2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Detection by Classification of Buildings in Multispectral Satellite Imagery

Tomohiro Ishii[1,2], Edgar Simo-Serra[1], Satoshi Iizuka[1], Yoshihiko Mochizuki[1],
Akihiro Sugimoto[3], Hiroshi Ishikawa[1], and Ryosuke Nakamura[2],
[1]Waseda University, [2]National Institute of Advanced Industrial Science and Technology, [3]National Institute of Informatics
Email: tomohiro.ishii@asagi.waseda.jp, {esimo, iizuka, motchy}@aoni.waseda.jp, sugimoto@nii.ac.jp, hfs@waseda.jp, r.nakamura@aist.go.jp

***Abstract*—We present an approach for the detection of buildings in multispectral satellite images. Unlike 3-channel RGB images, satellite imagery contains additional channels corresponding to different wavelengths. Approaches that do not use all channels are unable to fully exploit these images for optimal performance. Furthermore, care must be taken due to the large bias in classes, *e.g.*, most of the Earth is covered in water and thus it will be dominant in the images. Our approach consists of training a Convolutional Neural Network (CNN) from scratch to classify multispectral image patches taken by satellites as whether or not they belong to a class of buildings. We then adapt the classification network to detection by converting the fully-connected layers of the network to convolutional layers, which allows the network to process images of any resolution. The dataset bias is compensated by subsampling negatives and tuning the detection threshold for optimal performance. We have constructed a new dataset using images from the Landsat 8 satellite for detecting solar power plants and show our approach is able to significantly outperform the state-of-the-art. Furthermore, we provide an in-depth evaluation of the seven different spectral bands provided by the satellite images and show it is critical to combine them to obtain good results.**

I. Introduction

Everyday, dozens of satellites meticulously take large amounts of images of the Earth's surface. These images contain a large amount of information and have many applications such as land-use analysis, map making, and contingency planning against disaster. The vast amount of data that is provided daily exceeds the possibility of manual analysis: it is only possible to process this data by semi-automated and fully-automated tools. In this work we present an approach for training and employing a Convolutional Neural Network (CNN) for object detection in satellite imagery.

Our approach consists of training a CNN using image patches for classification of whether or not the image patch contains part of the building type we want to localize. Afterwards we convert the fully-connected layers to convolutional layers to be able to process images of arbitrary resolutions [1]. The resulting network can be used to detect buildings directly in satellite images in near real-time.

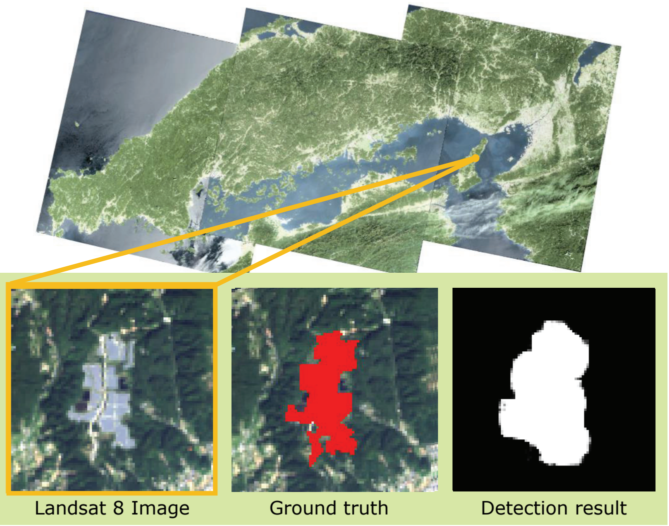


Fig. 1: Example result of the proposed method. We are able to train our CNN for classification and then apply it to detection by adopting the fully-convolutional layers. While these images are multispectral, we only show the RGB channels.

One of the more significant properties of satellite imagery is that, in general, satellite images contain more spectral information than standard RGB images. In particular, spectral bands that are not visible to the human eye such as infrared bands are also used. All these bands are represented as different channels of the input image. Thus, instead of 3-channel RGB images, it is common to work with images with more channels corresponding to different wavelengths. Objects that are hard to distinguish with the standard RGB channels might be easier to discern using the additional wavelengths. In particular, we consider 7-channel images as input.

In this work, we focus on the detection of solar power plants, which play a fundamental role in energy planning as a renewable energy. Photovoltaic installations are expected to surpass 310 gigawatts worldwide by the end of 2016, while only 40 gigawatts were installed at the end of 2010 [2]. Detecting solar power plants enables us to estimate energy productions and use this to assist landscape planning. Furthermore, as solar power plants require direct light, they are generally fully visible from satellite imaging, which allows for the robust evaluation of our approach.

We build a dataset for the detection of solar power plants using the Landsat 8, one of the latest Earth observation satellites, which observes the whole surface of the Earth with a 16-day repeat cycle [3]. Each image has 7 channels corresponding to different wavelengths, with half corresponding to the non-

978-1-5090-4847-2/16/$31.00 ©2016 IEEE

visible infra-red spectrum. The resolution is of roughly 30 meters per pixel. We use a database of existing solar power plants to annotate the images.

In summary, we present a new dataset for the detection of power plants in multispectral satellite imagery, and an approach to learn models for detection by training for classification that is able to significantly outperform the state-of-the-art on the task. We additionally provide extensive results on the contribution of each of the spectrum for the task.

II. Related Work

In the last few years, the field of image classification has come to be dominated by CNN [4], [5], which are significantly outperforming traditional methods [6]. While the focus of the work was classifying images, it was later extended to detection by combining region proposals with classification networks [7]. CNN have also been used for small image patches [8], and fully convolutional networks that are able to process images of any resolution have been recently been proposed [1], [9], [10]. In one approach, a classification network has been adapted to perform semantic segmentation [1] by converting the fully-connected layers to convolutional layers. We use this adaptation in our approach. However, unlike Long *et al.* [1], we do not use off-the-shelf classification networks since we deal with multispectral image inputs, and we do not finetune in the segmentation stage due to the bias in classes in our dataset.

CNNs have also been used in remote sensing. Minh and Hinton [11] created synthetic data from vector road maps to train a CNN for road detection. Our work is different in that we exploit multispectral images and that they rely on a large pixel-accurate dataset, which is not available in our case. Castelluccio *et al.* [12] explored the use of CNNs for multi-class image classification in aerial images using existing CNN models, *e.g.*, CaffeNet [13]. Penatti *et al.* [14] evaluated the performance of feature descriptors based on existing CNNs in aerial and remote sensing image classification. They showed that CNNs obtain the best performance for both aerial and remote sensing images. However, these methods rely on existing networks that are designed to process 3-channel images, unlike the images with multiple spectral bands we consider. By training an architecture from scratch, we are able to exploit all input channels which is important for classification as shown in Fig. 2. Additionally, we adapt our network trained for classification to detection and are able to efficiently process high resolution images.

III. MegaSolar Dataset

We build a dataset by using the publicly available multi-spectral images taken by the Landsat 8 satellite and annotating them with the location of *MegaSolar* solar power plants, taken from a public database. As there is a time lapse between the images and the power plant database, in addition to the fact that the database is not exhaustive, we take care in creating a set of positive and negative examples that can be used for training. This is critical for performance, as there is a large bias between positive and negative examples in the dataset.

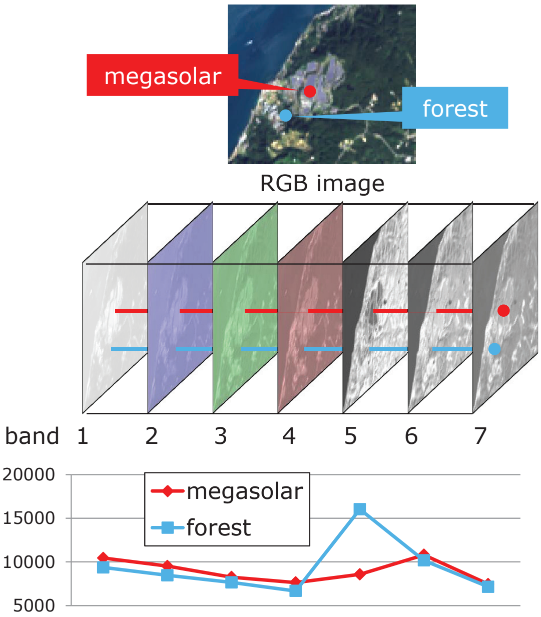


Fig. 2: Comparison of multispectral signals in an example Landsat 8 satellite image. Comparing two pixels corresponding to a *megasolar* solar power plant and forest, we can see that both pixels have nearly the same value in most channels. However, channel 5, corresponding to the 0.85-0.88μm wavelength, is significantly different. Thus, by exploiting all the channels, we are able to distinguish areas that would not be distinguishable by RGB information only. Bands 2, 3, and 4 are colored for ease of understanding.

TABLE I: Wavelengths observed by the Landsat 8 satellite. OLI corresponds to visible and near-infrared light, while TIRS corresponds to thermal infrared light. Bands 2, 3, and 4 correspond to the standard B, G, and R channels, respectively.

Sensor	Band	Wavelength [μm]	Resolution [m]
OLI	1	0.43–0.45	30
	2 (B)	0.45–0.51	30
	3 (G)	0.53–0.59	30
	4 (R)	0.64–0.67	30
	5	0.85–0.88	30
	6	1.57–1.65	30
	7	2.11–2.29	30
	8	0.53–0.68	15
	9	1.36–1.38	30
TIRS	10	10.60–11.19	100
	11	11.50–12.51	100

A. Landsat 8 Satellite Imagery

The Landsat 8 Satellite is a new Earth observation satellite that has been operating since 2013. It observes the whole surface of the Earth on a 16-day repeat cycle using 11 different bands with different wavelengths and spatial resolutions. An overview of satellite's imaging equipment can be seen in Table I. Of these bands, we use the first seven as they share a common resolution and have mostly no overlap. For our dataset, we use 20 satellite images of Japan taken in 2015. Each image captures roughly an area of $170\text{km} \times 183\text{km}$. Example images

TABLE II: Proposed model architecture. The model consists of three convolutions with Rectified Linear Unit (ReLU) transfer functions, followed by a single fully-connected layer. Note that as we use the negative log-likelihood loss, softmax transfer function is applied on the final layer.

layer	kernel size	output size	transfer function
input	-	$7 \times 16 \times 16$	-
convolution	3×3	$32 \times 14 \times 14$	ReLU
convolution	3×3	$32 \times 12 \times 12$	ReLU
convolution	3×3	$32 \times 10 \times 10$	ReLU
fully connected	-	2	softmax

are shown in Fig. 1.

In order to train for classification, we divide each of the image into cells of 16×16 pixels and treat each cell as an image patch which is used for the input. Note that, despite being seemingly small, each cell covers an area of about 480m^2. The cells will be then annotated, as we explain next.

B. MegaSolar Power Plants

We use the database maintained by *Electrical Japan*[1] to obtain the location of solar power plants. Due to the coarse resolution of the satellite imagery, and the fact that smaller plants are too numerous for manual annotation, we use only the power plants with an output greater than 5MW for positive samples. Each such plant is manually annotated in the satellite image with a polygon. Then, cells with more than 20% of the pixels covered by the polygon are considered positives, while those without a single pixel belonging to a power plant are considered negatives, except those that fall within 480m (16 pixel) radius of plants with under 5MW output, since the smaller power plants tend to make the negative samples ambiguous. Unlike the annotation, this removal can be done automatically. This results in a total of 426 positive cells and 3,210,627 negative cells. Note that there are 7,537 times more negative examples than positive examples. We will take this into account when training our model.

IV. Method

We train a CNN from scratch for classification. This allows the model to process multispectral input images. We then adapt it for detection by converting the fully-connected layers to convolutional layers. This allows near real-time processing of high resolution satellite images.

A. Model Architecture

Our model consists of three convolutional layers with a fully-connected layer as shown in Table II. We use Rectified Linear Unit (ReLU) transfer functions after each of the convolutional layers. The convolutional layers all use 3×3 kernels and no padding. Unlike most standard models, we do not use any pooling; and all convolutions have a stride of one. This is due to the small size of the input images. Additionally, note that the input has 7 channels, corresponding to the first seven bands from Table I. By exploiting more information in the input, it is possible to keep the number of layers low, which allows processing high resolution images in near real-time.

B. Training for Classification

We train our model for classification by our dataset of 16×16 pixel image patches with their corresponding annotations using a negative log-likelihood loss. In order to efficiently learn from scratch, we use Batch Normalization layers [15] after each convolution and before the corresponding ReLU transfer function. Note that these layers are only necessary for training. When evaluating, they can be reduced into a fixed linear transformation, which can be "folded" into the previous convolutional layer, thus adding no additional overhead. We also add a DropOut layer [16] that sets the output pixels to 0 with 50% possibility after the third convolutional layer and before the fully-connected layer. This reduces the overfitting of the model to the training data.

Due to the large bias between the positives and the negatives, we both augment the number of the positives and decrease the number of the negatives for training. As the grid cells are non-overlapping, we consider all possible grids when computing the positives, *i.e.*, we consider all 16×16 image patches with at least 20% of the pixels belonging to a solar power plant as positives. In contrast, the negatives are computed using a single grid. This allows roughly 16-fold increase in the number of positives. For the negatives, we perform random subsampling to reduce them to roughly 14% of the original amount. Both these modifications allow reducing the gap between the positives and the negatives such that there are only 67 times more negatives than positives, *i.e.*, over a 100 times reduction in the gap between negatives and positives. Note that this is only done for the training data: we use all the cells for both the validation and testing data. We further augment the data by randomly flipping all images both horizontally and vertically during training.

Despite augmenting the positive samples and decreasing the negative samples of the training set, there is still a large remaining bias between positives and negatives. Once a model is trained, in order to further reduce the effect of this bias, we use the validation set to determine the optimal classification threshold. As we will show, using the default threshold of 0.5 leads to high recall, but low precision and intersection-over-union values. Tuning the threshold is critical to increase both the precision and the intersection over union.

In summary, by reducing the dataset bias, accelerating the training, and reducing the overfitting, we are able to train models quickly and efficiently for high performance classification of power plants from image patches. Once the model is trained, by tuning the threshold, we are able to further overcome the dataset bias and significantly increase performance.

C. Adaptation to Detection

Inspired by Long *et al.* [1], we convert our trained classification network to a fully-convolutional network by reinterpreting

[1] http://agora.ex.nii.ac.jp/earthquake/201103-eastjapan/energy/electrical-japan/, in Japanese only

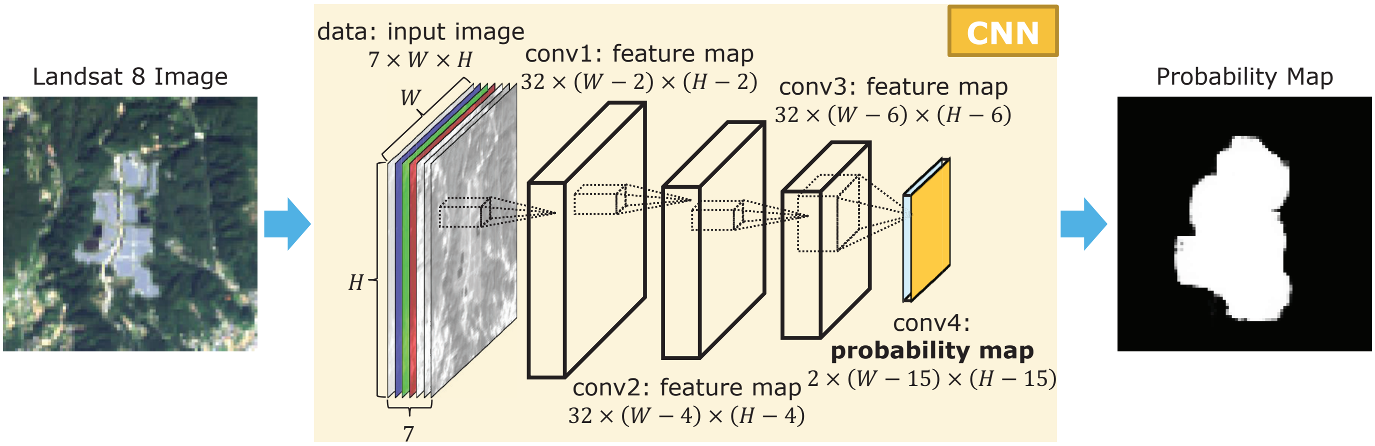


Fig. 3: The detector derived from our proposed model. The model is originally trained with a fully-connected layer for classification. The fully-connected layer can then be reinterpreted as a convolutional layer, which allows converting our model to a fully-convolutional one that can process images of any size. The resulting model outputs a probability map that is the size of the input image minus 15 pixels, as we do not use padding for the convolutional layers.

TABLE III: Details of the train, validation, and test split of the MegaSolar dataset. For each split, we note the number of positive samples P and the number of negative samples N.

Dataset		# of cells
original training	P^*_{train}	300
	N^*_{train}	2,247,428
augmented training	P_{train}	4,851
	N_{train}	320,000
validation	P_{val}	21
	N_{val}	160,533
testing	P_{test}	105
	N_{test}	802,666

the fully-connected layer as a convolutional layer. In the case of our proposed model, this simply consists of converting the last layer to a convolution with a 10 × 10 pixel kernel. The new model, in the case of a 7 × 16 × 16 input image, will output a 2 × 1 × 1 image instead of a 2-dimensional vector. Extrapolating this to $7 \times W \times H$ input images, the output will be a $2 \times (W - 15) \times (H - 15)$ image that can be interpreted as a probability map, due to the last softmax transfer function, as shown in Fig. 3. This map can be used directly for detections and the entire approach can be computed in a single forward pass on an image, in contrast to approaches that require bounding box proposals such as R-CNN [7].

V. Results

For evaluation, we split the MegaSolar dataset into three sets: train, validation, and test, so that each power plant is fully contained in either the train and validation sets, or the test set. All sets consist of a number of 16 × 16 pixel multispectral images with their associated label, *i.e.*, positive or negative. We use a 70:5:25 ratio for training, validation and testing respectively. The training set is also augmented in order to reduce the gap between the positives and the negatives during training. All methods are trained on the augmented training set, validated on the validation set, and tested on the test set. An overview of the different sets is shown in Table III.

A. Comparison with the State-of-the-Art

We compare against the state-of-the-art approach for recognition in aerial images by Penatti *et al.* [14]. It consists of using the pre-trained CaffeNet [17] and replacing the last two layers with a single layer for two-way classification. Instead of using the standard RGB channels for input, they use bands 5, 6, and 7. Finally, the whole network is fine-tuned for the task, which in this case is classification of solar power plants. As done in [14], the input images are enlarged to be able to be inputted to the network by the bicubic interpolation.

We also provide a comparison with an non-linear SVM using RBF kernels as a baseline. The SVM is trained on vectorized images using the first seven channels like our approach. The SVM hyperparameters are determined by using grid search on the validation set.

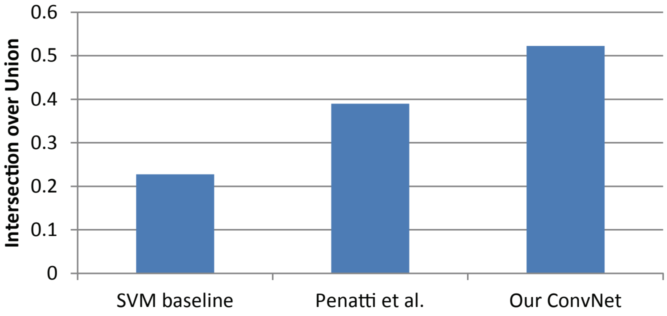


Fig. 4: Comparison against the state-of-the-art for the classification of solar power plants. We compare our best performing method against Penatti *et al.* [14] and a RBF-kernel SVM baseline. Our method significantly outperforms the other methods.

The comparison against the state-of-the-art can be seen in Fig. 4. We evaluate using the intersection over union metric on

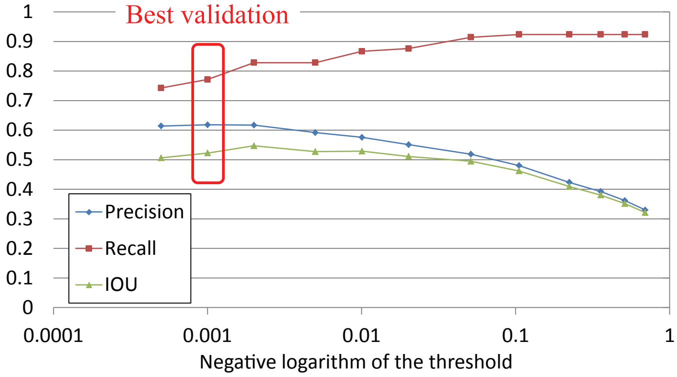


Fig. 5: Tuning the classification threshold hyperparameter. We show the results of changing the threshold on the train set. Note that we chose the best performing value on the validation set, which is close to the best on the test set, and highlighted in red. Intersection over union increases by nearly 20% due to the tuning of the threshold.

the test set for the SVM baseline, the state-of-the-art Penatti *et al.* [14], and our best performing model. For a fair comparison, we tune the classification threshold for all approaches on the validation set. We can see that the proposed significantly outperform both the baseline and the approach of Penatti *et al.* [14], despite training our model entirely from scratch.

B. Tuning the Classification Threshold

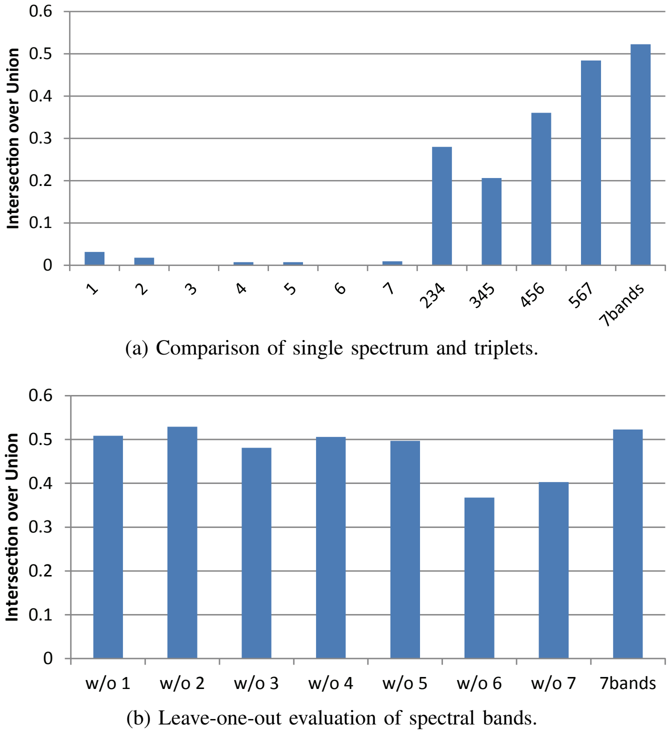
Due to the dataset bias, we tune the classification threshold of our model and show its effect in Fig. 5. This threshold plays a critical role when the dataset has a heavy positive-negative bias such as the one we use in this work, increasing performance by nearly 20% intersection over union.

C. Importance of Spectral Bands

We perform an analysis of the different bands to investigate their contributions to the final classification results. We compare using single bands, triplets of bands (including RGB), leaving a band out, and using all the bands. Results are shown in Fig. 6. We can see that using single bands leads to no performance. Using triplets of bands can already lead to high performance. In particular, it seems that the bands (5,6, and 7) used by Penatti *et al.* give the highest performance triplet. However, there is still performance to be gained by using all the bands. When leaving a band out, band 6 seems to give the largest performance drop, followed by band 7. Other bands such as 2 (corresponding to the blue channel), seem to give no change in result. In general, it seems as if the best approach is to give as much input data as possible and with training let the network figure out the best combination of channels.

D. Detection Results

We show results of adapting our classification network to detection in Fig. 7. It is able to detect all the power plants with over 5MW output successfully. Most of the smaller power plants are also detected, despite not being used in training, showing how our approach generalizes to most solar power plants just from learning from the larger solar plants.



(a) Comparison of single spectrum and triplets.

(b) Leave-one-out evaluation of spectral bands.

Fig. 6: Comparison of using different spectral bands as input of our model. We look at both the effect of using single bands and triplets of bands, and removing bands from our full model to see the contribution of the different bands. In general, upplying all the bands to the model gives the best performance.

TABLE IV: Computation time for detection on large images.

Input size (px)	CPU (s)	GPU (s)
512 × 512	1.772	0.024
1024 × 1024	7.434	0.098
2048 × 2048	29.886	0.395

E. Computation Time

We benchmark the network for classification of large images. In general, satellite images are of very high resolution. We show results in Table IV and can see that our approach can process large 2048×2048 pixel images in well under a second with a Titan X GPU.

VI. Conclusions

In this paper, we have presented an approach for learning to classify and detect objects in satellite images. Our approach is able to exploit multiple spectral bands which are common in satellite imagery, and we are able to cope with the dataset bias intrinsic to aerial detection tasks. We present a new dataset for the detection of solar power plants in multispectral images to evaluate our approach, although it can be applied to detect any type of building. Evaluation shows that our approach significantly outperforms the current state-of-the-art in aerial

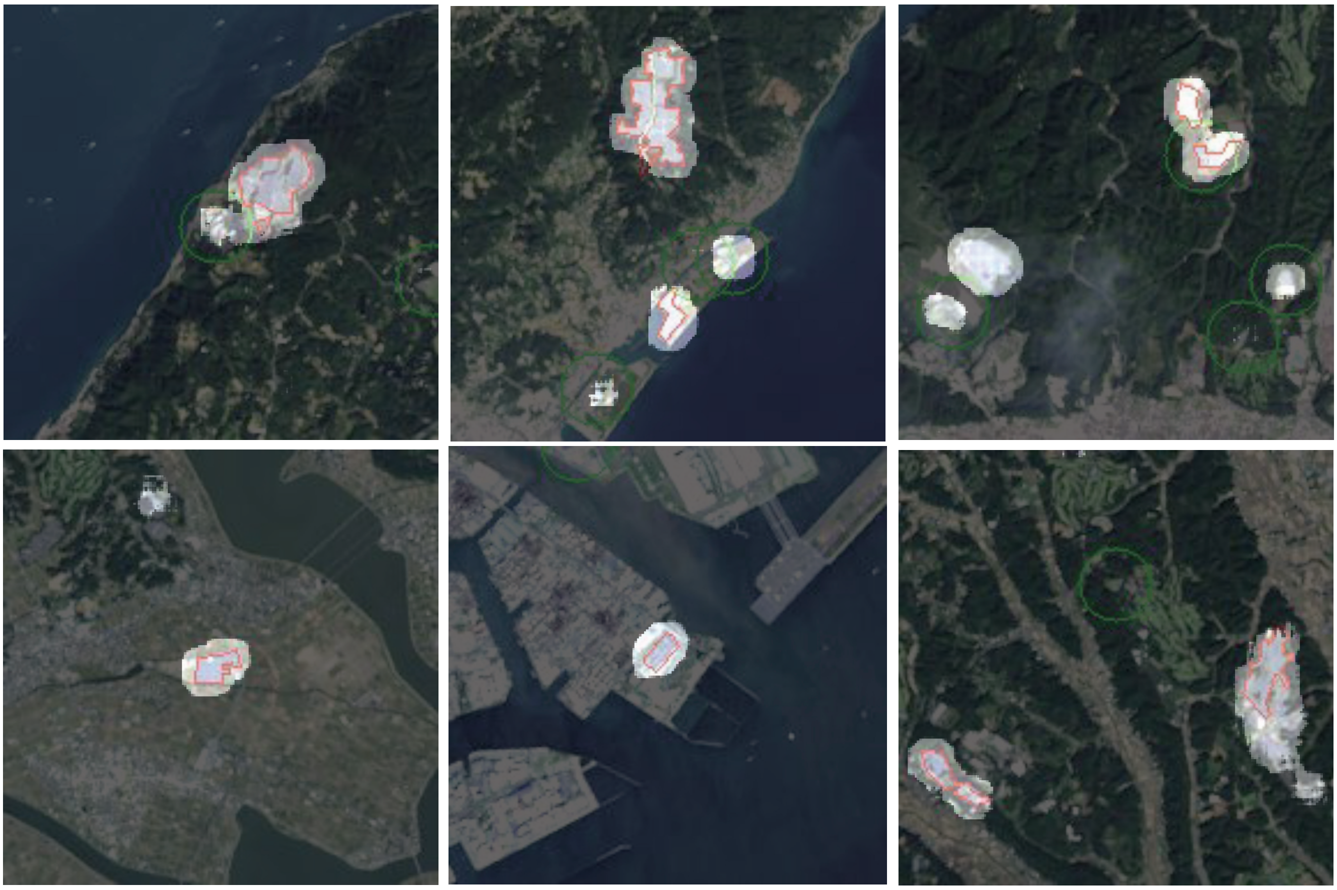


Fig. 7: Detection results for our appreach. Detection regions are highlighted. The solar power plants with more than 5MW output are enclosed by red polygons, while the smaller powerplants are denoted by a green circle with a radius of 480m.

image classification. Furthermore, we are able to adapt our classification network to detection and show that it can be used for processing large satellite images in near-realtime with accurate results.

Acknowledgment

This work was partially supported by JST CREST.

References

[1] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.

[2] IHS, Inc., "As European Solar Installations Slow, China, US and Japan Lead Global Installed PV Capacity in 2016, IHS Says," http://press.ihs.com/press-release/european-solar-installations-slow-china-us-and-japan-lead-global-installed-pv-capacity, 2016, [Online; accessed 13-April-2016].

[3] D. Roy, M. Wulder, T. Loveland, C. Woodcock, R. Allen, M. Anderson, D. Helder, J. Irons, D. Johnson, R. Kennedy *et al.*, "Landsat-8: Science and product vision for terrestrial global change research," *Remote Sensing of Environment*, vol. 145, pp. 154–172, 2014.

[4] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural networks*, vol. 1, no. 2, pp. 119–130, 1988.

[5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014.

[8] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer, "Discriminative Learning of Deep Convolutional Feature Point Descriptors," in *ICCV*, 2015.

[9] E. Simo-Serra, S. Iizuka, K. Sasaki, and H. Ishikawa, "Learning to Simplify: Fully Convolutional Networks for Rough Sketch Cleanup," *ACM Transactions on Graphics (SIGGRAPH)*, vol. 35, no. 4, 2016.

[10] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification," *ACM Transactions on Graphics (SIGGRAPH)*, vol. 35, no. 4, 2016.

[11] V. Mnih and G. E. Hinton, "Learning to label aerial images from noisy data," in *ICML*, 2012, pp. 567–574.

[12] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva, "Land use classification in remote sensing images by convolutional neural networks," *arXiv preprint arXiv:1508.00092*, 2015.

[13] Y. Jia, "Caffe: An open source convolutional architecture for fast feature embedding," http://caffe.berkeleyvision.org/, 2013.

[14] O. A. B. Penatti, K. Nogueira, and J. A. dos Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?" in *CVPR Workshops*, 2015.

[15] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[16] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *JMLR*, vol. 15, pp. 1929–1958, 2014.

[17] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *ICML*, 2014.