JOINT HEIGHT ESTIMATION AND SEMANTIC LABELING OF MONOCULAR AERIAL IMAGES WITH CNNS

Shivangi Srivastava, Michele Volpi, Devis Tuia

MultiModal Remote Sensing, Department of Geography, University of Zurich (Switzerland)

{shivangi.srivastava, michele.volpi, devis.tuia}@geo.uzh.ch

ABSTRACT

We aim to jointly estimate height and semantically label monocular aerial images. These two tasks are traditionally addressed separately in remote sensing, despite their strong correlation.Therefore, a model learning both height and classes jointly seems advantageous and so, we propose a multitask Convolutional Neural Network (CNN) architecture with two losses: one performing semantic labeling, and another predicting normalized Digital Surface Model (nDSM) from the pixel values. Since the nDSM/height information is used only in the second loss, there is no need to have a nDSM map at test time, and the model can estimate height automatically on new images. We test our proposed method on a set of sub-decimeter resolution images and show that our model equals the performances of two separate models, but at the cost of a single one.

Index Terms— Convolutional neural networks, Multitask learning, Digital Surface Model, Semantic labeling.

1. INTRODUCTION

We deal with the multi-task problem of estimating jointly a normalized Digital Surface Model (nDSM) and a semantic label map (land-use and land-cover) from single (monocular) images. These two products are very useful in a variety of applications and a single model providing them both as an output is highly desirable.

There are many ways to generate a nDSM: among them are stereo-pair photogrammetry [1], multi-angular/triplet photogrammetry [2], SAR interoferometry [3], structure from motion or Lidar processing. These methods are able to provide high to very high resolution nDSMs, but a lot of pre- and post-processing is required, including accurate denoising of the input data.Also, these methods are expensive, since they require both expert knowledge and high computational effort. Therefore, their deployment on a routine basis is at best difficult. Moreover, when considering historical aerial photography (i.e. when Lidar acquisitions or overlapping stereo pairs cannot be acquired), generating height maps becomes very challenging and can be achieved only at coarse scale by using existing high resolution DSM [4, 5]. In this paper, we propose a method to overcome these issues and construct height maps from a single aerial image.

The authors acknowledge the Swiss National Science Foundation (no. PP00P2-150593).

In computer vision this problem is known as depth estimation. Most literature considers techniques using stereo image pairs [6, 7]. In the last few years efforts have been made to generate depth map from single RGB images. In [8], authors propose a model to generate 3D depth maps from single images using hierarchical, multiscale Markov Random Field (MRF) learning model. In [9], authors regress the depth using a multi-scale CNN trained over single images. The last strategy is particularly appealing, since it casts the depth estimation problem as a regression one and only needs RGB channels to perform the estimation at test time.

Compared to the tasks above, performing depth estimation in aerial images bears additional complexity, since different objects can be composed of the same material (e.g. roofs and streets). To tackle this problem, we rely on the assumption that abrupt transitions in height often correspond to changes in classes and vice versa. Since CNNs are proven to be very effective in semantic class labeling [10, 11], we encode this assumption by using a multi-task CNN predicting both outputs jointly using only one image as input. This is done by training over a loss composed by a linear combination of sub-losses. Such use of multi-task CNNs would make learning multiple related tasks simpler, save time and effort if compared to training of task-specific CNNs, and possibly make learning models better. Recent works in vision go in this direction: [12] use a single CNN architecture to solve multiple problems, including object detection, semantic labeling, normal and depth estimation. In [13], authors used a single multiscale convolutional network architecture to address tasks of depth prediction, surface normal estimation, and semantic labeling simultaneously, while in [14] authors show the possibility of predicting semantic labeling as well depth from a single image using a multitask CNN.

Our method is closely related to the latter, but has the advantage of simplicity, since it does not include object proposals generation, nor complex post-processing steps: after learning on a set of image/nDSM pairs, it can predict height on unseen test images directly, while generating semantic labeling maps at the same time. It is therefore suited for routine applications involving sub-decimeter resolution images or data enrichment of historical images.

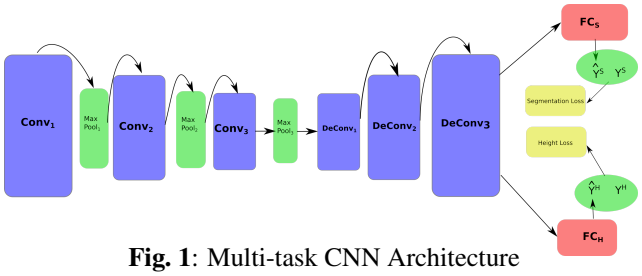

Fig. 1: Multi-task CNN Architecture

2. METHOD

Though it seems natural for our brain to work on many tasks seamlessly, it is less natural to train a CNN to solve multiple problems jointly. However, recent advances [12, 14] show that a CNN can be trained to solve multiple tasks in a single training schedule. In general, we can define a task as a loss to be minimized, and therefore multi-task learning involves the joint optimization of more than one training loss functions.

2.1. Joint Training Loss Function

Formally, a multi-task training loss function over D tasks can be defined by the following equation:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{d=1}^{D} \lambda_d \mathcal{L}^d(\mathbf{y}^d, \hat{\mathbf{y}}^d), \quad (1)$$

where $\mathbf{y}$ is a D-dimensional vector containing ground truth values for image $\mathbf{x}$ and $\hat{\mathbf{y}} = f(\mathbf{x})$ is the corresponding vector of predictions of CNN f. The hyperparameter λ_d weights losses $\mathcal{L}_d$ pertaining to each task.

Our method optimizes the network weights over two losses, corresponding to the tasks of semantic labeling and nDSM prediction. The former is defined in Eq. (2), and corresponds to a soft-max cross-entropy loss. The latter is defined in Eq. (3), and corresponds to a mean squared error loss. The joint loss function is a weighted combination of the two losses, as described as in Eq. (1).

$$\mathcal{L}_S(\mathbf{y}^S, \hat{\mathbf{y}}^S) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} \log\left(p(y_i = c|\mathbf{x}_i)\right) \quad (2)$$

$$\mathcal{L}_H(\mathbf{y}^H, \hat{\mathbf{y}}^H) = \frac{1}{N} \sum_{i=1}^{N} (y_i^H - \hat{y}_i^H)^2 \quad (3)$$

In the above loss functions, $\hat{y}^S$ and $\hat{y}^H$ denote respectively the CNN output for the semantic labeling (*S*) and nDSM/height (*H*) estimation. In the first case, the loss corresponds to the average negative log-likelihood for each one of the C classes, for each data sample composing the training minibatch. In the second, we train the network to minimize the average mean square error over the minibatch.

2.2. Our CNN Model Architecture

Figure 1 summarizes the architecture. Table 1 presents the input / output volumes of each layer. We follow a pyramidal encoder-decoder CNN, inspired by [10, 15], which means that we first spatially downsample the activations and then we upsample them to the size of the original inputs. We do so to allow the CNN to learn spatial arrangement of features at different scales, learning short to long range relationships across object classes and spatial features. The network is composed of a common trunk, sharing parameters among both tasks, followed by two, task specific, fully connected layers.

Table 1: Number of layers per block, number of filters per layer, dimension of input and output for a given block.

Blocks	#3x3 layers	#filters /layer	input	output
Conv1	4	64	65x65x3	34x34x64
Conv2	3	128	34x34x64	18x18x128
Conv3	2	256	18x18x128	9x9x256
Deconv1	1	512	9x9x256	17x17x256
Deconv2	1	512	17x17x256	33x33x512
Deconv3	1	512	33x33x512	65x65x512
FC1	1	512	65x65x512	65x65x1
FC2	1	512	65x65x512	65x65x6

Common trunk. The common trunk learns shared representation from the images, which carries the information for both the semantic labeling and height estimation task. It is composed by three downsampling convolutional blocks (Conv_1, Conv_2, Conv_3), followed by three upsampling deconvolutional blocks (Deconv_1, Deconv_2, Deconv_3).

- *Downsampling convolutional blocks.* We limit the size of convolution kernels to 3×3, as in the VGG network [16]. In few words, by replacing a filter of size 7×7 by a stack of 3 layers of size 3×3, filters can achieve the same receptive field size, while using less parameters (27 instead of 49 for each filter). Moreover, this allows us to put nonlinearities across blocks, making the model more flexible overall. In all layers, we employ ReLU nonlinearities of the form $y = \max(0, x)$. After each block we summarize spatial activations by using max-poolings of non-overlapping 2×2 blocks. Specifically, in our network we downsample with stacks of 4 (Conv_1), 3 (Conv_2) and 2 (Conv_3) 3×3 filters, corresponding to receptive fields of 9×9, 7×7 and 5×5, respectively (see Tab. 1).
- *Upsampling deconvolutional blocks.* All the deconvolution blocks are made of $3\times$ 3 deconvolutions (single filters) and ReLU layers, learning an upsampling of the activations of the previous block by a factor 2.

Task-specific layers and predictions. To learn separate tasks, the ouptput out of the Deconv_3 block ($\mathbf{x}_{DC3}$) enters two separate fully connected layers. Each fully connected layer (FC_S and FC_H, respectively) acts on each spatial location of $\mathbf{x}_{DC3}$, which has the same size as the original input image. FC_S leads to the prediction of semantic segmentation labels, while FC_H predicts the nDSM, both pixel-wise.

We formulate the output of each task in Eqs. (4) and (5). Specifically, each fully connected layer learns the pairs ($\mathbf{w}^S$, b^S) and ($\mathbf{w}^H$, b^H). For semantic labeling, we pass the output of FC_S into a soft-max to retrieve local probability scores (Eq. (4)), while for height estimation task we directly employ the output of FC_H as the map to the height values (Eq. (5)) :

$$p(\hat{y}_i{}^S|\mathbf{x}_i) = \frac{\exp(\mathbf{w}_c^S\mathbf{x}_i + b^S)}{\sum_c \exp(\mathbf{w}_c^S\mathbf{x}_i + b^S)} \quad (4)$$

$$\hat{y}_i{}^H = \mathbf{w}^H\mathbf{x}_i + b^H \quad (5)$$

Note that the final prediction for the semantic labeling task is given by the class label maximizing the posterior for each output location, or $\hat{y}_i^S = \arg\max_c p(\hat{y}_i^S = c|\mathbf{x}_i)$

Backpropagation. In the architecture proposed, the global loss is a weighted sum of sub-losses, as in (1). When backpropagating the gradient in each fully connected layer, we compute the partial derivatives with respect to ($\mathbf{w}^S$, b^S) and ($\mathbf{w}^H$, b^H). Thus, when backpropagating into FC_S, $\frac{\partial\mathcal{L}}{\partial(\mathbf{w}^S,b^S)}$ makes ($\mathbf{w}^H$, b^H) related terms vanish, and the same applies when computing gradients wrt $\frac{\partial\mathcal{L}}{\partial(\mathbf{w}^H,b^H)}$. The fully connected layers map gradients to the dimensionality of DeConv_3, where they are summed. In our formulation, we only employ a single hyperparameter λ weighting the losses by multiplying the loss value (and therefore the gradient) of the FC_H. By doing so, gradients are comparable when summed in DeConv_3.

3. DATA, SETUP AND RESULTS

In the experiments we employed the Vaihingen dataset, freely available in the ISPRS 2D semantic labeling challenge[1]. The data has been acquired over the city of Vaihingen, in Germany. It consists of aerial imagery (near infrared, red, green bands – NIR-R-G) along with a Digital Surface Model and a dense annotation ground truth for semantic labels. The ground sampling distance of both the NIR-R-G images and the DSM is 9 cm. A normalized DSM is available from [17].

We divided the dataset in a training set composed of 11 images and a test set composed of 5 images. We train our multi-task CNN by sampling 65×65 patches from the image data. Specifically, 400 samples per class were extracted from each training image. In order to have control over the semantic classes that the network sees during training, we counted the number of times a patch was *centered* on a given class. Thus, a total of 26400 patches were used while training. We augmented the training set by applying random rotations, flippings, and jittering. During training, we resampled the training set every 10 epochs, i.e. the whole training set is backpropagated 10 times before being resampled.

We train the network to predict class likelihoods (Eq. (4)) and nDSM scores (Eq. (5)) simultaneously. Since input-output spatial dimensions match, for each training patch we used the corresponding 65×65 patches from the dense semantic labels and the nDSM of [17] as ground truth. For training, the gradients of semantic labeling loss Eq.(2) and height loss Eq.(3) were jointly back-propagated in the network. The learning rate started at 0.01 and was decreased by a factor of 10 every ten epochs. Weight decay was set to 0.0001 and momentum to 0.9.

We compare our multi-task model with two single-task architectures, one for semantic labeling and the other one for height estimation. The single task models have exactly the same architecture of the multi-task model, except that the loss associated to the task not addressed is set to 0. This way, the comparisons are fair, as the architecture is composed by the same number of parameters, as long as a single task is concerned. Training procedure is kept as for the multi-task setting (by only considering one output at a time) and hyperparameters are also fixed to the same values.

4. RESULTS AND DISCUSSION

An example of the predicted semantic labels and corresponding nDSM map is given in Fig. 2. The maps produced by the multi-task trained jointly (Fig. 2(d)-(e)) do not significantly differ from those obtained by models trained specifically on each task (Fig. 2(f)-(g)). In the semantic segmentation maps in particular, both single and multitask models led to similar errors when visual appearance of classes is extremely ambiguous, such as some buildings that are confused with roads and viceversa (remember that nDSM is not used as an input).

Nonetheless, the prediction of some classes in the semantic labeling map seems smoother probably due to the training over different joint tasks, which leads to stronger regularization. In our case, the joint prediction of the NDSM seems to selectively regularize specific ground cover classes. When considering height prediction, the multi-task model seems to lead to both more consistent predictions in vegetation areas and better delineation of buildings.

Results are reported in Table 2 for both tasks. Height estimation is solved accurately, with an RMSE of 0.098. The single task model perform slightly worse, with an error of 0.099. This indicates that the multi-task model is as accurate as the model trained specifically for height estimation.

The semantic labeling task is solved with an average F1 score across classes of 62.39% (74.28% without considering the clutter class, as it is done in the official evaluation of results for this dataset) and an overall accuracy (OA) of 78.65% (78.79% without clutter). Performances for the single-task model are very similar or slightly worse. Although not exhaustive in term of exploration of the different architectures, these results show that joint semantic labeling and height estimation can be solved accurately, as much or slightly better than when using single-task specialized models. Since the architecture of the single-task models is the same, except for the absence of one fully connected layer, the computational cost is halved when training a single multi-task model.

[1] http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html

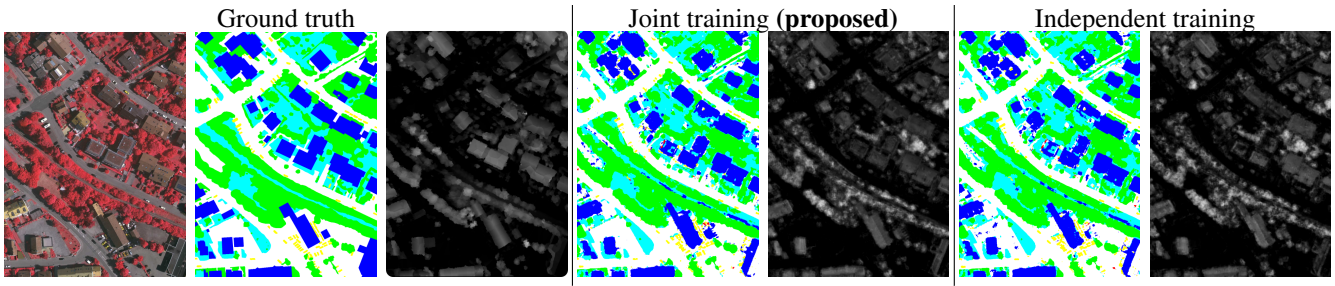

Fig. 2: Numerical results on one tile of the Vaihingen dataset: (a)-(c) ground truth. (d)-(e) predictions of the proposed multi-task model; (g)-(g) predictions of CNNs trained independently.

Table 2: Semantic segmentation and height estimation scores. First row: multi-task model (MT), second row: single task model (ST). OA: Overall accuracy, AA: Average per-class accuracy (Producer's).

	Semantic labeling										**Height estimation**	
	Per-class F1 scores						Global measures					
	Roads	Buil.	Low. veg.	Trees	Cars	Clutter	Av. F1	OA	Kappa	AA	MAE	RMSE
MT loss	81.23	86.24	67.29	80.11	56.16	3.31	62.39 (74.28†)	78.65 (78.79†)	71.75 (71.92†)	61.54 (73.39†)	0.063	0.098
ST loss	81.01	85.73	67.59	79.98	56.46	2.07	62.14 (74.22†)	78.44 (78.56†)	71.49 (71.63†)	61.35 (73.33†)	0.063	0.099

† corresponds to scores computed without considering "clutter"

5. CONCLUSIONS

We presented a multi-task model based on Convolutional Neural Networks (CNN) to jointly perform semantic labeling and height estimation from monocular, sub-decimeter resolution aerial images. The network provides a semantic labeling map, together with a normalized digital surface model for a given input color image. Therefore, it can be used for automatic height estimation. Results indicate that multi-task learning relying on CNN i) is flexible and accurate to solve multiple but related tasks jointly, and ii) reduces training and test time proportionally to the number of tasks.

6. REFERENCES

[1] J. Raggam, M. Buchroithner, and R. Mansberger, "Relief mapping using non-photographic spaceborne imagery." *ISPRS J. Int. Soc. Photo. Remote Sens.*, 1989.

[2] H. Raggam, "Surface mapping using image triplets: Case studies and benefit assessment in comparison to stereo image processing." *Photo. Eng. Remote Sens.*, 2006.

[3] A. Ferretti, "Guidelines for sar interferometry processing and interpretation," ESA, Tech. Rep. TM-19, 2007.

[4] G. Baatz, O. Saurer, K. Koser, and M. Pollefeys, "Large scale visual geo-localization of images in mountainous terrain." in *European ConfereECCVnce on Computer Vision*, 2012.

[5] T. Produit, D. Tuia, V. Lepetit, and F. Golay, "Pose estimation of web-shared landscape pictures." in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II, no. 3, 2014, pp. 127–134.

[6] W. Hoff and N. Ahuja, "Surfaces from stereo: integrating feature matching, disparity estimation, and contour detection." *IEEE Trans. Pattern Anal. Mach. Intell.*, 1989.

[7] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light." in *CVPR*, 2003.

[8] A. Saxena, S. H. Chung, and A. Y. Ng, "3-d depth reconstruction from a single still image," *Int. J. Computer Vis.*, vol. 76, no. 1, pp. 53–69, 2008.

[9] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network." in *NIPS*, 2014.

[10] M. Volpi and D. Tuia, "Dense semantic labeling of subdecimeter resolution images with convolutional neural networks," *IEEE Tran. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 881–893, 2017.

[11] J. Sherrah, "Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery." Tech. Rep. arXiv:1606.02585, 2016.

[12] I. Kokkinos, "Ubernet: Training a 'universal' convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory'." Tech. Rep. arXiv:1609.02132, 2016.

[13] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture." Tech. Rep. arXiv:1411.4734, 2014.

[14] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. Yuille, "Towards unified depth and semantic prediction from a single image." in *CVPR*, 2015.

[15] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *ICCV*, 2015.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition." in *ICLR*, 2015.

[17] M. Gerke, "Use of the stair vision library within the ISPRS 2D semantic labeling benchmark (Vaihingen)." ITC, University of Twente., Tech. Rep., 2015.