SANDIA REPORT

SAND2017-2423
Unlimited Release
Printed April 2017

Scalable Track Detection in SAR CCD Images

Tu-Thach Quach, James G. Chow

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@adonis.osti.gov
Online ordering:	http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.fedworld.gov
Online ordering:	http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2017-2423
Unlimited Release
Printed April 2017

Scalable Track Detection in SAR CCD Images

Tu-Thach Quach
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-1163

James G. Chow
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-1202

Abstract

Existing methods to detect vehicle tracks in coherent change detection images, a product of combining two synthetic aperture radar images taken at different times of the same scene, rely on simple, fast models to label track pixels. These models, however, are often too simple to capture natural track features such as continuity and parallelism. We present a simple convolutional network architecture consisting of a series of 3-by-3 convolutions to detect tracks. The network is trained end-to-end to learn natural track features entirely from data. The network is computationally efficient and improves the F-score on a standard dataset to 0.988, up from 0.907 obtained by the current state-of-the-art method.

Contents

Figures

Tables

1 Introduction

Multiple synthetic aperture radar (SAR) images taken at different times of the same scene can be combined to produce coherent change detection (CCD) images that can reveal subtle surface changes such as those made by tire tracks [1, 2]. Our goal is to segment the vehicle tracks in these images.

Vehicle track segmentation can be viewed as a binary labeling problem where a pixel is labeled with a 1 if it belongs to a track and 0 otherwise. A simple approach is to train a classifier on features extracted at each pixel. While this approach is simple and can produce good results, it fails to complete disconnected tracks that are caused by sensor noise and various environmental effects, such as vegetation and weather, that are prevalent in CCD images.

A common approach to address this discontinuity problem in image segmentation is to apply a pairwise Markov random field on adjacent pixels. In particular, pairwise Potts models are very popular, partly due to their simplicity. More importantly, these models allow for efficient inference via graph cuts [3]. A recent vehicle track segmentation approach uses a pairwise Potts model with some success [4]. A major problem with these pairwise models, however, is that they favor short, compact objects. Vehicle tracks, on the other hand, are long, thin objects.

Several higher-order models for image segmentation have been proposed to address the shortcomings of the simple pairwise model. In particular, the cooperative cut approach introduces potential functions that operate on subsets of edges in the graph [5]. Inferencing with cooperative cut is, in general, NP-hard. For a certain class of cooperative-cut potential functions, exact inference is still possible via graph cuts [6]. A recent vehicle track segmentation approach imposes higher-order constraints through the use of a constrained Delaunay triangulation (CDT) to discover and complete missing pieces of tracks [7].

These models, however, still cannot capture important properties of natural tracks. Specifically, it is not clear how track curvatures and their parallel nature can be enforced in these models. This work presents an approach to vehicle track segmentation that captures the properties of natural tracks. We use a simple convolutional network that consists of a series of 3-by-3 convolutions to label track pixels. Unlike standard convolutional networks, max pooling is not required, which destroys too much information for accurate dense pixel labeling problems. Instead, we utilize dilated convolutions to increase the receptive field of the network without sacrificing resolution. Using a 6-layer network, we improve the F-score on a standard vehicle track data set to 0.988, up from 0.907 obtained by the CDT method.

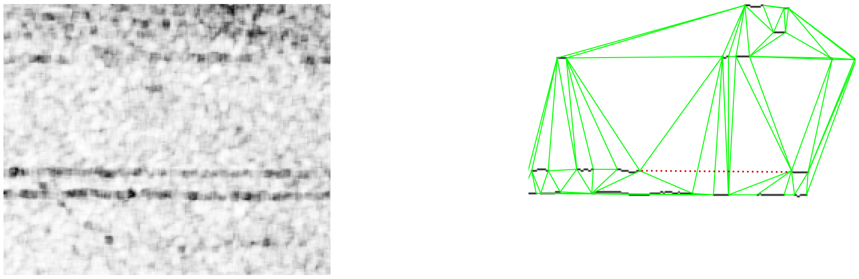
The details of our approach are presented in Section 3. Experimental results are presented in Section 4. Concluding thoughts are provided in Section 5.

2 Related Work

Unlike optical images, SAR images are noisy and unimodal [8], making the segmentation task more difficult. Despite these challenges, several track segmentation algorithms have been proposed. Given a search cue, e.g., starting location of a path, a single track pair can be extracted by tracing parallel lines (tire tracks) starting at the search cue [9]. The proposed method can find a single track provided that the user supplied the initial search cue. A related method uses cubic spline fitting to extract vehicle tracks [10]. It is limited to a single non-overlapping track. In practice, a scene can have an arbitrary number of tracks, including no tracks. In addition, these tracks may come from different types of vehicles. Any automatic technique must be able to account for these conditions.

A recent method addresses some of these limitations by finding the simplest set of tracks that explains the observed data [11]. It is a greedy method that finds one track at a time, until the objective function can no longer decrease. The obtained tracks, which are line segments, are iteratively merged to form full tracks. The merging step is important as the initially obtained tracks may not be complete and the merging procedure allows the algorithm to discover missing parts of tracks. It is, however, also computationally expensive as it considers every possible pairwise merges recursively until convergence. Furthermore, the approach uses a parallel track template to find candidate tracks. It is unclear how that approach can be generalized to find single tracks, such as those made by motorcycles, or parallel tracks made by various vehicle sizes.

One closely related work is the pairwise Potts approach that uses the radial derivative of the local Radon transform centered on each pixel as features [4]. The shortcoming of the simple pairwise Potts model is a major limitation of this approach. In order to take into account the long, thin characteristic of tracks, a recent work relies on the CDT, first used for contour detection in natural images [12], to find missing pieces of tracks to complete them [7]. The approach first identifies initial high-quality track segments using the ridge feature [13]. The remaining segments are discovered using the CDT where the initially identified segments form the constrained edges. A binary pairwise Potts model is then constructed on the edges of the CDT to infer salient tracks. Although the approach still uses a pairwise Potts model, the costs are associated with the edges (not image pixels) of the CDT, allowing it to overcome the short-boundary problem associated with pairwise Potts models based on adjacent pixels. A problem with the CDT approach is that it is sensitive to the initial constrained edges. As a consequence, the obtained triangulation may miss some completion edges as shown in Figure 1.



(a)

(b)

Figure 1: The CDT is dependent on the constrained edges and may not discover all completion edges: (a) input track image and (b) obtained CDT. Completion edges are in green. A large portion of the top track (red dots) is not discovered by the CDT. Best viewed in color.

3 Track Segmentation

Let x be an input image and z be the corresponding ground-truth binary image where $z(i) = 1$ if pixel i is a track pixel and $z(i) = 0$, otherwise. Our goal is to build and train a convolutional network that can predict z given x. It is possible to cast the track detection problem into an object detection framework where a bounding box is placed around a detected object. The result, however, might not be meaningful and can also be difficult to accomplish. Several of the reasons are already mentioned. In particular, tracks are elongated objects that do not have natural boundaries, e.g., a track may span an entire image. A bounding box around such irregular objects is not very useful. As a result, we cast the track detection problem into a dense pixel labeling, or binary segmentation, problem.

Our network is illustrated in Figure 2. As with many standard convolutional networks, the input image is passed through a series of convolutions. Each convolution is followed by ReLU [14]. Unlike standard networks, we do not utilize pooling or down-sampling layers, which are often used to exponentially increase the receptive field of view of the network and, at the same time, decrease the computations in latter layers. The use of pooling layers, however, discards information that might be pertinent for track detection. In addition, down sampling may decrease the resolution of the output image and an upsampling process is needed to recover the original size. As a consequence, we provide an alternative to max pooling in our network.

In order to obtain the same benefit of down sampling, e.g., increased field of view, without sacrificing resolution, we use dilated convolutions [15]. Dilated convolution is similar to standard convolution, but the filters can be dilated or upsampled by inserting zeros between coefficients. Dilated convolution with a rate of 1 is the standard convolution. In general, with a rate of r, there are $r - 1$ zeros inserted between coefficients. We use 3-by-3 dilated convolutions in all layers. The dilation rate, however, increases so that the receptive field becomes larger for latter layers. Note that increasing the dilation rate does not increase computation because the effective kernel size is still 3-by-3.

Our network produces several side-output predictions, one after each convolutional layer. These side-output predictions are then used in conjunction with the ground-truth image to form the objective function. The use of side-output predictions is inspired by deep supervised networks (DSN) [16]. The purpose of DSN is to improve the effectiveness of training the hidden layers by informing them of the objective function directly, rather than relying on the final layer to propagate the information back.

Let $z^+ = \{i : z(i) = 1\}$ and $z^- = \{i : z(i) = 0\}$ be the set of track and non-track pixels, respectively. Let $\alpha = |z^+|/(|z^+| + |z^-|)$ and $\beta = 1 - \alpha$. Let y_j be the side-output activation of layer j. The objective function of this layer is the class-balanced cross-entropy loss:

$$\mathcal{L}_j = -\beta \sum_{i \in z^+} \log \sigma\left(y_j(i)\right) - \alpha \sum_{i \in z^-} \log\left(1 - \sigma(y_j(i))\right), \tag{1}$$

where σ is the sigmoid function. The class-balanced objective function is important because

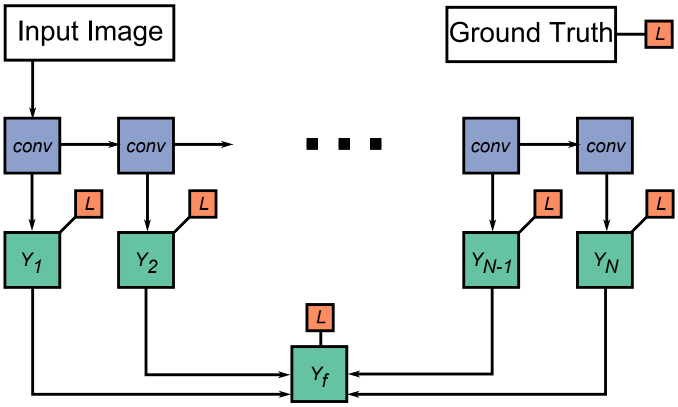


Figure 2: The convolutional network for track segmentation. The input image is passed through a series of dilated convolutions (the c boxes). The network produces a side output (the y boxes) after each convolutional layer. All side-output predictions use the same ground-truth image to compute side losses, which are aggregated to form the total loss.

there are far fewer track pixels than non-track pixels. Without proper balancing, the network will favor non-track pixels over track pixels.

In addition to the side-output layers, the network also has a weighted multi-scale context fusion layer, $y_f = \sum_j w_j y_j$, that linearly combines the activations of previous layers according to weight w_j for layer j. The fusion weights are essentially 1-by-1 convolution. The loss function of this fusion layer is also the above class-balanced cross-entropy loss. The final objective function is the sum of all the side-output losses and the fusion loss.

4 Experimental Results

In the following experiments, we use a 6-layer track detection network. The details of the network parameters are summarized in Table 1. Due to the increased dilation rate, the receptive field of the last layer is relatively large at 127 pixels, allowing the network to connect and form contiguous tracks.

We train the network using a mixture of real and simulated images. The 60 real CCD images are 512-by-512 in size and contain vehicle tracks with hand-labeled ground truth. The simulated, openly available dataset consists of 40 CCD images containing simulated vehicle tracks along with ground truth [17]. Each 600-by-800 CCD image is generated from a real SAR image pair and contains a single simulated tire track. We randomly select 30 images to train the network. The remaining 10 images are used to quantify the performance of the network. In summary, we have 90 training images and 10 test images.

We train the network for 12000 iterations using stochastic gradient descent. The initial learning rate is 1e-3 and is reduced by a factor of 10 after every 4000 iterations. The mini-batch size is a single image and the weight decay is 1e-4. To enrich our small training set, we randomly flip each image horizontally and vertically. In addition, we also add a small Gaussian noise, $\mathcal{N}(0, 1)$, to each image. Training takes 2.5 hours on a single K40 GPU.

The performance of the network is evaluated using the precision-recall framework and the F-score (maximum of $2 \cdot precision \cdot recall / (precision + recall)$). This is accomplished by thresholding the output image to obtain a binary image. We then morphologically thinned the binary image to produce a single-pixel wide track image. A ground-truth track pixel is correctly detected if and only if there is a predicted track pixel that is within a Euclidean distance of 3 pixels from it (true positive). Any predicted track pixel that does not have a corresponding ground-truth pixel is considered a false positive. A buffer of 3 pixels is used because ground-truth track pixels are actually several pixels wide.

The precision-recall curves of various output layers are shown in Figure 3. For comparison, we also include the result using the CDT method [7], which is just a single point on the plot as it is a hard-output track detector. The network substantially outperforms the CDT method. By incorporating output from multiple layers, the fusion layer achieves the best F-score of 0.988, slightly better than layer 6.

We observe that earlier layers tend to identify primitive track features, while latter layers can see coarser, global features that are more relevant to tracks. In particular, latter layers can bridge gaps, forming more contiguous tracks. This can be seen in the examples shown in Figure 4 of real vehicle tracks. In terms of running time, with a 2000-by-2000 image, the network takes 2.5 seconds. We use Tensorflow to implement the network.

Table 1: Summary of 6-layer track detection network.

Layer	Dilation Rate	Output Channels	Receptive Field
1	1	32	3
2	2	32	7
3	4	64	15
4	8	64	31
5	16	128	63
6	32	128	127

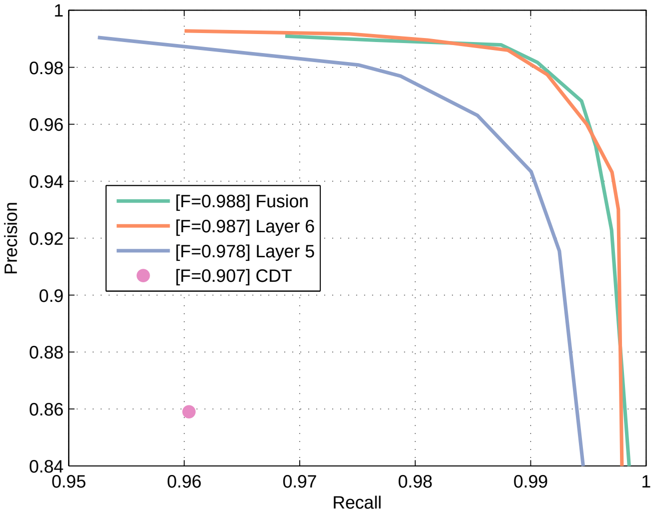


Figure 3: Precision-recall curves of a 6-layer network at various output layers ranked by their F-score (number in square bracket). For comparison, the results using the CDT method is also included. The network improves substantially over the state-of-the-art CDT method.

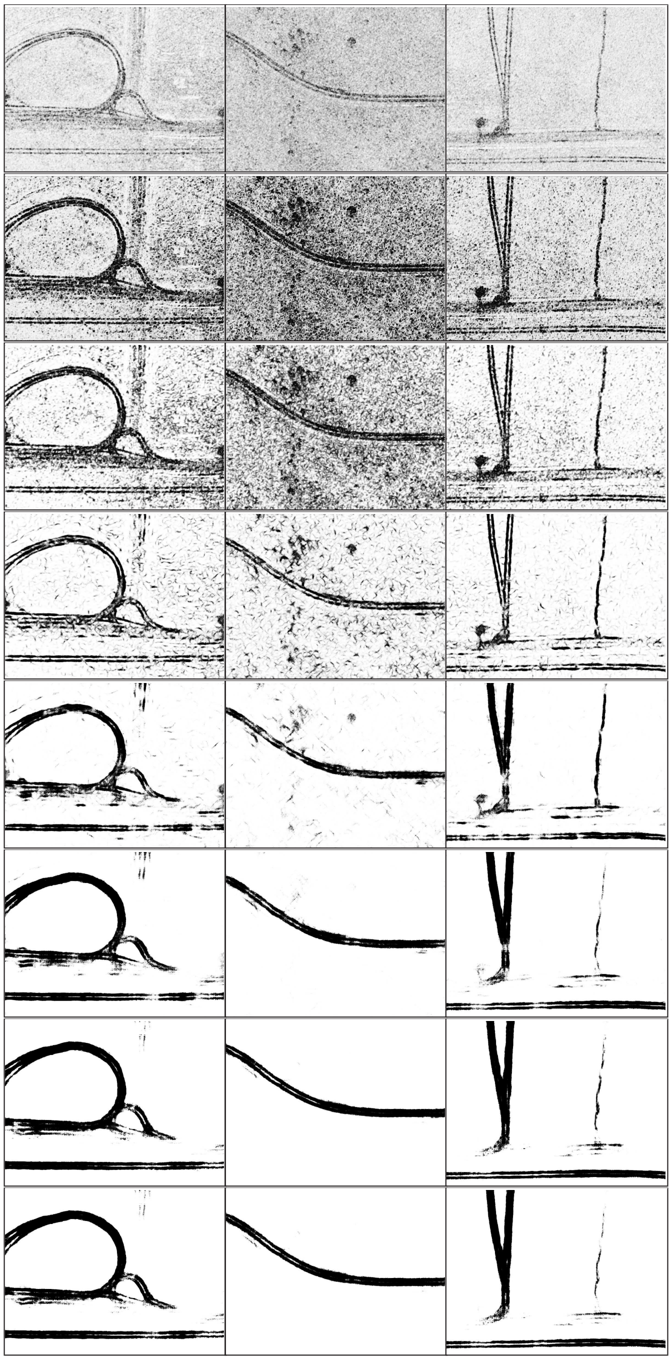


Figure 4: Example results on real SAR CCD images from the UUR dataset. The top row is the input image and the subsequent rows correspond to the output of different layers in ascending order. The last row is the output of the fusion layer. With larger receptive fields, the latter layers are able to form contiguous tracks more definitively than earlier layers, which tend to identify only primitive track characteristics.

5 Discussion and Conclusion

We have presented an approach to segment vehicle tracks in SAR CCD images. The approach uses a simple convolutional network that captures track features at various scales through the use of dilated convolutions. The resulting network improves substantially over the current state-of-the-art CDT method. This improvement comes from the network's ability to complete tracks in the latter layers, as those have large receptive fields.

The design of our network is flexible, allowing for an arbitrary number of layers. The results also suggest that latter layers tend to do better than earlier layers. This naturally leads to the possibility of improving the network's performance by adding additional layers. This does come at an increase in computational cost. This cost can be substantial for networks that do not use max pooling, as the sizes of the latter layers do not decrease. In preliminary experiments, we find that adding one or two more layers only improves the accuracy marginally, but the increase in computational cost is more noticeable. Investigating this trade-off and extending our network to detect other types of tracks are topics for future research.

References

[1] D. G. Corr and A. Rodrigues, "Coherent change detection of vehicle movements," in *Geoscience and Remote Sensing Symposium.* IEEE, 1998, pp. 2451–2453.

[2] C. V. Jakowatz, D. E. Wahl, P. H. Wahl, D. C. Ghiglia, and P. A. Thompson, *Spotlight-mode synthetic aperture radar: a signal processing approach.* Boston, MA: Kluwer Academic Publishers, 1996.

[3] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, 2004.

[4] R. Malinas, T.-T. Quach, and M. W. Koch, "Vehicle track detection in CCD imagery via conditional random field," in *Asilomar Conference on Signals, Systems, and Computers.* IEEE, 2015.

[5] S. Jegelka and J. Bilmes, "Submodularity beyond submodular energies: Coupling edges in graph cuts," in *CVPR.* IEEE, 2011, pp. 1897–1904.

[6] P. Kohli, A. Osokin, and S. Jegelka, "A principled deep random field model for image segmentation," in *CVPR.* IEEE, 2013, pp. 1971–1978.

[7] T.-T. Quach, R. Malinas, and M. W. Koch, "Low-level track finding and completion using random fields," in *Image Processing: Machine Vision Applications IX.* IS&T, 2016.

[8] J.-S. Lee and I. Jurkevich, "Segmentation of SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 27, no. 6, pp. 674–680, 1989.

[9] M. Cha and R. Phillips, "Automatic track tracing in SAR CCD images using search cues," in *Asilomar Conference on Signals, Systems and Computers.* IEEE, 2012, pp. 1825–1829.

[10] M. Cha, R. Phillips, and M. Yee, "Finding curves in SAR CCD images," in *ICASSP.* IEEE, 2011, pp. 2024–2027.

[11] T.-T. Quach, R. Malinas, and M. W. Koch, "A model-based approach to finding tracks in SAR CCD images," in *CVPR Workshops.* IEEE, 2015, pp. 41–47.

[12] X. Ren, C. C. Fowlkes, and J. Malik, "Scale-invariant contour completion using conditional random fields," in *ICCV.* IEEE, 2005, pp. 1214–1221.

[13] T. Lindeberg, "Edge detection and ridge detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 117–154, 1998.

[14] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *International Conference on Machine Learning*, 2010.

[15] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *International Conference on Learning Representations*, 2016.

[16] C.-Y. Lee, S. Xie, P. W. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *AISTATS*, 2015.

[17] W. J. Bow Jr., "Sandia SAR data collect 2006," Sandia National Laboratories, Tech. Rep. SAND2006-2290P, 2006.

DISTRIBUTION:

	MS 1163	W. Bow, 5448
	MS 0532	S. Castillo, 5340
	MS 1202	J. Chow, 5349
	MS 0519	M. Lewis, 5349
	MS 1163	T.-T. Quach, 5448
	MS 1173	A. Roesler, 5440
1	MS 0359	D. Chavez, LDRD Office, 1911
1	MS 0899	Technical Library, 9536 (electronic copy)

Sandia National Laboratories