Sketch Tokens: A Learned Mid-level Representation for Contour and Object Detection

Joseph J. Lim
Massachusetts Inst. of Technology
lim@csail.mit.edu

C. Lawrence Zitnick
Microsoft Research
larryz@microsoft.com

Piotr Dollár
Microsoft Research
pdollar@microsoft.com

Abstract

We propose a novel approach to both learning and detecting local contour-based representations for mid-level features. Our features, called ***sketch tokens****, are learned using supervised mid-level information in the form of hand drawn contours in images. Patches of human generated contours are clustered to form sketch token classes and a random forest classifier is used for efficient detection in novel images. We demonstrate our approach on both top-down and bottom-up tasks. We show state-of-the-art results on the top-down task of contour detection while being over 200× faster than competing methods. We also achieve large improvements in detection accuracy for the bottom-up tasks of pedestrian and object detection as measured on INRIA [2] and PASCAL [4], respectively. These gains are due to the complementary information provided by sketch tokens to low-level features such as gradient histograms.*

1. Introduction

For visual recognition, mid-level features provide a bridge between low-level pixel-based information and high-level concepts, such as object and scene level information. Effective mid-level representations abstract low-level pixel information useful for later classification while being robust to irrelevant and noisy signals.

In this work, we propose a novel approach to both learning and detecting local edge-based mid-level features, and demonstrate their effectiveness for both bottom-up and top-down tasks. Our features, called *sketch tokens*, capture local edge structure. The classes of sketch tokens range from standard shapes such as straight lines and junctions to richer structures such as curves and sets of parallel lines (Fig. 1).

Given the vast number of potential local edge structures, we must select an informative subset to represent by the sketch tokens. We propose a novel approach to defining token classes using *supervised mid-level* information. The supervised mid-level information is obtained from human labeled edges in natural images [1]. Patches centered on contours are extracted from the hand drawn sketches and clustered to form a set of token classes. This results in a diverse, representative set of sketch tokens (Fig. 1).

Our goal is to efficiently predict the occurrence of sketch tokens given an input color image. We propose a data driven approach that classifies each image patch with a token label given a collection of low-level features. We solve this large multi-class problem using random decision forests. For contour detection, we show state-of-the-art results, while boosting efficiency over 200 fold using standard datasets [1]. Results on the INRIA pedestrian dataset [2] show a large reduction in error rate over previous state-of-the-arts. We also show our mid-level features are complementary to the Histogram of Oriented Gradients descriptor [2] on the challenging PASCAL object detection dataset [4].

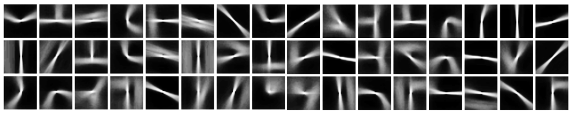


Figure 1. Examples of sketch tokens learned from hand drawn sketches represented using their mean contour structure. Notice the variety and richness of the sketch tokens.

2. Sketch Tokens

2.1. Defining sketch token classes

Our goal is to define a set of token classes that represent the wide variety of local edge structures in an image. These include straight lines, t-junctions, y-junctions, corners, curves, parallel lines, etc. We propose a method for discovering these classes using human-labeled sketches [1].

Let us assume we have a set of images with a corresponding set of binary images S representing the hand drawn contours. We define the set of sketch token classes by clustering patches s extracted from S. Example cluster means are illustrated in Figure 1. Notice the variety of the sketch tokens, ranging from straight lines to more complex structures.

2.2. Detecting sketch tokens

Given a set of sketch token classes, we wish to detect their occurrence in color images. As input, features are computed from color patches x extracted from the training images I. Ground truth class labels are supplied by the clustering results described above.

Feature extraction: For feature extraction, we use an approach inspired by Dollár *et al.* [3] and compute multiple feature channels per image. Two types of features are then employed: features directly indexing into the channels and self-similarity features.

Our channels are composed of color, gradient, and oriented gradient information in a patch extracted from a color

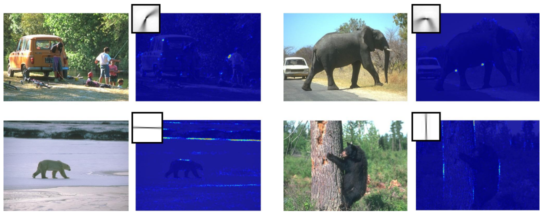


Figure 2. Illustration of the sketch token responses for four tokens. Notice the high selectivity of each sketch token (best in color.)

Method	ODS	OIS	AP	Speed
Human	.80	.80	-	-
Canny	.60	.64	.58	1/15 s
gPb (local) [1]	.71	.74	.65	60 s
SCG (local) [7]	.72	.74	.75	100 s
Sketch tokens	**.73**	**.75**	**.78**	**1 s**
gPb (global) [1]	.73	.76	.73	240 s
SCG (global) [7]	.74	.76	.77	280 s

Table 1. Contour detection result on BSDS500: We achieve state-of-the-art results among all local methods. The methods shown in the last two rows perform complex global resulting in slightly better performance. However, our approach is 240-280x faster.

image. Three color channels are computed using the CIE-LUV color space. We compute several normalized gradient channels that vary in orientation and scale [3, 2].

The second type of feature used by our method is based on self-similarity. The self-similarity features capture the portions of an image patch that contain similar textures based on color or gradient information. More details on how we capture the self-similarity can be found in [6].

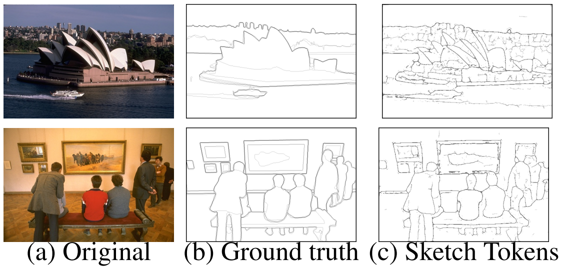
Classification: Two considerations must be taken into account when choosing a classifier for labeling sketch tokens in image patches. First, every pixel in the image must be labeled, so the classifier must be efficient. Second, the number of potential classes for each patch ranges in the hundreds. In this work we use a random forest classifier, since it is an efficient method for multi-class problems.

3. Experimental Results

Contour detection results: Sketch tokens provide an estimate of the local edge structure in a patch. The details on how to compute the binary labeling of pixel contours from mid-level sketch tokens can be found in [6].

We test our contour detector on the BSDS500 [1]. In Table 1, we compare our method against competing methods. Our method achieves state-of-the-art results among all local methods, and achieves nearly the accuracy of global approaches. Qualitative results are shown in Figure 3. Our detector processes a 480 × 320 image in under 1 second. This is over 200× more efficient than approaches.

INRIA pedestrian: For pedestrian detection we use an improved implementation of Dollár *et al.* [3] . We add channels corresponding to our sketch token probability maps. Results are shown in Table 2 . The baseline approach of [3] uses 10 channel features (LUV+M+O) and achieves a log-average miss rate (MR) of 17.2%. Our approach using 150 sketch tokens achieves a MR of 19.5%. Combining sketch tokens and the 10 low-level features achieves 14.7%.



(a) Original (b) Ground truth (c) Sketch Tokens

Figure 3. Examples of contour detection on the BSDS500 [1]. Note how our method captures finer details such as the structure of Opera House on the 1st row and human legs on the 2nd row. See [6] for more results.

channels	# channels	miss rate
LUV+M+O	10	17.2%
ST	151	19.5%
ST+LUV+M+O	**161**	**14.7%**

Table 2. Accuracy of [3] combined with sketch tokens on INRIA [2] with varying choice of channels.

	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow
HOG	**27.9**	56.5	1.9	6.2	21.2	48.2	**52.7**	**7.6**	17.7	21.2
ST+HOG	23.8	**58.2**	**10.5**	**8.5**	**27.1**	**50.4**	52.0	7.3	**19.2**	**22.8**

	table	dog	horse	moto	person	plant	sheep	sofa	train	tv
HOG	14.7	3.0	55.4	42.9	**33.9**	6.0	11.9	21.7	43.2	37.7
ST+HOG	**18.1**	**8.0**	**55.9**	**44.8**	32.4	**13.3**	**15.9**	**22.8**	**46.2**	**44.9**

Table 3. PASCAL 2007 results for DPMs: On average Sketch Tokens+HOG outperformed HOG by 2.5 AP.

PASCAL VOC 2007: Our final set of results use the PASCAL VOC 2007 dataset [4]. We perform experiments with the deformable parts model (DPM) of Felzenszwalb *et al*. [5]. We propose adding our sketch tokens to the HOG features for training the DPMs. Results are shown in Table 3 for DPMs. In nearly all cases top Average Precision (AP) scores are achieved with a combination of HOG and sketch tokens. This demonstrates that sketch tokens may provide a complementary information over the HOG descriptor.

References

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 33, 2011. 1, 2

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 1, 2

[3] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, 2009. 1, 2

[4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, June 2010. 1, 2

[5] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9):1627–1645, 2010. 2

[6] J. Lim, L. Zitnick, and P. Dollár. Sketch tokens: A learned mid-level representation for contour and object detection. In *CVPR*, 2013. 2

[7] X. Ren and B. Liefeng. Discriminatively trained sparse code gradients for contour detection. In *NIPS*, 2012. 2