is commonly limited to only certain types (e.g., commercial, industrial, etc.) due to pertinent zoning laws. Because the construction materials and numerous elements of such building types are generally standardised (e.g., due to prefabrication), it follows that they are of little contextual added value to the model. In addition, this approach caters to the fact that the tiling structure of the 3DBAG is proportional to building density. In fact, the selected sampling radius results in a tile coverage of approximately 40%, while an increase of 5 km or circa 33% results in it reaching over 70% (Figure 10).



(a)

(b)

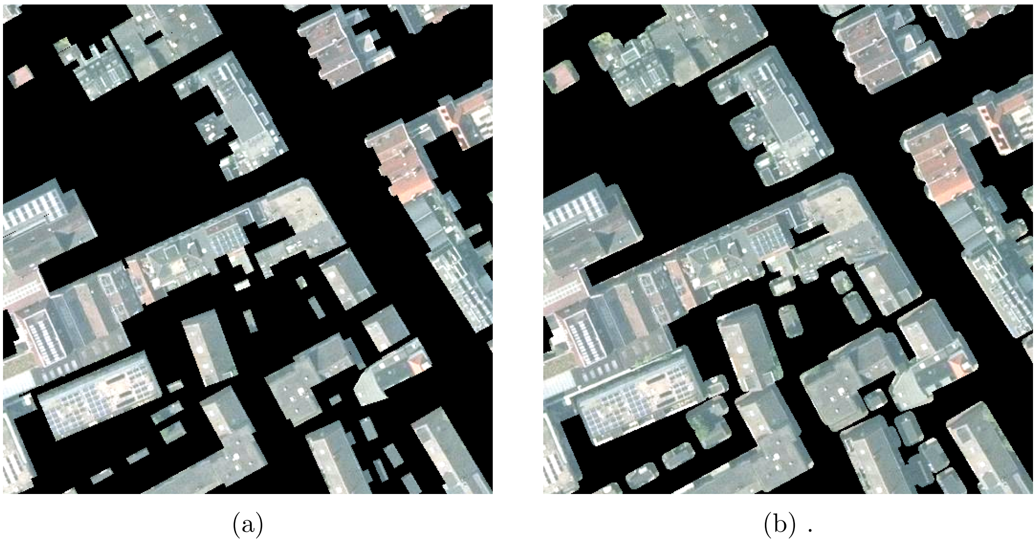
Figure 10: Spatial (a) and 3DBAG (b) coverage. Individual tile sampling regions are delineated by orange regions.

2.2.1.2 Raster Stack Postprocessing & Splitting

To ensure that raster stacks exclusively contain contextually relevant information, they are masked using the corresponding roof surface footprints before being passed as input to the model for either inference or training, such that the background cells across all their bands are assigned a value of zero. This approach allows for the background to be ignored during training, hence avoiding potential model confusion and class imbalance issues, were the background considered as a separate "material" class. However, ignoring the background, which is, by definition, adjacent to regions of interest can result in decreased predictive power along their boundaries due to oversmoothing effects. In addition, material maps generated by the model must be remasked to remove background predictions because the nature of convolutional neural networks prohibits their suppression otherwise.

This considered, the masked stack is subsequently split into 512 × 512 pixel chips. At this point it should be noted that this operation is not strictly necessary to train the model. Actually, it is possible to configure it in such a way that tiling is performed extemporaneously on a stack or even patch level so that they are divide into subsegments. However, pre-training patching is beneficial because it enables more efficient annotation (Section 2.2.2) and granular control over the reference data. In particular, chips with a background content of 50% or larger are discarded from the following stages of the pipeline.

Once the remaining patches have been identified, they are saved to disk as tile-specific GeoTIFF files. To mitigate the oversmoothing effects potentially introduced by background exclusion, a variation of these chips, henceforth referred to as *chip* or *patch buffering*, is also then generated by dissolving and buffering the surface footprints by 1.28 m (i.e., 1% of a length of 512 pixels with a spatial resolution of 0.25 m; Figure 11), and masking them. The buffered patches, which are in fact the ones passed to the model for training (Section 2.2.5), are finally saved alongside their original equivalents.



(a) (b) .

Figure 11: Example raster stack chip (a) and its buffered equivalent (b). Only the true-color bands of the stack are rendered for visualisation purposes.

2.2.2 Reference Dataset Annotation

Once the reference dataset has been generated, its original (i.e., not buffered) chips[3] are manually annotated with the following material classes: asphalt shingles, bituminous membranes, clay tiles, loose gravel, metal, solar panels, and vegetation.

An ad hoc class is used to label cells designating materials which do not belong to any of the aforementioned classes (e.g., asphalt, non-bituminous coatings, concrete or gravel tiles, glass, etc.). This class also includes roof openings and superstructures (e.g., open-air atria, chimneys, heating, ventilation and air conditioning systems, etc.). In addition, a special class represents ambiguous and irrelevant pixel regions such as the background and non-roof building surfaces (e.g., facáde segments), which may remain present after the previous stage due to relevant data distortion and misalignment issues (Figure 5). These regions are henceforth collectively referred to as *invalid*.

To accelerate the annotation process, a third-party, human-in-the-loop software solution is used. This software, IRIS, courtesy of the Φ-lab of the European Space Agency, is designed to facilitate the annotation of Earth observation imagery using a gradient-boosted random forest to generate segmentation masks based on user-provided examples. In addition, IRIS can be served as a remote web application, where multiple annotators can work on the same dataset seperately from each other, and thus statistically compare and finally merge their masks into potentially more accurate products.

At this point it should be noted that although it will eventually be ignored by the model during training, the background must be still annotated due to technical limitations of IRIS, namely the inability to exclude certain pixel values from the process. Hance, the invalid class is used to temporarily represent it because it is semantically the

[3] The original chips are used exclusively for this stage.

most in agreement with it. In actuality, background cells are relabelled with a unique, internal class during the following stage (Section 2.2.3), because the model is required to learn invalid class to be able to recognise said data quality issues. Furtermore, material identification is currently performed exclusively by visual interpretation, and is thus based on the opinion of the annotator. Hence, it is possible that true material information is not eventually conveyed to the model, particularly in cases of high inter-class ambiguity due to factors such as suboptimal image quality (e.g., reduced contrast, under- or over-exposure, imbalanced histogram, etc.), material color and condition, etc. This issue can introduce human bias into the training process and cause model confusion, and is therefore an inherent limitation of the proposed methodology.

2.2.3 Segmentation Mask Postprocessing

Once the segmentations masks corresponding to the reference dataset have been produced, they are used to generate the henceforth called *buffered masks*, to accompany the buffered chips of the corresponding reference dataset at the training stage (Section 2.2.5). This process is exactly the same as chip buffering (Figure 12).

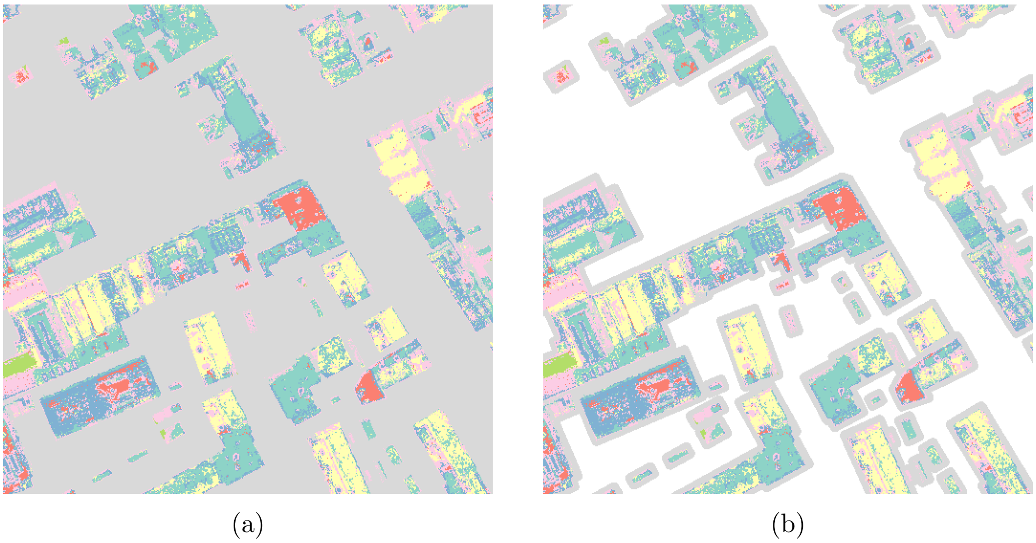


Figure 12: An arbitrary segmentation mask (a) and its buffered equivalent (b). The background and invalid classes are visualised in gray and white, respectively.

In addition, masks marked by the annotator as “noisy” are convoluted using a 3 × 3 median kernel.

2.2.4 Reference Data Splitting & Augmentation

This stage is still under active research. Once the reference dataset and the corresponding segmentation masks have been generated, they are randomly split into training (70%), validation (15%), and test (15%) subsets. In addition, the following training-time augmentations are applied on a per-batch basis:

1. image normalisation per channel to the range $[0, 1]$;
2. random image and mask rotation between 0 and 360 degrees with a probability of 50%;
3. random image and mask vertical flip with a probability of 50%;
4. random image and mask vertical flip with a probability of 50%;

2.2.5 Training, Validation &Testing

2.2.5.1 Model Architecture & Loss Function Selection

This stage is still under active research. In general, an appropriate model architecture and loss function, as well as their corresponding hyperparameters, where applicable, will be identified through experimentation in the following weeks.

The architectures to be explored are those of a traditional feed-forward fully convolutional network (FCN), U-net (Ronneberger et al., 2015), DeepLabv3 (Chen et al., 2017), and DeepLabv3+ (Chen et al., 2018). ResNet 18 and 50 will be tested as candidate backbones for the latter three architectures. In addition, pretraining will be explored using appropriate ImageNet (Deng et al., 2009), Sentinel 2 encoder weights (Mañas et al., 2021; Stewart et al., 2023). The choice of particular architecture at any given stage of research will depend on data availability at that time. Testing will begin with an FCN variation which is yet to be determined. In case this search space is deemed to be inadequate or incomplete for any reason, architectures from relevant works, e.g.,those of Krówczyńska et al. (2020); Santos et al. (2023), and Wyard et al. (2023) will be identified and appropriately adapted and integrated into the proposed methodology, and their applicability will be evaluated in terms of their performance (Section 2.2.5.3). At this point it should be noted that modelling approaches used in related fields such as building deliniation from aerial and satellite imagery land use and land cover classification, and urban scene segmentation will be studied but only considered for adoption if strictly necessary. This is because their added value in the context of this thesis is diminished due to the general lack of comprehensive training datasets, which they oftentimes assume.

The explored loss functions will be focal loss (Lin et al., 2017), the Jaccard, also known as the intersection over union (IoU) index. The background class will be ignored from all relevant calculations.

2.2.5.2 Training & Validation Parameters

This stage is still under active research.Training will be performed using either the Adam optimiser or stochastic gradient descent with sn adaptive learning rate. In addition early stopping will be implemented based on the validation loss with a patience of 10 epochs. Training will be conducted for a maximum of 1000 epochs or until early stopping.

2.2.5.3 Performance Evaluation

This stage is still under active research.Perfomance will be evaluated using the following metrics: average and overall accuracy, mean IoU (mIoU), precision, and recall. In addition, a class confusion matrix will be provided such that the user can compute any other metric of intersest. In addition, the map produced by the model for a particular 3DBAG tile with buildings of known roofing material surfaces will be manually checked for obvious errors, as a qualitative meausre of user's accuracy.

3 Future Work

Immediate uture work includes:

- Examining the replacement of the spatial index used for rasterisation with an STR packed R*-tree (instead of the binary tree used curently) for increased query performance.
- Implementing a multithreaded rasterisation version of IDW (right now it is single-threaded and poses a major bottleneck (e.g., 30 minutes per tile or 50M points))
- Implementing photometric augmentaions on the true-color bands of the chips used for training. The slope and reflectance bands should not be changed because they represent ground and material information, respectively. The applicability of near-infrared-level augmentations will be examined.
- Ensuring that the augmentations do not distort the mask (e.g., with interpolation when rotating).
- producing stict annotation guidelines and procedures so that masks can be standardised.
- Examining if IRIS can be modified to exlude the background at the annotation stage.
- Setting up IRIS on a public server so that the annotation can be partly croud-sourced and thus accelerated.
- Gathering more training data

References

M. Abbasi, S. Mostafa, A. S. Vieira, N. Patorniti, and R. A. Stewart. Mapping roofing with asbestos-containing material by using remote sensing imagery and machine learning-based image classification: A state-of-the-art review. *Sustainability*, 14(13), 2022. ISSN 2071-1050. doi: 10.3390/su14138068. URL https://www.mdpi.com/2071-1050/14/13/8068.

D. Abriha, Z. Kovács, S. Ninsawat, L. Bertalan, B. Balázs, and S. Szabó. Identification of roofing materials with discriminant function analysis and random forest classifiers on pan-sharpened worldview-2 imagery – a comparison. *Hungarian Geographical Bulletin*, 67(4):375–392, Dec. 2018. doi: 10.15201/hungeobull.67.4.6. URL https://ojs.mtak.hu/index.php/hungeobull/article/view/1082.

L. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017. URL http://arxiv.org/abs/1706.05587.

L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *CoRR*, abs/1802.02611, 2018. URL http://arxiv.org/abs/1802.02611.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009. doi: 10.1109/CVPR.2009.5206848.

M. Krówczyńska, E. Raczko, N. Staniszewska, and E. Wilk. Asbestos—cement roofing identification using remote sensing and convolutional neural networks (cnns). *Remote Sensing*, 12(3), 2020. ISSN 2072-4292. doi: 10.3390/rs12030408. URL https://www.mdpi.com/2072-4292/12/3/408.

T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017. URL http://arxiv.org/abs/1708.02002.

O. Mañas, A. Lacoste, X. Giró-i-Nieto, D. Vázquez, and P. Rodríguez. Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data. *CoRR*, abs/2103.16607, 2021. URL https://arxiv.org/abs/2103.16607.

P. E. Phelan, K. Kaloush, M. Miner, J. Golden, B. Phelan, H. Silva, and R. A. Taylor. Urban heat island: Mechanisms, implications, and possible remedies. *Annual Review of Environment and Resources*, 40(1):285–307, 2015. doi: 10.1146/annurev-environ-102014-021155. URL https://doi.org/10.1146/annurev-environ-102014-021155.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. URL http://arxiv.org/abs/1505.04597.

C. L. B. Santos, R. P. Medina, and J. V. Taylar. Classification of roof construction materials using satellite images with convolutional neural network. In *2023 International Conference on Digital Applications, Transformation & Economy (ICDATE)*, pages 1–5, July 2023. doi: 10.1109/ICDATE58146.2023.10248935.

A. J. Stewart, N. Lehmann, I. A. Corley, Y. Wang, Y.-C. Chang, N. A. A. Braham, S. Sehgal, C. Robinson, and A. Banerjee. Ssl4eo-l: Datasets and foundation models for landsat imagery, 2023.

C. Wyard, H. Fauvel, B. Palmaerts, B. Beaumont, and E. Hallot. From dl approach conception to operational product design : identifying roof materials for policy makers. In *2023 Joint Urban Remote Sensing Event (JURSE)*, pages 1–4, May 2023. doi: 10.1109/JURSE57346.2023.10144142.