Working title: **How good are retrained deep neural networks at classifying images of landscapes?**

Authors: Dan, Jon, Andy, ?

Short format article (want open access and rapid publication ahead of workshops): Geosciences?

**Abstract**

There is a growing need for fully automated pixel-scale classification of large datasets consisting of color digital photographic imagery, in the analysis and interpretation of natural landscapes and geomorphic processes. The application of deep learning, specifically deep convolutional neural networks (DCNNs), to the classification of remotely sensed imagery has the potential to both outperform and simplify such tasks, compared to traditional approaches. However, the general usefulness of deep learning applied to conventional photographic imagery at the landscape scales is, at yet, largely unproven. DCNNs are computationally intensive to train and deploy, very data hungry (often requiring millions of examples to train from scratch), and require expert knowledge to design and optimize. However, these issues are mostly overcome for general applications using ‘transfer learning’, in which existing DCNN architectures, built by specialists using massive image databases and designed to recognize features/objects in images in a generic sense, and with well-studied properties, are retrained to classify specific objects and features. If DCNN-based image classification is to gain wider application and acceptance with the geoscience community, demonstrable successes need to be coupled with accessible tools with which to transfer deep neural networks to landforms and land uses in images of landscapes. In this paper, we present an efficient approach to train/apply DCNNs with/on sets of photographic images, based on a powerful and efficient graphical method, called a conditional random field (CRF), to generate DCNN training and testing data using minimal manual supervision. We apply the method to several sets of images of natural landscapes, consisting of imagery from satellites, aircraft, unmanned aerial vehicles, and fixed camera installations, synthesizing the findings to examine the general effectiveness of transfer learning to landscape scale image classification. Finally, we show how DCNN predictions on small regions of images might be used in conjunction with a CRF for highly accurate pixel level classification of images.

**1. Introduction**

The task of classifying natural objects and textures in images of landforms is increasingly widespread in a wide variety of geomorphological research (e.g. *Franklin and Wulder, 2002*; *Smith and Pain, 2009*; *Mulder et al., 2011*; *Sekovski et al., 2014*; *Ma et al., 2017*; *Cheng et al., 2017*; *O’Connor et al., 2017*), providing impetus for the development of completely automated methods to maximize speed and objectivity. The task of labeling image pixels into discrete classes is called object class segmentation or semantic segmentation, whereby an entire scene is parsed into object classes at a pixel level (e.g. *Long et al. 2015*; *Volpi and Tuia, 2017*). In this paper, we utilize two emerging themes in computer vision research, namely deep learning and structured prediction, that, when combined, are extremely effective in application to pattern recognition, and semantic segmentation of highly structured, complex objects in images of natural scenes.

Deep learning is the application of artificial neural networks with more than one hidden layer, to the task of learning and subsequently recognizing patterns in data (*LeCun et al. 2015; Goodfellow et al. 2016*). A class of deep learning algorithms called deep convolutional neural networks (DCNNs) are extremely powerful at image recognition, resulting in a massive proliferation of their use (*Szegedy et al. 2017*; *Chen et al. 2018*), across almost all scientific disciplines (e.g. *Litjens et al. 2017*; *Lui et al. 2017*; *Maggiori et al. 2017*). A major advantage to DCNNs over conventional machine learning approaches to image classification is that it does not require so-called ‘feature-engineering’ or ‘feature extraction’, which is the art of transforming the image data so that it is either more amenable to a speciﬁc machine-learning algorithm, or providing the algorithm more data by computing derivative products from the imagery, such as rasters of texture or alternative colorspaces (e.g. *Belgiu and Drăgut 2016; Cheng et al., 2017*). In deep learning, features are automatically learned from data using a general-purpose procedure. Another reputed advantage is that DCNN performance tends to keep improving with more and more data, whereas machine learning performance tends to plateau (*Dauphin et al. 2014*). For these reasons, DCNN techniques will find numerous applications where automated interpretation and quantification of natural landforms and textures are used to solve geomorphological problems.

However, many claims about the efficacy of DCNNs for image classification are largely based upon analyses of conventional photographic imagery of familiar, most anthropogenic objects, and it remains to be seen that this is still always the case for classification of images of natural textures and objects.

Images of natural landscapes – variations in lighting and weather that affect distributions of color and illumination

Changing scale and viewpoint variation

Intra-class variation (water – sun glint, turbidity, aeration)

Distinction of objects and features against background (example involving texture)

Changing seasons (deciduous vegetation)

The most popular DCNN architectures have been designed and trained on large generic image libraries such as Imagenet (*Deng et al. 2009*), mostly developed as a result of international computer vision competitions (*Russakovsky et al. 2015*) and primarily for application to close-range imagery with small spatial footprints (*Garcia-Garcia et al. 2017*), but more recently have been used for landform/landuse classification tasks in large spatial footprint imagery such as that used in satellite remote sensing (e.g. *Hu et al., 2015*; *Langkvist et al. 2016; Palafox et al., 2017*; *Lu et al., 2017; Marmanis et al. 2017*). These applications have involved design and implementation of new or modified DCNN architectures, or relatively large existing DCNN architectures. Also, these applications have largely been limited to satellite imagery only. In this contribution, one major objective is to examine the accuracy of DCNNs for oblique and nadir conventional medium-range imagery. Another objective is to evaluate the smallest, most lightweight existing DCNN models, retrained for specific landuse/landcover purposes, with no retraining from scratch and no modification or fine-tuning to the data. We utilize a concept known as ‘transfer learning’, where a model trained on one task is re-purposed on a second related task (*Goodfellow et al., 2016*). Though powerful, DCNNs are also computationally intensive to train and deploy, very data hungry (often requiring millions of examples to train from scratch), and require expert knowledge to design and optimize. Collectively, this may impede widespread adoption of these methods within the geoscience community. Fortunately, several open-source DCNN architectures have been designed for general applicability to the task of recognizing objects and features in non-specific photographic imagery. Here, we use existing pre-trained DCNN models that are designed to be transferable for generic image recognition tasks, which facilitates rapid DCNN training when developing classifiers for specific image sets. Training is rapid because only the final layers in the DCNN need to be retrained to classify a specific set of objects.

INSERT CONCISE EXPLANATION OF STRUCTURE OF THIS PAPER

**2. Methods**

**2.1. Generating DCNN training libraries**

Automated classification of pixels in digital photographic images involves predicting labels, ***y***, from observations of features, ***x***, which are derived from relative measures of color in red, green and blue spectral bands in imagery. In the geosciences, the labels of interest naturally depend on the application but may be almost any type of surface landcover (such as specific sediment, landforms, geological features, vegetation type and coverage, water bodies, etc) or description of landuse (rangeland, cultivated land, urbanized land, etc). The relationships between ***x*** and ***y*** are complex and non-unique, because the labels we assign depend nonlinearly on observed features, as well as on each other. For example, neighboring regions in an image tend to have similar labels (i.e. they are spatially autocorrelated). Depending on the location and orientation of the camera relative to the scene, labels may be preferentially located. Some pairs of labels (e.g. ocean and beach sand) are more likely to be proximal than others (e.g. ocean and arable land).

A natural way to represent the manner in which labels depend on each other is provided by graphical models (REFS).

INSERT CONCISE EXPLANATION OF GRAPHICAL MODELS HERE

Much work in learning with graphical models has focused on generative models that explicitly attempt to model a joint probability distribution *P*(***x***,***y***) over inputs, ***x***, and outputs, ***y***. However, this approach has important limitations for image classification where, not only is the dimensionality of ***x*** potentially very large, but also the features may have complex dependencies, such as the dependencies or correlations between multiple metrics derived from images. Therefore, modeling the dependencies among ***x*** is difficult and leads to unmanageable models, but ignoring them can lead to poor classifications.

A solution to this problem is a discriminative approach, similar to that taken in classifiers such as logistic regression. The conditional distribution *P*(***y***|***x***) is modeled directly, which is all that is required for classification. Dependencies that involve only variables in ***x*** play no role in *P*(***y***|***x***), so an accurate conditional model can have much simpler structure than a joint model, *P*(***x***,***y***). The posterior probabilities of each label are modeled directly, so no attempt is made to capture the distributions over ***x***, and there is no need to model the correlations between them. Therefore, there is no need to specify an underlying prior statistical model, and the conditional independence assumption of a pixel value given a label, commonly used by generative models, can be relaxed.

This is the approach taken by conditional random fields (CRFs), which are a combination of classification and graphical modeling known as structured prediction (*Lafferty et al., 2001*). They combine the ability of graphical models to compactly model multivariate data (the continuum of landcover and landuse labels) with the ability of classification methods to leverage large sets of input features, derived from imagery, to perform prediction. In CRFs based on ‘local’ connectivity, nodes connect adjacent pixels in ***x*** (*Lafferty et al., 2001*; *Kumar and Herbert, 2006*), whereas in the fully connected definition, each node is linked to every other (*Tappen et al., 2007*; *Krahenbuhl and Koltun, 2011*). CRFs have recently been used extensively for task-specific predictions such as in photographic image segmentation (*Zhu et al., 2001*; *He et al., 2004*; *Chen et al., 2016*; *Garcia-Garcia et al., 2017*) where, typically, an algorithm estimates labels for sparse (i.e. non-contiguous) regions (i.e. supra-pixel) of the image. The CRF uses these labels in conjunction with the underlying features (derived from a photograph), to draw decision boundaries for each label, resulting in a highly accurate pixel-level label image (REFS).

INSERT CONCISE EXPLANATION FOR HOW CRFS WORK HERE

Here, we use the fully connected CRF approach detailed in Krahenbuhl and Koltun (2011). Unary potentials are derived from manual on-screen annotations on imagery. We developed a user-interactive program that segments an image into smaller chunks. On each chunk, cycling through a pre-defined set of classes, the user is prompted to draw (using the cursor) example regions of the image that correspond to each label. Using this information, the CRF algorithm estimates the class of each pixel in the image (Fig. 1). Finally, the image is divided up into tiles of a specified size, *T*. If the proportion of pixels within the tile is greater than a specified amount, *Pclass*, then the tile is written to a file in a folder denoting its class. We prepared a video of this process that is included as Supplemental data A. This simultaneously and efficiently generates both ground-truth label imagery (to evaluate classification performance) and sets of data suitable for training a DCNN. A single photograph typically takes 5-30 minutes to process in this way, depending on the complexity and size of the image, so all the data required to retrain a DCNN (see section below) may take only up to a few hours to generate.

(Figure 1 near here)

**2.2. Retraining a deep neural network (transfer learning)**

Among many suitable popular and open-source frameworks for image classification using deep convolutional neural networks, we chose MobileNets v. 2.0 (*Sandler et al., 2018*) because it is relatively small and efficient (computationally faster to train and execute) compared to many of its competitor architectures designed to be transferable for generic image recognition tasks, such as Inception (*Szegedy et al., 2016*), Resnet (*He et al., 2016*), and NASnet (*Zoph et al., 2017*), and it is smaller and more accurate than Mobilenets v. 1.0 (*Howard et al., 2017*). It also is pretrained for various tile sizes (image windows with horizontal and vertical dimensions of 96, 128, 192, and 224 pixels) which allows us to evaluate the effect of that on classifications. However, all of the aforementioned models are implemented within Tensorflow-Hub (*Tensorflow-Hub, 2018*), which is a library specifically designed for reusing pre-trained TensorFlow (*Abadi et al., 2015*) models on new tasks. Therefore, the interested reader may modify our data and code (provided as Supplemental data X) and explore variation in classification accuracies among multiple DCNN architectures.

For all datasets, we only used tiles (in the training and evaluation) where 90% of the tile pixels were classified as a single class (that is, *Pclass* > 0.9). This avoided including tiles depicting mixed landcover/use classes. We chose tile sizes of *T* = 96x96 pixels and *T* = 224x224 pixels, which is the full range available for Mobilenets, in order to compare the effect of tile size. All model training was carried out in python using tensorflow library version 1.7.0 and tensorflow-hub version 0.1.0. For each dataset, model training parameters (1000 training steps, and a learning rate of 0.01) were kept constant, but not necessarily optimal. For most data sets, there are relatively small numbers of very general classes (water, vegetation, etc) which in some ways is a more difficult classification task than much more specific classes, owing to the greater within-class variability to be expected from having broadly defined categories.

**2.3. CRF-based semantic segmentation**

We have developed a method that harnesses the classification power of the DCNN, with the discriminative capabilities of the CRF, for pixel-scale semantic segmentation of imagery.

Image is windowed into small regions of pixels, the size of which is dictated by the size of the tile used in the DCNN training (here, 96x96 or 224x224 pixels). Some windows, ideally with an even spatial distributed across the image, are classified with a trained DCNN. Collectively, these predictions serve as unary potentials (known labels) for a CRF to build a probabilistic model for pixelwise classification given the known labels and the underlying image

Fig. 2

(Figure 2 near here)

**3. Data**

The chosen data sets that encompass a variety of environments (coastal, fluvial and lacustrine) and collection platforms (oblique stationary cameras, oblique aircraft, nadir UAV, nadir satellite) and types (photographic images and orthomosaics).

(Figure 3 near here)

**3.1. NWPU**

"NWPU-RESISC45", which is a publicly available benchmark for REmote Sensing Image Scene Classification (RESISC), created by Northwestern Polytechnical University (NWPU). The entire dataset, described by *Cheng et al. (2017)*, contains 31,500 images in 45 scene classes with 700 images in each class. The majority of those classes are urban/anthropogenic. We chose to use a subset of 11 classes corresponding to natural landforms and landcover (Fig. 3), namely: beach, chaparral, desert, forest, island, lake, meadow, mountain, river, sea ice, and wetland. All images are 256x256 pixels. We randomly chose 350 images from each class for DCNN training, and 350 for testing.

(Figure 4 near here)

**3.2. Seabright beach, CA.**

The dataset consists of 13 images collected from a fixed-wing aircraft in February 2016, of which a random subset of seven were used for training, and six for testing. Training and testing tiles were generated for seven classes (Table S1 and Fig. 4).

(Figure 5 near here)

**3.3. Lake Ontario, NY.**

The dataset consists of 48 images collected from a UAV in July 2017, of which a random subset of 24 were used for training, and 24 for testing. Training and testing tiles were generated for X classes (Table S2 and Fig. 5).

(Figure 5 near here)

**3.4. Elwha River, WA.**

Orthomosaics generated from images collected in September 2012 and February 2017

16 for training, 8 for testing (Table S3 and Fig. 6)

(Figure 6 near here)

**3.5. Grand Canyon, AZ.**

Riparian environment

The dataset consists of 14 images collected from a stationary autonomous camera system, from 7 sites, of which 1 from each site were used for training, and 1 from each site for testing. Imagery came from various seasons. The camera system, sites and imagery is described in *Grams et al. (2018)*. Table S4 and Fig. 7

(Figure 7 near here)

**4. Results**

**4.1. DCNN transfer learning**

Table 5

Figures S1, S2, S3, S4, S5

**4.2. CRF-based semantic segmentation**

Table 6?

**5. Discussion and Conclusions**

**Acknowledgments**

**References**

Abadi M., Agarwal A., Barham P., Brevdo E., Chen Z., and 35 others. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/about/bib Software available from <https://www.tensorflow.org>

Belgiu, M. and Drăguţ, L., 2016. Random forest in remote sensing: A review of applications and future directions. ISPRS Journal of Photogrammetry and Remote Sensing, 114, pp.24-31.

Y. Boykov, and V. Kolmogorov. An experimental comparison of mincut/max-ﬂow algorithms for energy minimization in vision. IEEE T. Pattern Anal., vol. 26, no. 9, pp. 1124—1137, 2004.

Chen, L.C., Yang, Y., Wang, J., Xu, W. and Yuille, A.L., 2016. Attention to scale: Scale-aware semantic image segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3640-3649).

Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K. and Yuille, A.L., 2018. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence, 40(4), pp.834-848.

Cheng, G., Han, J. and Lu, X., 2017. Remote sensing image scene classification: benchmark and state of the art. Proceedings of the IEEE, 105(10), pp.1865-1883.

Dauphin, Y.N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S. and Bengio, Y., 2014. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In Advances in neural information processing systems (pp. 2933-2941).

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L., 2009, June. Imagenet: A large-scale hierarchical image database. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (pp. 248-255). IEEE.

Franklin, S.E. and Wulder, M.A., 2002. Remote sensing methods in medium spatial resolution satellite data land cover classification of large areas. Progress in Physical Geography, 26(2), pp.173-205.

A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. arXiv preprint arXiv:1704.06857, 2017.

Goodfellow, I., Bengio, Y., Courville, A. and Bengio, Y., 2016. Deep learning (Vol. 1). Cambridge: MIT press.

Grams, P.E., Tusso, R.B. and Buscombe, D., 2018. Automated remote cameras for monitoring alluvial sandbars on the Colorado River in Grand Canyon, Arizona (No. 2018-1019). US Geological Survey Open File Report.

He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M. and Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Hu, F., Xia, G.S., Hu, J. and Zhang, L., 2015. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. Remote Sensing, 7(11), pp.14680-14707. 8-56.

P. Krahenbuhl, and V. Koltun. Efﬁcient inference in fully connected CRFs with Gaussian edge potentials. In Adv. Neur. In. (NIPS), Granada, Spain, pp. 109—117, Dec. 2011.

S. Kumar, and M. Hebert. Discriminative random ﬁelds. Int. J. Comput. Vision, vol. 68, no. 2, pp. 179—201, Jun. 2006.

J. Lafferty, A. McCallum, and F. Pereira. Conditional random ﬁelds: Probabilistic models for segmenting and labeling sequence data. Int. Conf. Mach. Learn. (ICML), Williamstown, USA, pp. 282—289, July 2001.

M. Längkvist, A. Kiselev, M. Alirezaie, and A. Loutfi, “Classification and Segmentation of Satellite Orthoimagery Using Convolutional Neural Networks,” Remote Sensing, vol. 8, no. 4, pp. 329, 2016.

LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. Nature, 521(7553), p.436.

Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A., van Ginneken, B. and Sánchez, C.I., 2017. A survey on deep learning in medical image analysis. Medical image analysis, 42, pp.60-88.

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y. and Alsaadi, F.E., 2017. A survey of deep neural network architectures and their applications. Neurocomputing, 234, pp.11-26.

Long, J., Shelhamer, E. and Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).

Lu, H., Fu, X., Liu, C., Li, L.G., He, Y.X. and Li, N.W., 2017. Cultivated land information extraction in UAV imagery based on deep convolutional neural network and transfer learning. Journal of Mountain Science, 14(4), pp.731-741.

Ma, L., Li, M., Ma, X., Cheng, L., Du, P. and Liu, Y., 2017. A review of supervised object-based land-cover image classification. ISPRS Journal of Photogrammetry and Remote Sensing, 130, pp.277-293.

Maggiori, E., Tarabalka, Y., Charpiat, G. and Alliez, P., 2017. Convolutional neural networks for large-scale remote-sensing image classification. IEEE Transactions on Geoscience and Remote Sensing, 55(2), pp.645-657.

D. Marmanis, M. Datcu, T. Esch, and U. Stilla, “Deep Learning Earth Observation Classification Using ImageNet Pretrained Networks,” IEEE Geosci. Remote Sens. Lett., vol. 13, no. 1, pp. 105-109, 2016.

Mulder, V.L., De Bruin, S., Schaepman, M.E. and Mayr, T.R., 2011. The use of remote sensing in soil and terrain mapping—A review. Geoderma, 162(1-2), pp.1-19.

O’Connor, J., Smith, M.J. and James, M.R., 2017. Cameras and settings for aerial surveys in the geosciences: Optimising image data. Progress in Physical Geography, 41(3), pp.325-344.

Palafox, L.F., Hamilton, C.W., Scheidt, S.P. and Alvarez, A.M., 2017. Automated detection of geological landforms on Mars using Convolutional Neural Networks. Computers & geosciences, 101, pp.4

Olga Russakovsky\*, Jia Deng\*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (\* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. and Chen, L.C., 2018. Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation. arXiv preprint arXiv:1801.04381.

Sekovski, I., Stecchi, F., Mancini, F. and Del Rio, L., 2014. Image classification methods applied to shoreline extraction on very high-resolution multispectral imagery. International journal of remote sensing, 35(10), pp.3556-3578.

Smith, M.J. and Pain, C.F., 2009. Applications of remote sensing in geomorphology. Progress in Physical Geography, 33(4), pp.568-582.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2818-2826).

Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A.A., 2017, February. Inception-v4, inception-resnet and the impact of residual connections on learning. In AAAI (Vol. 4, p. 12).

M. Tappen, C. Liu, E. Adelson, and W. Freeman. Learning Gaussian conditional random ﬁelds for low-level vision. IEEE. Conf. Comp. Vision (CVPR), Minneapolis, USA, pp. 1—8, June 2007.

Tensorflow-Hub, 2018. <https://www.tensorflow.org/hub/modules/image>. Accessed: 5/23/2018

Volpi, M. and Tuia, D., 2017. Dense semantic labeling of subdecimeter resolution images with convolutional neural networks. IEEE Transactions on Geoscience and Remote Sensing, 55(2), pp.881-893.

H. Zhu, F. Meng, J. Cai, and S. Lu. Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation. J. Vis. Commun. Image R., vol. 34, pp .12—27, 2016.

Zoph, B., Vasudevan, V., Shlens, J. and Le, Q.V., 2017. Learning transferable architectures for scalable image recognition. arXiv preprint arXiv:1707.07012.

**Tables**

Table 1: Accuracies, F1 scores and mean classification probabilities for each data set and tile size

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ***T* = 96** | | | ***T* = 224** | | |
| **Data set** | **Mean accuracy** | **Mean F1 score** | **Mean probability** | **Mean accuracy** | **Mean F1 score** | **Mean probability** |
| 1. NWPU | 87% | 93% | 0.87 | 89% | 94% | 0.89 |
| 2. Seabright | 94% | 97% | 0.94 | 91% | 95% | 0.92 |
| 3. Ontario | 83% | 91% | 0.87 | 96% | 98% | 0.96 |
| 4. Elwha | 87% | 93% | 0.84 | 85% | 91% | 0.88 |
| 5. Grand Canyon | 92% | 96% | 0.93 | 94% | 97% | 0.95 |

**Figures**

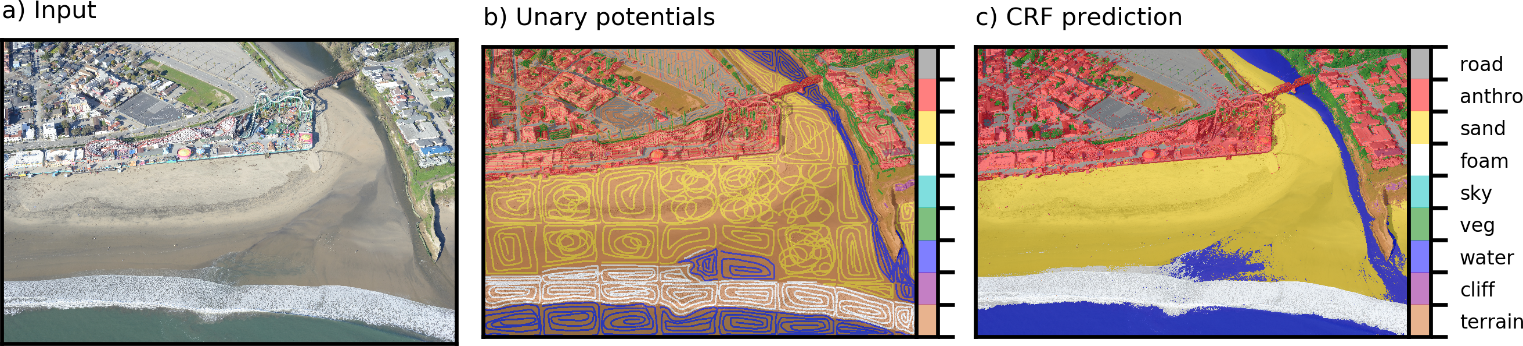


Figure 1. Application of the semi-supervised CRF at Seabright beach for generation of DCNN training tiles and ground-truth labeled images. From left to right, the input image, the hand-annotated sparse labels, and the resulting CRF-predicted pixelwise labeled image.

Figure 2. Application of the unsupervised CRF for pixelwise classification, based on unary potentials of regions of the image classified using a DCNN. Example comes from XXXX. From left to right, the input image, the DCNN-estimated sparse labels, and the resulting CRF-predicted pixelwise labeled image.



Figure 3. Example tiles from NWPU data set. Classes, from left to right, are beach, chaparral, desert, forest, island, lake, meadow, mountain, river, sea ice, and wetland.



Figure 4. Example tiles from Seabright beach. Classes, from left to right, are anthropogenic/buildings, foam, road/pavement, sand, other natural terrain, and vegetation.



Figure 5. Example tiles from Lake Ontario shoreline. Classes, from left to right, are anthropogenic/buildings, sediment, other natural terrain, vegetation, and water.

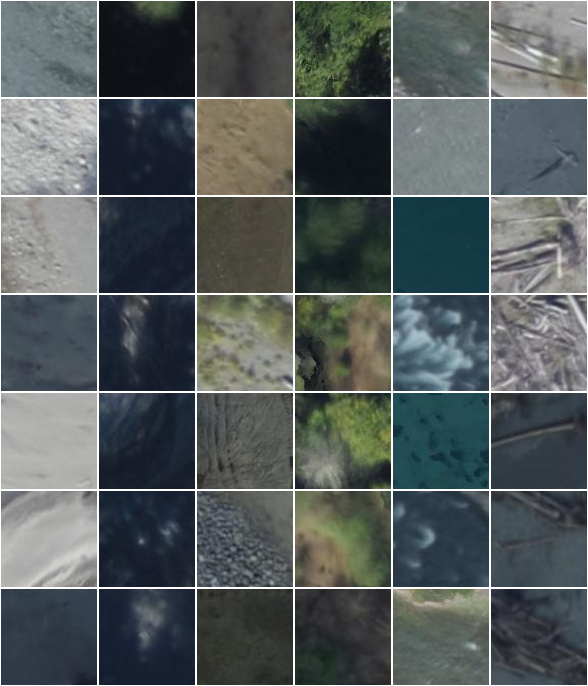


Figure 6. Example tiles from Elwha. Classes, from left to right, are sediment, shadow, other natural terrain, vegetation, water, and wood.



Figure 7. Example tiles from Grand Canyon. Classes, from left to right, are rock/scree, sand, vegetation, and water.

**Supplemental Tables**

Table S1: Classes and number of tiles used for the Seabright data set

|  |  |  |
| --- | --- | --- |
| **Class** | **Number of training tiles (*T*=96/224)** | **Number of evaluation tiles (*T*=96/224)** |
| Anthropogenic | 23,566 / 4,548 | 15,575 / 3,031 |
| Road and pavement | 314 / 60 | 525 / 103 |
| Sand | 38,250 / 6,887 | 25,318 / 5,802 |
| Vegetation | 386 / 76 | 240 / 38 |
| Other terrain | 77 / 24 | 117 / 22 |
| Water | 11,394 / 1,723 | 14,360 / 2,251 |
| Foam | 5,076 / 735 | 5,139 / 843 |
| Total: | 76,063 / 14,053 | 61,274 / 12,090 |

Table S2: Classes and number of tiles used for the Ontario data set

|  |  |  |
| --- | --- | --- |
| **Class** | **Number of training tiles (*T*=96/224)** | **Number of evaluation tiles (*T*=96/224)** |
| Anthropogenic/buildings | 467 / 219 | 3,216 / 333 |
| Sediment | 2,856 / 289 | 3,758 / 407 |
| Vegetation | 33,871 / 5,139 | 33,421 / 5,001 |
| Other terrain | 1,596 / 157 | 1,094 / 92 |
| Water | 80,304 / 13,332 | 77,571 / 12,950 |
| Total: | 119,094 / 19,136 | 119,060 / 18,783 |

Table S3: Classes and number of tiles used for the Elwha data set

|  |  |  |
| --- | --- | --- |
| **Class** | **Number of training tiles (*T*=96/224)** | **Number of evaluation tiles (*T*=96/224)** |
| Sediment | 1,095 / 209 | 730 / 57 |
| Shadow/null | 3,792 / 329 | 1,822 / 144 |
| Other terrain | 1,410 / 210 | 920 / 46 |
| Vegetation | 22,899 / 3,941 | 6,493 / 964 |
| Water | 2,253 / 205 | 974 / 95 |
| Wood | 799 / 84 | 398 / 25 |
| Total: | 32,248 / 4,978 | 11,337 / 1,331 |

Table S4: Classes and number of tiles used for the Grand Canyon data set

|  |  |  |
| --- | --- | --- |
| **Class** | **Number of training tiles (*T*=96/224)** | **Number of evaluation tiles (*T*=96/224)** |
| Rock/scree/terrain | 15,059 / 2,405 | 12,151 / 1,999 |
| Sand | 751 / 39 | 1,069 / 91 |
| Riparian vegetation | 2,971 / 408 | 2,158 / 305 |
| Water | 8,568 / 1,462 | 5,277 / 1,130 |
| Total: | 27,349 / 4,314 | 20,655 / 3,525 |

**Supplemental Figures**



Figure S1. Matrices of correspondences (proportion correctly classified) between true (left axis) and DCNN-estimated (bottom axis) labels, based on tiles generated from NWPU imagery, on size 96 (A) and 224 (B) pixels.

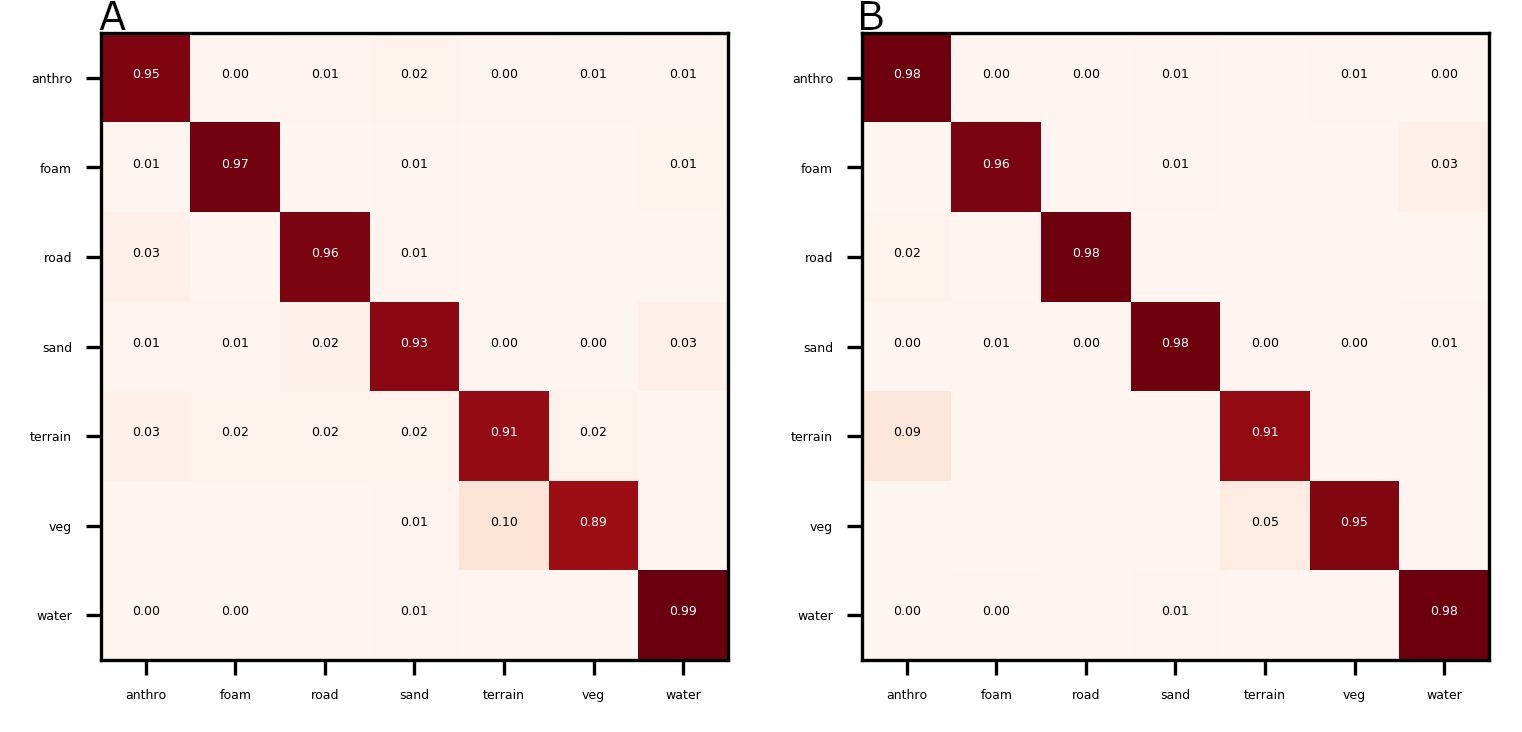


Figure S2. Matrices of correspondences (proportion correctly classified) between true (left axis) and DCNN-estimated (bottom axis) labels, based on tiles generated from imagery at Seabright beach, on size 96 (A) and 224 (B) pixels.

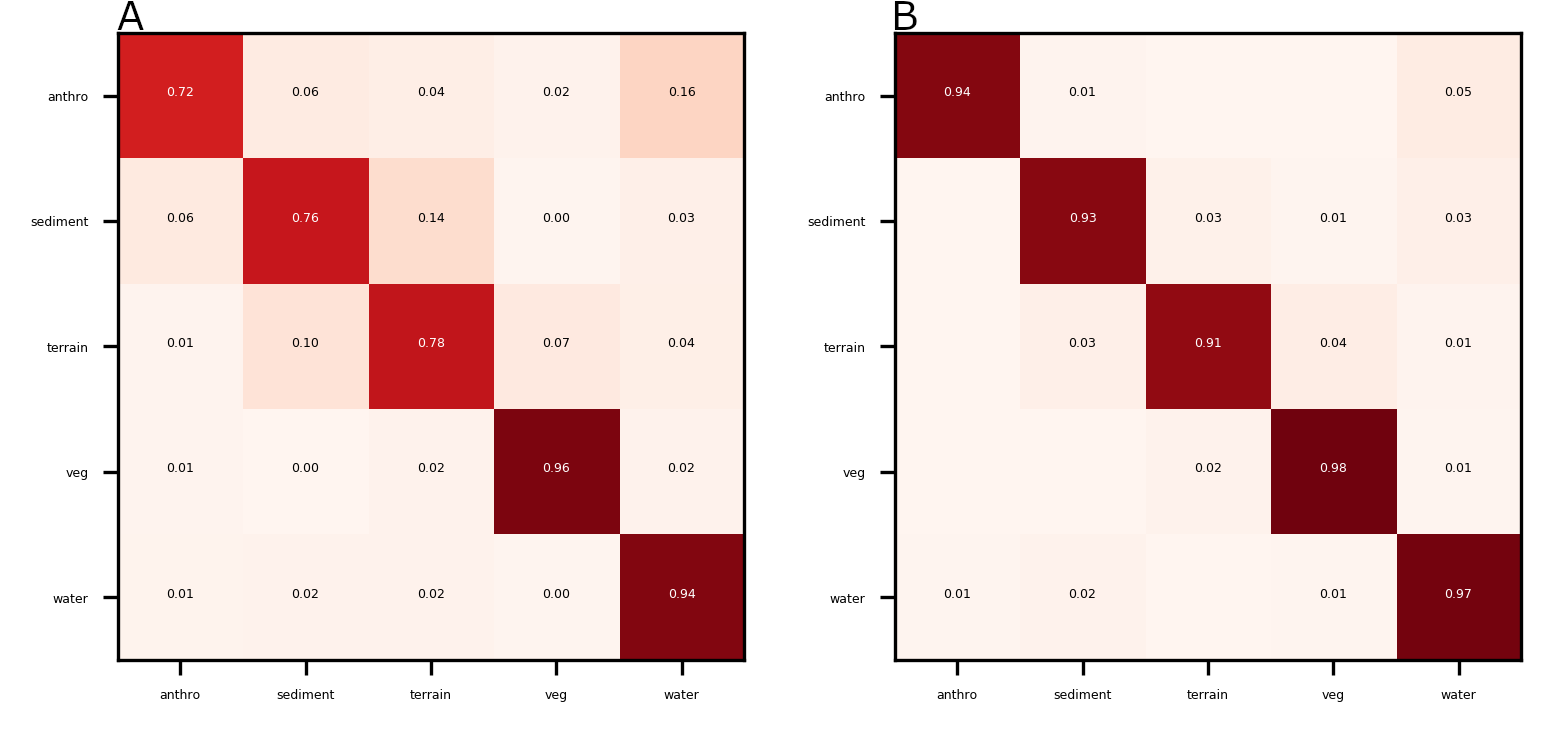


Figure S3. Matrices of correspondences (proportion correctly classified) between true (left axis) and DCNN-estimated (bottom axis) labels, based on tiles generated from imagery at Lake Ontario, on size 96 (A) and 224 (B) pixels.

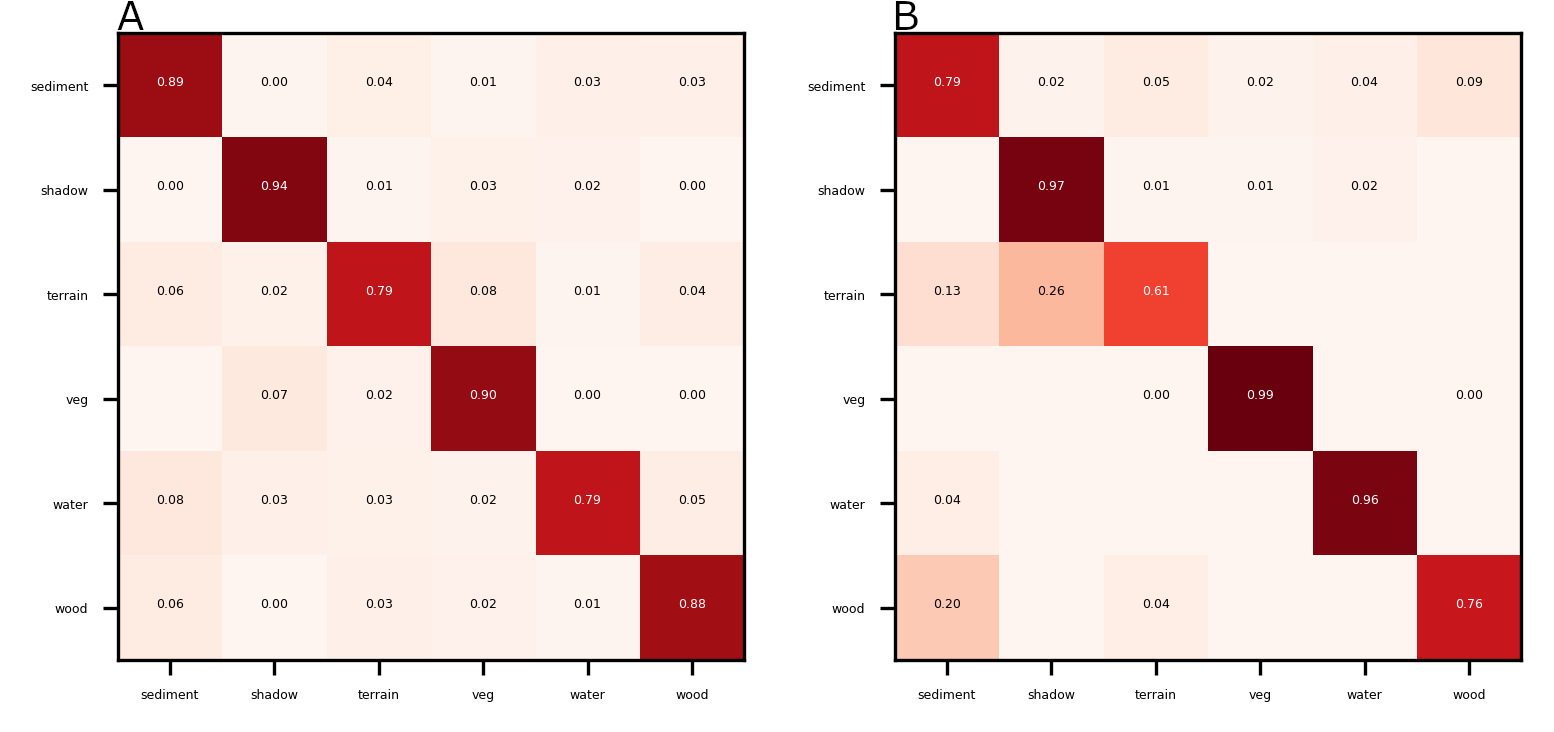


Figure S4. Matrices of correspondences (proportion correctly classified) between true (left axis) and DCNN-estimated (bottom axis) labels, based on tiles generated from Elwha imagery, on size 96 (A) and 224 (B) pixels.

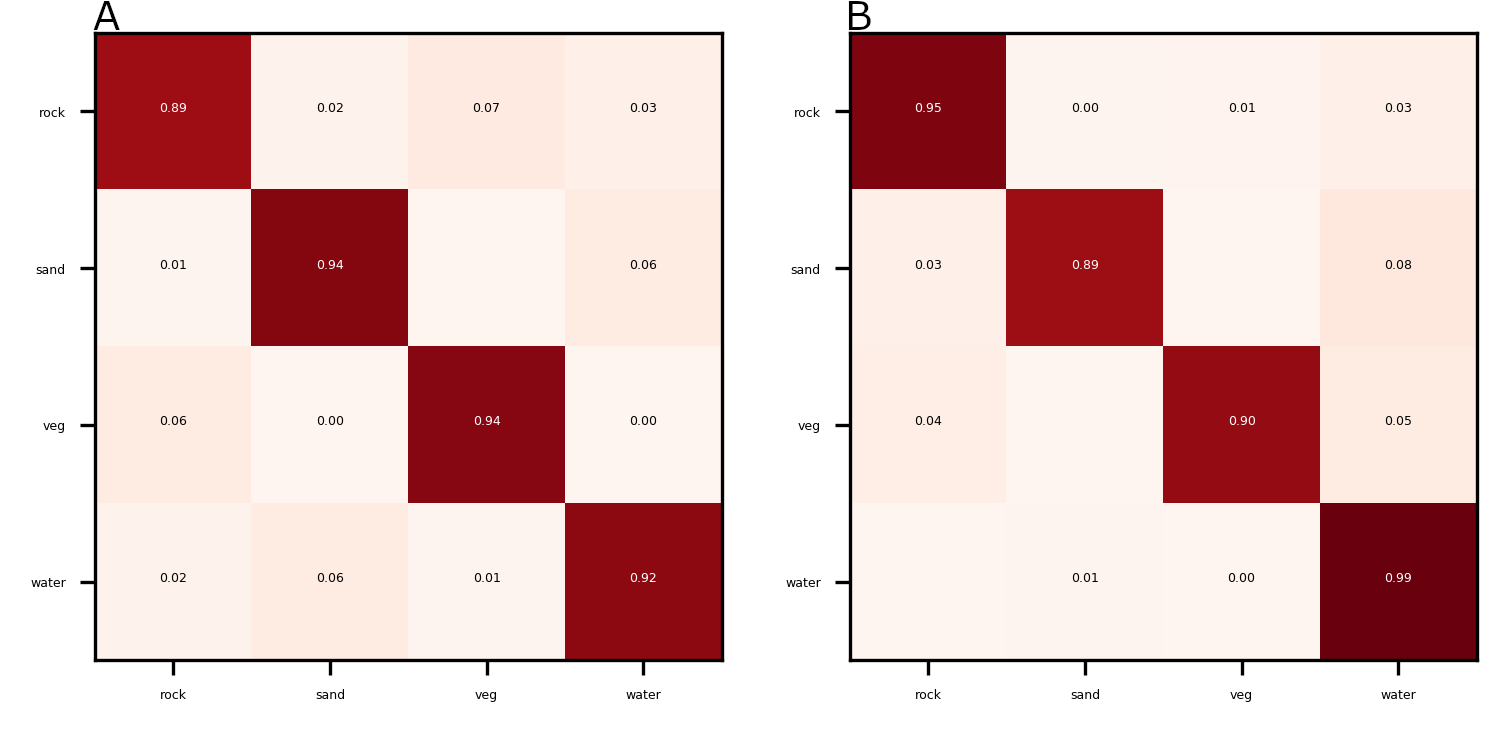


Figure S5. Matrices of correspondences (proportion correctly classified) between true (left axis) and DCNN-estimated (bottom axis) labels, based on tiles generated from imagery at Grand Canyon, on size 96 (A) and 224 (B) pixels.