Rooftop HVAC equipment detection from aerial imagery

Neil Chen
Stanford University
Department of Computer Science
nqc@stanford.edu

Abstract

Building operations contribute to 40% of US energy-related CO2 emissions, primarily from HVAC systems. Identifying and enhancing HVAC efficiency is crucial for emission reduction. We introduce a transfer learning approach to object detection for rooftop units. We demonstrate that convolutional neural networks can be adapted to identify rooftop HVAC equipment given building imagery. Quantitative and qualitative evaluations demonstrate the effectiveness of this approach despite a wide range of equipment types and occlusions. Code and data are available at `https://github.com/NWChen/rtus`.

1. Introduction

Our problem is automatic detection of rooftop HVAC equipment, also known as rooftop units (RTUs) from aerial imagery. Building operations contribute to 40% of US energy-related CO2 emissions, primarily from cooling/heating loads handled by HVAC systems. Emissions reductions globally rely nontrivially on identifying and enhancing HVAC equipment efficiency. Automatic detection and/or identification of rooftop HVAC equipment can accelerate efforts to retrofit and upgrade the existing HVAC equipment fleet, as well as evaluate the distribution of RTU equipment currently in use.

HVAC equipment for commercial buildings is often located on these buildings' rooftops. Imagery/photography of these rooftops is often available via satellite or aerial sensing. Rooftop HVAC equipment comes in many shapes and sizes, and can be easily confused for other mechanical, electrical, or plumbing (MEP) equipment visible from a rooftop. Additionally, while rooftop imagery is fairly abundant, few labelled datasets exist for this type of equipment. Rooftop HVAC equipment is often a similar color relative to the rest of a given roof, and can be occluded by other MEP features.

The input to our algorithm is overhead rooftop imagery generated by remote sensing equipment such as satellite photography. In particular, our algorithm accepts rooftop imagery that has already been cropped to a given building or built environment feature. We make this design decision because there already exists extensive literature concerning the extraction of individual buildings or building rooftops from aerial imagery, but comparatively little work has been published for identifying specific features on such rooftops. We use a variety of convolutional neural network architectures designed for object detection to output predicted bounding boxes and/or masks over rooftop units.

We evaluate a transfer learning approach based on the Faster-RCNN object detection architecture. There is no existing state-of-the-art for the specific inputs and outputs of this problem. We use common object detection metrics to evaluate the effectiveness of our approach. Quantitative and qualitative evaluations demonstrate the effectiveness of our approach despite a wide range of equipment types and occlusions. We also demonstrate the effectiveness of our approach on a small dataset.

2. Related Work

We review related work in two domains: rooftop extraction and rooftop object detection.

2.1. Rooftop Extraction

This problem can be decomposed into two unique problems: rooftop extraction and RTU detection. Rooftop extraction is the more well-researched problem. Wu et al [20] used a heuristic-based approach to remove vegetation and terrain regions from aerial stereo images with digital surface models (heightmaps), isolating rooftops in the process. Abraham et al [1] used a similar heuristic-based approach but also evaluated the effectiveness of road detection and a mean-shift algorithm to identify individual rooftops.

Li et al [11] developed a deep generative adversarial network to attempt building extraction from remote sensing imagery. In this approach, a DenseNet[8]-based generator produces image classification maps, while a discriminator learns structural features. This approach overcomes some problems caused by spatial inconsistency in overhead imagery. Gao et al [6] framed rooftop extraction as an instance

segmentation problem and used a Mask R-CNN based approach to segment rooftops from aerial orthoimagery.

More recent work in deep-learning-based computer vision approaches has also been applied to the rooftop extraction problem. For example, Wang et al [19] implement a vision transformer approach to achieve state-of-the-art performance (IoU) for building extraction in large remote sensing imagery. This approach addresses the computational complexity of a vision transformer-based approach using a dual-path structure: the model encodes spatial details in one context path, and global dependencies in a global context path. With more than 300 million existing buildings across the world, considerations for computational cost are very important.

Buildings can exhibit unique shapes depending on locale and imagery approach. For example, low-density residential housing tends to exhibit a distribution of rooftop shapes that is dissimilar to the distribution of high-density commercial building rooftops in urban environments. Sun et al [18] propose a revised U-Net model to specifically extract roofs in rural areas from satellite imagery. Orthographic aerial imagery was used as input to obtain roof area, ridgeline (highest point), and orientation using a U-Net convolution neural network trained on imagery for northern China. Combined approaches separately tailored to the residential and commercial sectors may best handle the unique distributions in building contours for each building use type.

2.2. Rooftop Object Detection

Building rooftops can contain many types of equipment observable from overhead imagery, including but not limited to solar panels, antennae/satellite dishes, HVAC equipment, skylights, water towers, and more. An example is shown in **Figure** 1 Because of the potential impact of evaluating solar potential, methods for segmenting/detecting photovoltaic equipment on rooftops are by far the most well-researched among rooftop equipment.

Malof et al [14] gathered 100 unique residential buildings from publicly available satellite orthoimagery, of which 50 contained a rooftop photovoltaic installation and 50 did not. They implement a support vector machine classifier with a radial basis function kernel to detect rooftop panels, achieving a 94% recall on this dataset. The simplicity of this approach implies benefits for model interpretability, but is limited to the very small input dataset used.

Li et al [10] evaluated a combination of SVM and CNN approaches to automatically segment photovoltaic equipment on rooftop imagery. Given a dataset of 269,632 satellite images across the US, this approach achieves a Matthews correlation coefficient (MCC) of 0.17, outperforming existing pre-trained CNN approaches.

Solar potential estimation is a similar task blending the problems of rooftop extraction and photovoltaic equipment detection. Lee et al [9] achieved a 91.1% recall on a rooftop identification task using overhead satellite imagery. This approach uses a feature pyramid network to identify roof segments and nearby structures and output a roof orientation matrix, roof mask, and vegetation and background masks. They then apply a heuristic-based approach for computing solar potential given solar irradiation. Combined approaches involving deep models for rooftop extraction and heuristic-based approaches for evaluating solar potential demonstrate the most promise and applicability for commercial use, as solar potential is often limited by factors not observable from aerial imagery such as cost and energy infrastructure.

Figure 1. Overhead aerial imagery of a commercial building in New York City, with some examples of RTUs outlined in blue. Other rooftop features such as greenery, ducting, seating, and tiling also clutter the scene.

CNN/R-CNN approaches have been used in similar domains: for example, Castello et al [2] trained a U-Net on remote sensing imagery to identify solar panels on rooftops. Yao et al [21] used Faster R-CNN to identify chimneys and condensing towers on a self-collected dataset. Fernandes et al [5] applied a Detectron2 Mask-RCNN based approach on a dataset of 56 images with image augmentation to detect RTUs. This last approach reported a train mean average precision (MAP) of 87.5% and a test MAP of 91.1%. Notably, Fernandes et al collected their own rooftop imagery via drone photography. The resulting training data were imaged from a shallower angle and expose different features of RTUs than would otherwise be visible from aerial imagery.

3. Methods

3.1. Faster R-CNN

Faster R-CNN [17] is a state-of-the-art object detection model. As a two-stage object detector, Faster R-CNN builds on the Fast R-CNN [7] architecture by using a region proposal network (RPN). Given an image as input, the RPN module outputs a set of object proposals represented by rectangles. Each proposal is associated with a score. A region of interest (RoI) pooling layer crops and resizes feature maps according to region proposals from the RPN. The new feature maps from the RPN can then be used for classification and bounding box regression.

Like the original implementation of Faster R-CNN, our approach combines two loss functions for bounding box regression. Let a ground truth object be defined by the bounding box rectangle (x, y, w, h) and a region of interest be defined by the rectangle $(x_{roi}, y_{roi}, w_{roi}, h_{roi})$. The target vector y_r represents a distance encoding between the ground truth and RoI:

$$y_r = \begin{bmatrix} \frac{x - x_{roi}}{w_{roi}} \\ \frac{y - y_{roi}}{h_{roi}} \\ log(\frac{w}{w_{roi}}) \\ log(\frac{h}{h_{roi}}) \end{bmatrix}$$

The classifier loss L_c represents binary cross-entropy loss, and the regression loss L_r represents mean squared error (MSE):

$$L_c = -\frac{1}{n}\sum_{i=1}^{n}(y_{c,i}\log\hat{y}_{c,i} + (1 - y_{c,i})\log(1 - \hat{y}_{c,i}))$$

$$L_r = \frac{1}{2n}\sum_{i=1}^{n} y_{c,i}||y_{r,i} - \hat{y}_{r,i}||^2$$

$$L = L_c + L_r$$

Our specific implementation of Faster R-CNN uses a pretrained ResNet-50-FPN backbone with a Fast R-CNN detector network fine-tuned for the rooftop unit object detection task. The pretrained Faster R-CNN implementation available in the `torchvision` package handled 91 output classes (including the background). For our object detection task, we are only interested in 2 classes: RTU and background. In particular, we are only fine-tuning the softmax classifier and bounding box regressor stage of the Fast R-CNN head. The backbone was pretrained on the COCO 2017 dataset [13].

3.2. Other Approaches

As we will discuss in 5.2, we investigated other object detection algorithms for this task. These include:

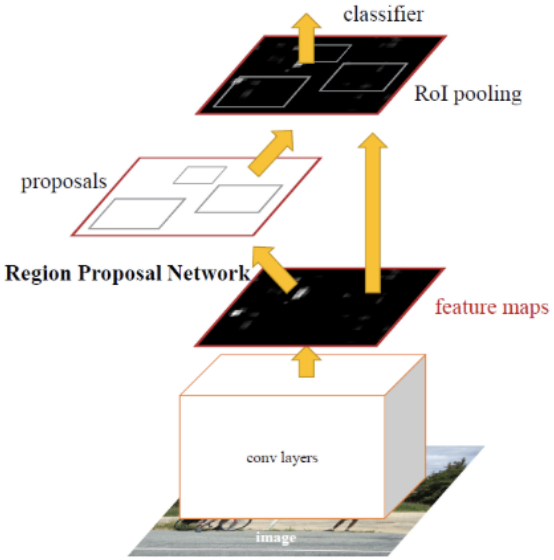

Figure 2. Faster R-CNN architecture. Figure from original paper[17].

OWL-ViT. OWL-ViT [15] is an open-vocabulary object detection model based on the Vision Transformer [4] architecture. This approach involves separately training a vision encoder and a text encoder. Contrastive loss is evaluated over this combined text and image embedding. At inference time, query strings are embedded with the text encoder. OWL-ViT appends small object classification/localization heads to the output of the image encoder; as a result the model can perform open-vocabulary object detection.

RetinaNet. RetinaNet [12] is a one-stage object detector. Unlike Faster R-CNN, one-stage object detectors such as RetinaNet do not use a region proposal network to generate regions of interest. This tends to come with a tradeoff in accuracy because of difficulty localizing small or overlapping objects. RetinaNet proposes an improvement to the single-stage object detector approach by defining a focal loss term to remediate class imbalance between foreground and background. This addresses flaws of one-stage object detectors in localizing small objects, which are otherwise prone to foreground-background class imbalance.

4. Dataset and Features

No existing rooftop imagery dataset was available with labelled RTU equipment. The input dataset for our model was manually assembled and uses the Aerial Imagery for Roof Segmentation (AIRS) dataset for source aerial imagery. The Aerial Imagery for Roof Segmentation (AIRS) [3] dataset provides 7.5cm-resolution aerial imagery of over 220,000 buildings. The dataset is designed specifically for roof semantic segmentation problems. Buildings in the AIRS dataset can contain 0 or more instances of a vari-

ety of rooftop HVAC equipment, including but not limited to air conditioners, VAV systems, evaporators, blower fans, chillers, condensers, VRFs/VRVs, and heat pumps. It is not always straightforward even for a human observer to differentiate between the systems, but the broad category of "rooftop unit" can generally be easily discerned. Images are specified in `tif` format.

We manually extract individual buildings from AIRS imagery in New York City and Taiwan. Then, we manually labeled bounding boxes for RTUs atop each building. This dataset contains 98 high-resolution building rooftop images and bounding boxes for RTUs located in these images, corresponding to 98 unique buildings. A given building rooftop can contain more than one RTU. Across all 98 images in our manually-collected dataset there are 635 unique instances of RTUs. The train, validation, and test sets include 78, 10, and 10 images respectively. Bounding boxes were confined to a single output class, resulting in only two output classes: background and RTU. **Figure** 3 provides an example of RTUs in bounding boxes on a single building rooftop.

Manual annotations for bounding boxes were stored in the COCO JSON format. A customer dataloader was implemented in `Torch`. Our bounding boxes in COCO JSON format was transformed from a `xmin, ymin, width, height` format to the `xmin, ymin, xmax, ymax` format expected by `Torch`.

Images in the dataset were then preprocessed to 640px x 640px. Larger images were cropped; smaller images were filled with 0 value pixels. Input images were normalized to pixel ranges between $[0, 1]$. We normalize the images using the sample mean and standard deviation. Each image has 3 (R, G, B) channels, resulting in a $(3, 640, 640)$ tensor for each image example. This dataset is available at `https://github.com/NWChen/rtus`.

Finally, we experiment with additional data augmentation. These transforms include horizontal and vertical flipping with $p = 0.5$, 90- and 270-degree rotations with probability $p = 0.5$, and saturation jitter of 0.3. All transformations were performed using the `torchvision` library.

As seen in **Figure** 4, RTUs can come in many different shapes and sizes. They can also be occluded by other rooftop features, vegetation, and dirt or debris.

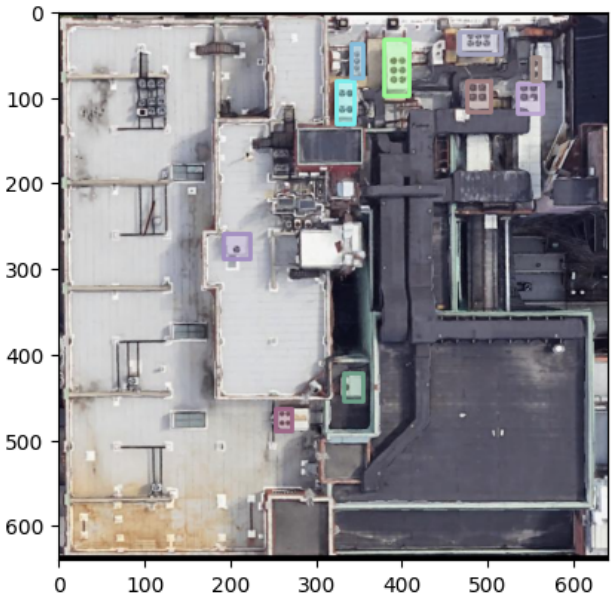

Figure 3. An example of a rooftop consisting of multiple RTUs. Colored bounding boxes surround each RTU instance.

Figure 4. An example of a rooftop containing many RTUs (outlined in blue) of varying type and dimension.

5. Experiments

Experiments were tested in a Google Cloud environment with a Tesla T4 GPU. `PyTorch` implementations were used for pretrained models.

We use a stochastic gradient descent (SGD) optimizer and a learning rate scheduler that decays learning rate by a factor of 0.1 every 3 epochs to improve convergence speed and model performance. We selected a mini-batch size of 4 images given the small input dataset. We use cross-validation with 5 folds. Models were trained for 20 epochs.

For hyperparameter optimization, several values of learning rate $(0.0005 \leq n \leq 0.005)$, momentum $(0.1 \leq n \leq 1.0)$, and weight decay $0.0001 \leq n \leq 0.0008$ were used. **Table** 1 details results with each of these hyperparameters.

Figure 5. A rooftop with 3 RTUs. Ground truth bounding boxes are in blue.

Figure 6. Predictions (in red), including an incorrectly classified false positive in the center of the image.

5.1. Metrics

Common classification or object detection metrics include precision, recall, and F1 score, which are defined as follows:

$$\text{Precision} = \frac{\text{True positive}}{\text{True positive} + \text{False positive}}$$

$$\text{Recall} = \frac{\text{True positive}}{\text{True positive} + \text{False negative}}$$

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

True positives were identified using the intersection-over-union (IoU) metric with a threshold of 0.5. For two bounding boxes A and B, this metric is defined as

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}$$

such that for an IoU > 0.5 between a ground truth bounding box and a predicted bounding box is considered a true positive.

5.2. Results

With precision of 0.5805 and recall of 0.9167, test set F1 score of 0.7111 was observed in the final model. Overfitting is unlikely given the similarity between training and test set performance. With high learning rates (e.g. above 0.001, loss explosion was observed. In these cases, losses rapidly jumped to NaN. **Figure** 7 shows precision, recall, and loss

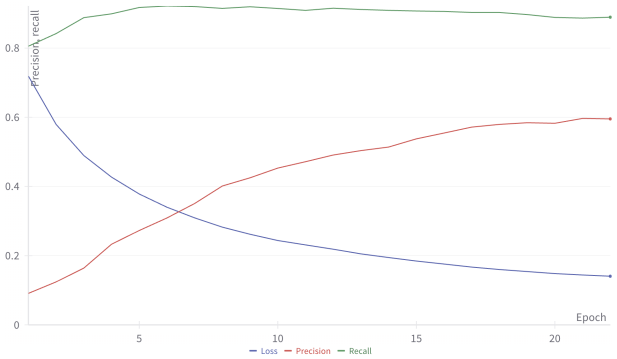

Figure 7. Precision, recall, and loss over 20 training epochs of the final model.

curves as a function of training eopchs in the final Faster R-CNN-based model.

Qualitatively, false positives often occurred in cases where other rooftop features exhibited similar contours to RTUs. For example, **Figure** 5 shows a rooftop image with bounding boxes in blue. In **Figure** 6 we see that the model misidentified an additional feature near the bottom-center of the image as an RTU. This is presumably due to a large circular feature being bound by a rectangular feature, which is a common pattern for RTUs that have intake/exhaust fans facing the sky.

We also observe examples of false positives where the model struggles to differentiate between a single RTU and separate but adjacent RTUs. For example, **Figure** 9 shows a case where the model detected multiple separate RTUs (in

Method	Hyperparameters			Precision	Recall	F1 score
	Learning rate	Momentum	Weight decay			
Faster R-CNN	0.001	0.9	0.0005	0.6408	0.9167	0.7543
Faster R-CNN	0.001	0.4682	0.0005	0.4083	0.67	0.5074
Faster R-CNN	0.00275	0.55	0.0005	0.5156	0.8944	0.6541
Faster R-CNN	0.00275	0.55	0.0005	0.5439	0.9028	0.6788
Faster R-CNN	0.00275	0.1	0.0008	0.4552	0.9167	0.6083
Faster R-CNN	0.005	0.55	0.0001	0.6878	0.9028	0.7808
Faster R-CNN	0.005	0.55	0.0005	0.6394	0.9236	0.7557
Faster R-CNN	0.005	0.55	0.0008	0.6947	0.9167	0.7904
Faster R-CNN	0.005	0.1	0.0001	0.6195	0.8819	0.7278
Faster R-CNN	0.005	0.1	0.0005	0.5936	0.9028	0.7163
Faster R-CNN	0.005	0.1	0.0008	0.5397	0.8958	0.6736

Table 1. Precision, recall, and F1 scores on validation data for various hyperparameter settings with a Faster R-CNN-based approach.

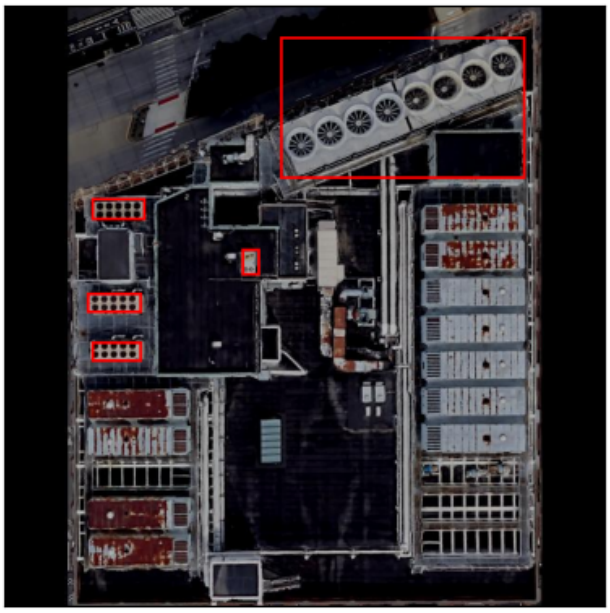

Figure 8. Predictions (in red) of RTUs, including a large unit at a non-right angle.

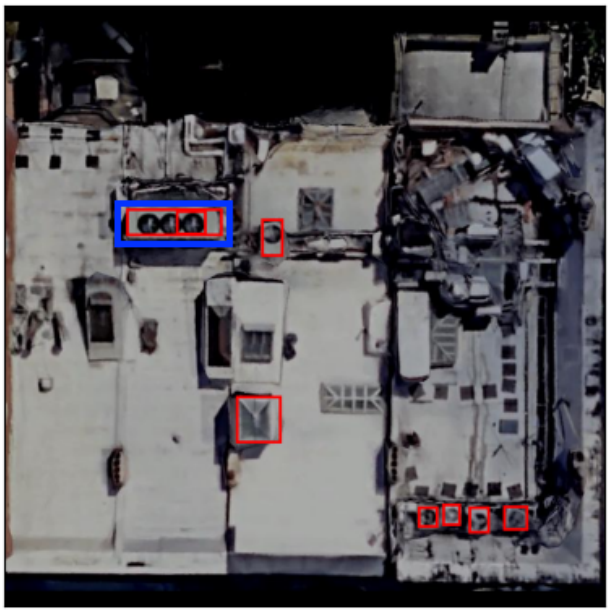

Figure 9. Predictions (in red) of separate RTUs. The correct detection is in blue, for a large, single RTU.

red) despite only a single ground truth RTU (in blue) being present. This type of false positive error can occur even with human classifiers due to the proximity of RTUs to one another on rooftops.

The Faster R-CNN approach was robust to rotation and flip of the same image, and qualitatively demonstrated the ability to detect RTUs in a various orientations, including those at non-right angles. An example of this is visible in **Figure** 8.

5.3. Other Approaches

We experimented with other algorithms but did not investigate quantitative results due to poor performance, including:

OWL-ViT. We attempted to use the OWL-ViT [15] open-vocabulary object detection network, which uses CLIP as a multi-modal backbone, to identify rooftop equipment using text queries. This approach involves zero-shot text-conditioned object detection. OWL-ViT was unable to identify any rooftop equipment from raw imagery, and produced no bounding boxes in 3 random samples. This may be because RTUs are a rare object class easily confused for other features in aerial imagery, and may therefore not be suitable for the one/few-shot use case.

RetinaNet. We attempted to use the RetinaNet [12] single-stage object detector to identify RTUs using the input dataset. We achieved best recall of 0.52 and precision of 0.1195 with this approach for an F1-score of 0.1943.

We believe the poor F1-score performance of this approach may be attributable to incorrectly defined weights in the pretrained backbone of the implementation, and more work is needed to investigate whether a RetinaNet-based approach can perform as well as our demonstrated Faster R-CNN approach.

6. Conclusion and Future Work

We demonstrated that, despite a very small input dataset of fewer than 100 images with 635 examples of the target class, a transfer learning approach based on Faster R-CNN can effectively detect RTUs across a wide distribution of commercial rooftop aerial imagery. We achieved an F1 score of 0.7111 for object detection of RTUs.

While we demonstrated the effectiveness of deep convolutional networks for object detection of RTUs given a single target class generically representing all RTUs, future work could evaluate the performance of pretrained object detectors on multiple subtypes of RTU. These include heat pumps, condensers, chillers, blowers, evaporators, VAV systems, and more. Automatic detection of these subtypes could provide valuable insight for evaluating building/equipment retrofit suitability.

This work also evaluates object detection given input data that has already isolated rooftops of individual buildings. Additional future work could evaluate the performance of an end-to-end pipeline that, given aerial imagery, could both isolate individual buildings and identify RTUs on individual building rooftops.

7. Contributions and Acknowledgements

The author contributed all work represented in this paper, but the CS231n teaching staff made this work possible.

PyTorch [16] was used for all data loading, model training, and evaluations. The Roboflow and Weights and Biases platforms were used for image annotation and training observability, respectively. The author thanks the creators of these tools.

References

[1] L. Abraham and M. Sasikumar. Unsupervised building extraction from high resolution satellite images irrespective of rooftop structures. *International Journal of Image Processing (IJIP)*, 6(4):219–232, 2012.

[2] R. Castello, S. Roquette, M. Esguerra, A. Guerra, and J.-L. Scartezzini. Deep learning in the built environment: Automatic detection of rooftop solar panels using convolutional neural networks. In *Journal of Physics: Conference Series*, volume 1343, page 012034. IOP Publishing, 2019.

[3] Q. Chen, L. Wang, Y. Wu, G. Wu, Z. Guo, and S. L. Waslander. Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings. *ISPRS journal of photogrammetry and remote sensing*, 147:42–55, 2019.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[5] S. Fernandes, R. Najibi, A. Prakash, R. Singh, M. Zafiris, and J. Granderson. Thermal anomaly and rooftop unit (rtu) detection in buildings through machine learning. In *Remote Sensing Technologies and Applications in Urban Environments VII*, volume 12269, pages 51–57. SPIE, 2022.

[6] K. Gao, M. Chen, S. Narges Fatholahi, H. He, H. Xu, J. Marcato Junior, W. Nunes Gonçalves, M. A. Chapman, and J. Li. A region-based deep learning approach to instance segmentation of aerial orthoimagery for building rooftop extraction. *Geomatica*, 75(1):148–164, 2022.

[7] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[8] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[9] S. Lee, S. Iyengar, M. Feng, P. Shenoy, and S. Maji. Deeproof: A data-driven approach for solar potential estimation using rooftop imagery. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2105–2113, 2019.

[10] Q. Li, Y. Feng, Y. Leng, and D. Chen. Solarfinder: Automatic detection of solar photovoltaic arrays. In *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 193–204. IEEE, 2020.

[11] X. Li, X. Yao, and Y. Fang. Building-a-nets: Robust building extraction from high-resolution remote sensing images with adversarial networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(10):3680–3687, 2018.

[12] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[14] J. M. Malof, R. Hou, L. M. Collins, K. Bradbury, and R. Newell. Automatic solar photovoltaic panel detection in satellite imagery. In *2015 International Conference on Renewable Energy Research and Applications (ICRERA)*, pages 1428–1431. IEEE, 2015.

[15] M. Minderer, A. Gritsenko, and N. Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36, 2024.

[16] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

[17] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[18] T. Sun, M. Shan, X. Rong, and X. Yang. Estimating the spatial distribution of solar photovoltaic power generation potential on different types of rural rooftops using a deep learning network applied to satellite images. *Applied Energy*, 315:119025, 2022.

[19] L. Wang, S. Fang, X. Meng, and R. Li. Building extraction with vision transformer. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2022.

[20] B. Wu, S. Wu, Y. Li, J. Wu, Y. Huang, Z. Chen, and B. Yu. Automatic building rooftop extraction using a digital surface model derived from aerial stereo images. *Journal of Spatial Science*, 67(1):21–40, 2022.

[21] Y. Yao, Z. Jiang, H. Zhang, B. Cai, G. Meng, and D. Zuo. Chimney and condensing tower detection based on faster r-cnn in high resolution remote sensing images. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3329–3332. IEEE, 2017.