Road Segmentation with a Two-Layer Model and the Hough transform

Kevin Kipfer, Lukasz Dykcik, Matthias Erdin
Group: *0xdeadbeef*
Department of Computer Science, ETH Zürich, Switzerland

***Abstract*—Satellites produce large amounts of data that need to be processed. An automated approach is needed, because visual inspection does not scale. One of the major challenges is to distinguish roads from background, the so called road segmentation problem. In order to solve this problem, we trained a two-layer model on 100 satellite images with the goal to detect streets in 50 test images. Our layers consist of classical methods, namely Gradient Boosting and Random Forest classifiers. By estimating color probability and using a technique to detect straight roads based on the Hough transform, we were able to obtain an F1 measure of 0.895. We show that it is possible to achieve good results without using neural networks.**

I. Introduction

Since satellite images became widely available, an automated way of processing them is required. Detecting road regions provides valuable information that leads to a better understanding of the content of satellite images. [1]

In this paper we propose a novel solution to the road segmentation problem. The solution combines local methods, like looking at colors of a single pixel, with a more global view, such as using the Hough transform in order to find straight lines. The different features can be successfully analyzed with a two-layer model using a Gradient Boosting and a Random Forest classifier.

In comparison to the two baseline methods logistic regression and one layered convolutional neural network, our solution produces more accurate results. A comparison between the baseline algorithms and our solution is presented in the Results section. Other, more sophisticated methods usually require large amounts of data to train the classifier [2] or take advantage of data coming from other sources, such as road data vectors [3] or graphs representing the street network [4]. In contrast to most other solutions, our method can segment roads well even if trained on small data sets. Moreover it does not require any additional hints about the road map. Unlike the popular Neural Networks [5], our method allows to control and understand the used features and the inner working of the classifier.

II. Models and Methods

Detecting Streets in satellite images is a challenging task. Although they usually have some common structure and a more or less constant width, the appearance varies a lot caused by different illuminations, the wide spread use of concrete and the fact that streets are sometimes hidden by trees and other objects. Moreover a street during rush hour doesn't differ much from a car park, a street covered by trees looks similar to a forest and even the roofs of buildings can be colored the same way. In order to distinguish such cases we require features that capture the structure not only of the patch itself but from its surroundings too. Hence we extracted a 5 by 5 patches wide area around each patch and tried to learn their appearance.

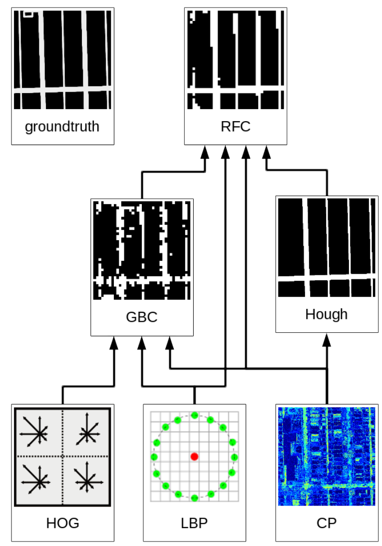

Figure 1. Architecture of the final approach. A two-layer model that uses HOG, LBP, CP and Hough as input. The first layer is called Leaf Model and and uses a Gradient Boosting Classifier (GBC). The second layer uses the output of the first layer and the Hough transform as a feature and trains a Random Forest Classifier (RFC). (Image sources: [6] [7])

In the end we chose 3 features that are widely used in object recognition tasks: HOG, LBP and a histogram that captures information about the colors. Since those features alone miss a lot of information we derived our own feature called color probability (cp). Furthermore we extended the Hough transform on top of cp such that it can give us a more global view on the image.

We then present two models that solve the road segmentation task. The first approach is a somewhat conservative approach called leaf model. We gather features and feed it into a classifier. The second method tries to improve the detection performance by adding a second layer on top of the leaf model as can be seen in figure 1.

A. Color Probability

We can use colors of individual pixels to detect roads. Each pixel is represented as three 8-bit values, one value for each color channel, that means each color was encoded with a 24-bit number. We counted how many times each color was observed and how many times this color was marked as a road. Then, by dividing those two values we could calculate how often the particular color is a road and treat that value as the probability that the observed color belongs to a road. Probabilities of missing colors were linearly interpolated using values of the nearest colors with already calculated probability.

Finally, using the probability values for each possible color, we can detect roads in test images. We process the image pixel by pixel and assign a probability value for each pixel. As a result we get a black and white image with bright regions where colors have a high probability to be part of a road and dark regions where it is unlikely. Figure 2 presents an example of the output of the prediction based on color probabilities.

Figure 2. Ouptut of Color Probability Model

B. Hough-based Road Detection

The majority of roads in the processed data set are straight roads reaching from edge to edge in an image. An important subtask therefore is the detection of lines in images. A well-known feature extraction technique for this purpose is the Hough transform [8]. One well working example can be seen in figure 3. The output here only contains lines that reach from edge to edge, which might be too long, and misses weak lines that cover only half of the image or less. In many cases, this technique recovers the groundtruth surprisingly well.

The input for the transform is the color probability map, with a threshold applied. The product is a view on the image in Hough space, where every point (pixel) represents a line in the original image space. The intensity of the pixels corresponds to the prevalence of active pixels along certain lines in image space. Roads appear as noisy blobs, as can be seen in the leftmost column of figure 4. The extension

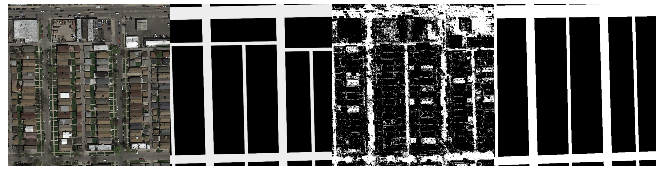

Figure 3. Hough-based road detection applied to one satellite image. From left to right: The original satellite image, the corresponding groundtruth image, the output of color probabilities with a threshold applied, and the output of the detector.

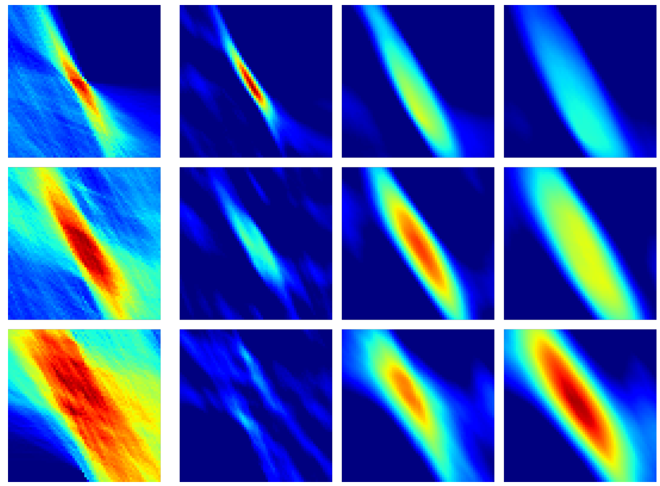

Figure 4. Accumulator space, before and after convolution. The leftmost column is the output of the Hough transform, columns 2-4 have convolution filters applied that are tuned for roads that are 12, 32, and 48 pixels wide, respectively. Top to bottom are actual occurences of roads that are approximately 12, 24, and 48 pixels wide.

along the y-axis (distance/translation) is determined by the width of the road.

In order to deal with the noisy output, and to detect roads of certain widths, three separate convolutions along the distance axis are applied. The convolutions use rectangular filters that are tuned to respond well to lines that are 12, 32 and 48 pixels wide, respectively.

The resulting images are then searched for maxima. Non-maximum suppression is applied, such as to only extract roads that do not overlap in very low angles. The extraction of roads is terminated after reaching a minimum threshold.

C. One Layer Model: Leaf Model

For each patch we apply a histogram, the mean amplitude, the variance and the local energy (sum of squared amplitudes) for all of the three color channels. Additionally we use the result of the color probability estimator. Since its result is per pixel, it's easy to split the resulting image into patches and analyze it with a histogram and image moments. [9] [10]

In order to capture texture and structure we use HOG and LBP per neighborhood (5 times 5 patches) based on gray level images as features. The Histogram of Oriented

Gradients (HOG) contains a lot of information about contour, silhouette and the overall structure of an object. Dalal et al. [11] use it for example for Human Detection. HOG can be calculated by placing a grid over an image and calculating the gradients for each pixel. Hence each pixel gets a magnitude and an orientation. The magnitude can be used to build a histogram for each cell of the grid. The HOG feature is the concatenation of all the histograms. Wang et al. [12] showed that LBP can considerably improve the detection performance on some datasets. We try to exploit that fact. In order to generate LBP one has to compare a pixel with all of its 8 neighbors. If the pixels value is greater than its neighbors' the result is 0, and 1 otherwise. Further information can be found in the papers from Ahonen et al. [13] and Ojala et al. [14] Since we want rotation invariance we don't produce histograms for each cell of a grid but instead only generate 3 histograms from the LBP result. One for the patch in the center, one for the 3 by 3 patches wide neighborhood and one for the 5 by 5 patches wide neighborhood.

All those features are fed into a Gradient Boosting classifier. In our experience Gradient Boosting can perform well on small datasets. Furthermore it is less prone to overfitting than other boosting methods. This Algorithm tries to build a strong learner by combining many weak learners. The difference to other ensemble methods is that it uses a form of gradient descent. The score gets improved iteratively by estimating the residuum with a weak learner. More information about it can be found in the paper of Jerome H. Friedman [15]. Hastie et al. show in their book Elements of Statistical Learning [15] that regularization via shrinkage improves performance considerably. In combination with shrinkage, reducing the variance via bagging can produce a more accurate model. Subsampling without shrinkage usually does poorly. Increasing the number of estimators can improve the detection rate. On the other hand overfitting becomes more likely too. For this reason we chose 1100 estimators with a learning rate of 0.1 and a subsampling rate of 0.5

Unfortunately HOG is very sensitive to varying orientations. Although we could improve our results it is mostly a form of overfitting towards a certain angle. The training set mostly consists of horizontal and vertical streets, thus the classifier finds them easily while diagonals are almost completely missed. We regularize with data augmentation [5]. Due to the limited size of the training set, we artificially increase the number of training images by adjusting the brightness and by rotating the images. This decreases the variance.

D. *Two-Layer model: Deep Forest*

Although smoothening the shape of the predicted streets and removing outliers can marginally improve the predictions, the results were not convincing enough. Attempts to exploit the fact that most streets are straight by adding the result of the Hough streets detector as an additional feature to the Leaf Model weren't successful. Since it contained too much information in comparison with the other features it began to dominate the prediction. As a result the prediction yielded a similar output as Hough itself. In order to use Hough we require a second comparably strong indicator that can counteract its dominance.

For this reason we implemented a two-layer model. The basic idea is to feed both the output of Hough and the output of the leaf model into a classifier. The label of the patch and the ones of the patches in the neighborhood are used as features. Hence the distribution of positive and negative patches gives valuable insights into the overall structure of an area. If many neighboring patches yield street then it's far more likely that the current patch should yield street too. Therefore holes can be filled and noise can be removed.

Inspired by the work of Dieleman et al. [16] [17] we decided to get two predictions from the leave model, one for the image to predict and one for a by 45 degree rotated version of the same image. Both are analyzed using the features mentioned above and fed into the classifier. This allows the model to use the same feature extraction pipeline to analyze the input from different angles. In other words the leaf model can look at the image in the orientation it prefers. Thus the weakness introduced by HOG can be overcome without loosing its strength.

Since we don't want to rely only on the outputs of the first layer, we also include the rotation invariant features LBP, CP and the histogram that the first layer computed. In the next step we feed all those features into a random forest classifier which produces the final output. Random Forests [18] basically work by taking random sets of features as input, using them to build many weak learners such as decision trees and averaging the results. This leads to robust predictions with respect to noise. Therefore having many estimators can increase the stability of the model but it also slows down the execution. In our case we decided to use 2600 trees.

In order to train our model we need to train the leaf model first. Afterwards we get the predictions of the first layer for all the training data and train the classifier of the second layer with it. Such a procedure heavily suffers from over fitting. The leaf model is too sensitive to images that it has seen before. Hence all the predictions that the second layer gets from the first layer are almost perfect for all the training examples. In order to fight this problem the leave model is only trained on the images 1 to 60 in the training set. Furthermore we exploit the fact that HOG favors certain orientations by only teaching it a by 45 degree rotated version of the examples 40 to 100. Rotating images also causes a shift of the patches. The second layer is trained on the original training set without any data augmentation.

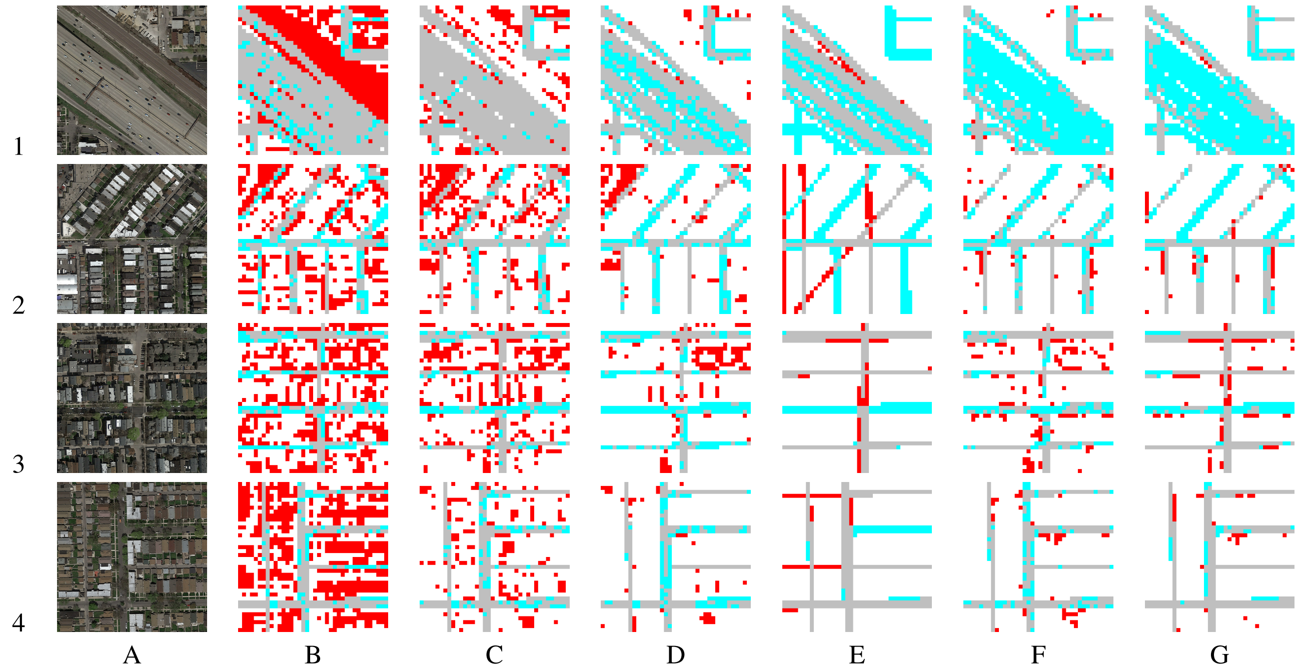

Figure 5. A: Satellite Images, B: Log.Reg. Baseline, C: CNN Baseline, D: Color Probability, E: Hough-based Detector, F: Leaf Model, G: Final Prediction. – Red patches are false positives, blue patches are false negatives.

III. Results

Figure 5 shows an overview of the final results. Our solution has limitations; it performs suboptimally in three specific cases that appear infrequently. The first case is shown in row 1, where a wide highway is missed almost entirely, even though color probability and Hough do detect it at least partially. The second case is shown in row 2, where the evaluated satellite image consists of short diagonal roads that are mostly covered with trees. None of our techniques are able to detect these roads. The third case is shown in row 3, where a horizontal road in the center of the image has an usually dark shade. Such roads are not detected at all. The fourth row shows the typical case that covers most data, where our solution performs reasonably well. It clearly outperforms the logistic regression baseline (column A). It also outperforms the convolutional neural network baseline (column B), which detects roads generally well but suffers from noise significantly. The last row also showcases the architecture, where the Two-Layer Model (column G) combines the predictions from color probability (column D), Hough-based detection (colum E) and the Leaf Model (column F) into the final result (column G) that has higher prediction accuracy than any of its inputs have.

IV. Discussion

Each of the four major building blocks of our algorithm contributes significantly to the final result. The color probability estimation takes advantage of different colors between roads and background. The Hough transform exploits the fact that roads are usually straight and that it is unlikely to catch an endpoint of a road in a processed image. Its generation is efficient and contains a lot of information about the road map. The possibilities to use it as a feature or as a regularizer might be worthy to explore.

The first layer draws conclusions from the available features and makes a good initial prediction. The second layers visibly improves the output of the first one by efficiently removing noise and gaps. This shows that analyzing the output with a second classifier can indeed be a good idea even though the training is time consuming and one has to take care of overfitting.

Although the algorithm overall produces satisfactory results, it has some weaknesses as shown in the result section. For example distinguishing highways from parking lots poses a problem, since both areas have too many similarities. The problem could probably be resolved if the images were analyzed on different scales in order to capture the shape of highways better.

V. Summary

In most cases our algorithm performs well and gives a good solution for the road segmentation problem. Its main advantages over other approaches prevalent in image segmentation include a satisfactory performance even if trained with a relatively small amount of training data.

References

[1] J. Porway, K. Wang, B. Yao, and S. C. Zhu, “A hierarchical and contextual model for aerial image understanding,” in

Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008, pp. 1–8.

[2] M. W. Koch, M. M. Moya, J. G. Chow, J. Goold, and R. Malinas, "Road segmentation using multipass single-pol synthetic aperture radar imagery," in *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2015, pp. 151–160.

[3] J. Yuan and A. M. Cheriyadat, "Road segmentation in aerial images by exploiting road vector data," in *Computing for Geospatial Research and Application (COM.Geo), 2013 Fourth International Conference on*, 2013, pp. 16–23.

[4] R. Pteri and T. Ranchin, "Multiresolution snakes for urban road extraction from ikonos and quickbird images," in *in 23nd EARSeL Annual Symposium Remote Sensing in Transition*, 2003, pp. 2–5.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[6] "Symbolic image for hog," http://doi.ieeecomputersociety.org/cms/Computer.org/dl/trans/tp/2010/07/figures/ttp20100712397.gif, accessed: 2016-06-30.

[7] "Symbolic image for lbp," https://en.wikipedia.org/wiki/Local_binary_patterns#/media/File:Lbp_neighbors.svg, accessed: 2016-06-30, By Xiawi - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=11743214.

[8] P. Hough, "Machine analysis of bubble chamber pictures," *Proc. Int. Conf. High Energy Accelerators and Instrumentation*, 1959.

[9] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE transactions on information theory*, vol. 8, no. 2, pp. 179–187, 1962.

[10] J. Flusser and T. Suk, "Rotation moment invariants for recognition of symmetric objects," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3784–3790, 2006.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1. IEEE, 2005, pp. 886–893.

[12] X. Wang, T. X. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 32–39.

[13] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[14] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *European conference on computer vision*. Springer, 2004, pp. 469–481.

[15] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics Springer, Berlin, 2001, vol. 1.

[16] S. Dieleman, K. W. Willett, and J. Dambre, "Rotation-invariant convolutional neural networks for galaxy morphology prediction," *Monthly notices of the royal astronomical society*, vol. 450, no. 2, pp. 1441–1459, 2015.

[17] S. Dieleman, J. De Fauw, and K. Kavukcuoglu, "Exploiting cyclic symmetry in convolutional neural networks," *arXiv preprint arXiv:1602.02660*, 2016.

[18] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Road Segmentation with a Two-Layer Model and the Hough Transform

Authored by (in block letters):
For papers written by groups the names of all authors are required.

Name(s):	First name(s):
Kipfer	Kevin
Erdin	Matthias
Dykcik	Lukasz

With my signature I confirm that
- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	Signature(s)
Zurich, 30.06.2016	K. Kipfer M. Erdin [illegible]

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.