Mining Discriminative Triplets of Patches for Fine-Grained Classification

Yaming Wang[1], Jonghyun Choi[2], Vlad I. Morariu[1] and Larry S. Davis[1]
[1]University of Maryland, [2]Comcast Labs DC
{wym, jhchoi, morariu, lsd}@umiacs.umd.edu

IEEE 2016 Conference on Computer Vision and Pattern Recognition
CVPR 2016

1. Introduction

1.1. Background

- Fine-grained Classification

- Subtle differences in highly localized regions

1.2. The Problems

- Extra part/3D annotations needed for accurate discriminative region localizations
- Previous mid-level approaches are not accurate enough to localize discriminative regions automatically

1.3. Our Contribution

- **Triplet of patches**

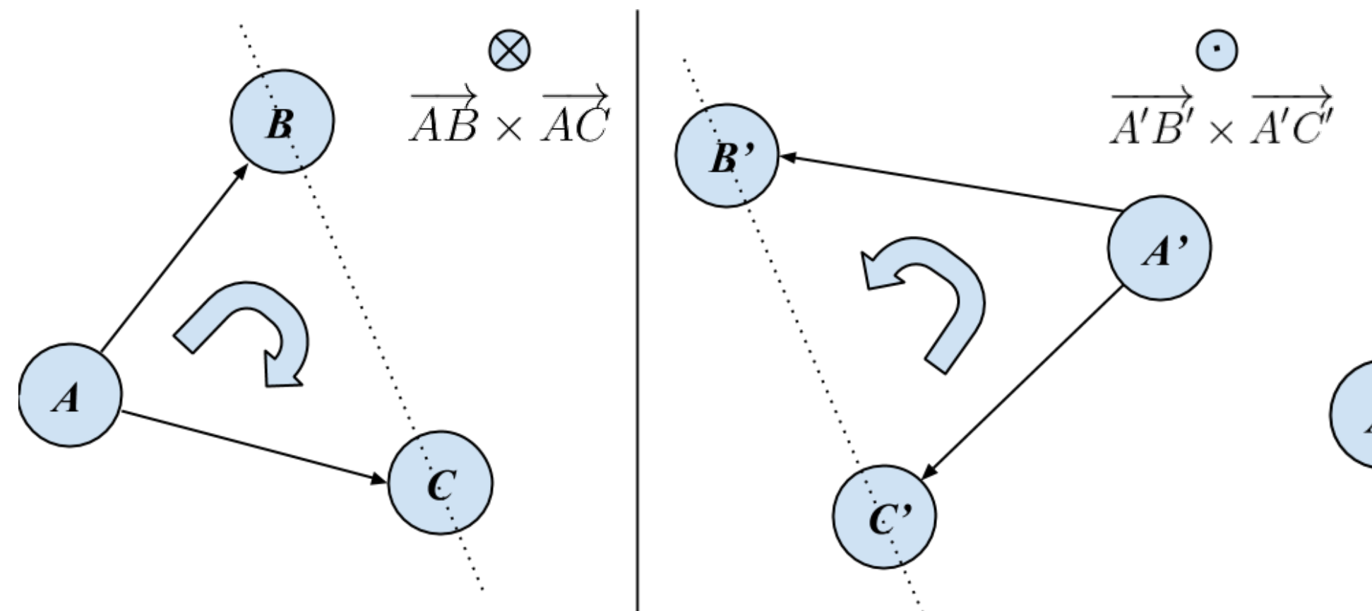
  Accurate localization without expensive annotations
- **Automatic discovery of discriminative triplets**

  Local initialization – Global mining

2. Triplet of Patches with Geometric Constraints

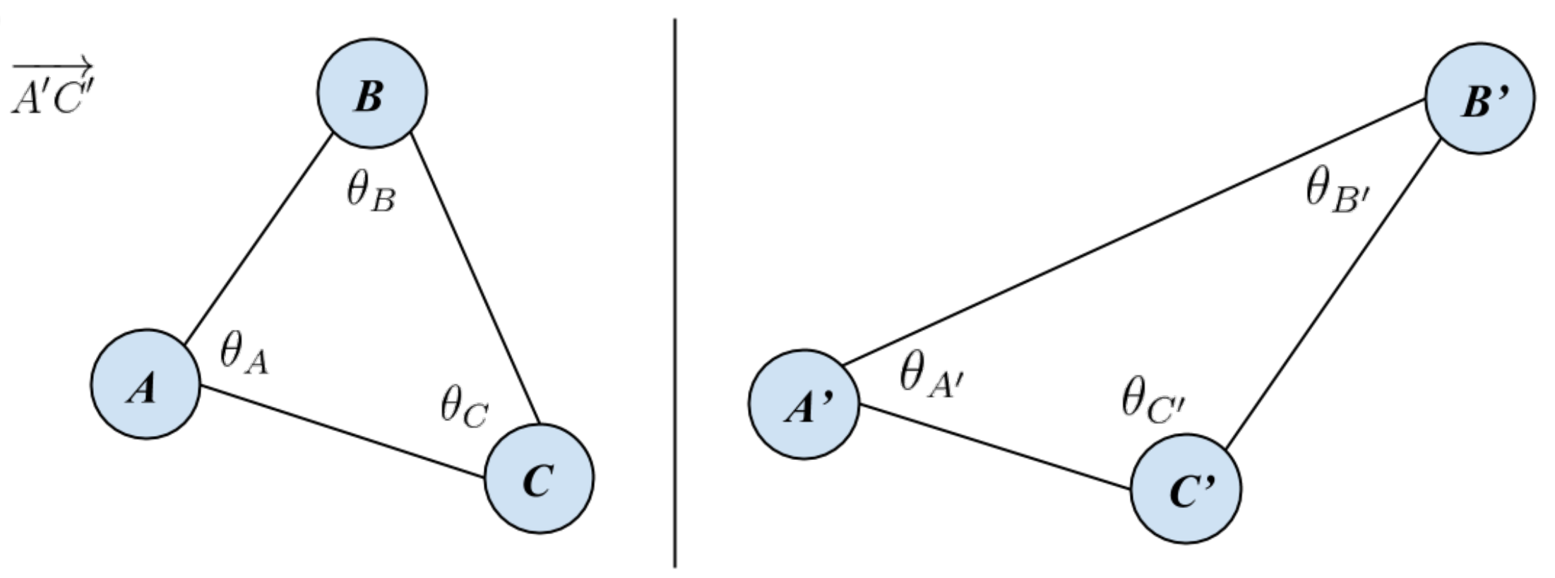
2.1. Geometric Constraints

- **Order Constraint**
- **Shape Constraint**


⊗
$\overrightarrow{AB} \times \overrightarrow{AC}$
B
A
C
⊙
$\overrightarrow{A'B'} \times \overrightarrow{A'C'}$
B'
A'
C'

$G_{ABC} = 1$ $G_{ABC} = -1$

$$p_o = 1 - \eta_o \mathbf{1}(G_{A'B'C'} \neq G_{ABC})$$



$$p_s = 1 - \eta_s \frac{\sum |\cos(\theta_i) - \cos(\theta_{i'})|}{6}$$

2.2. Triplet Detector

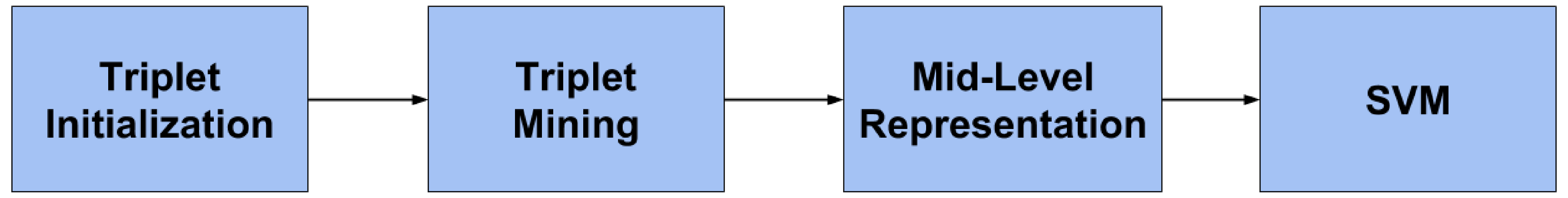
$$\{T_A, T_B, T_C\} \xrightarrow[\omega_i = \Sigma^{-1}(T_i - \mu)]{} \{\omega_A, \omega_B, \omega_C, G_{ABC}, \Theta_{ABC}\}$$

Given $\{T_{A'}, T_{B'}, T_{C'}\}$,

$$S_{A'B'C'} = \underbrace{\left(\omega_A^{\mathrm{T}} T_{A'} + \omega_B^{\mathrm{T}} T_{B'} + \omega_C^{\mathrm{T}} T_{C'}\right)}_{\textbf{Appearance Score}} \cdot \underbrace{p_o}_{\textbf{Order Penalty}} \cdot \underbrace{p_s}_{\textbf{Shape Penalty}}$$

$$\{A^*, B^*, C^*\} = \operatorname{argmax} S_{A'B'C'}$$

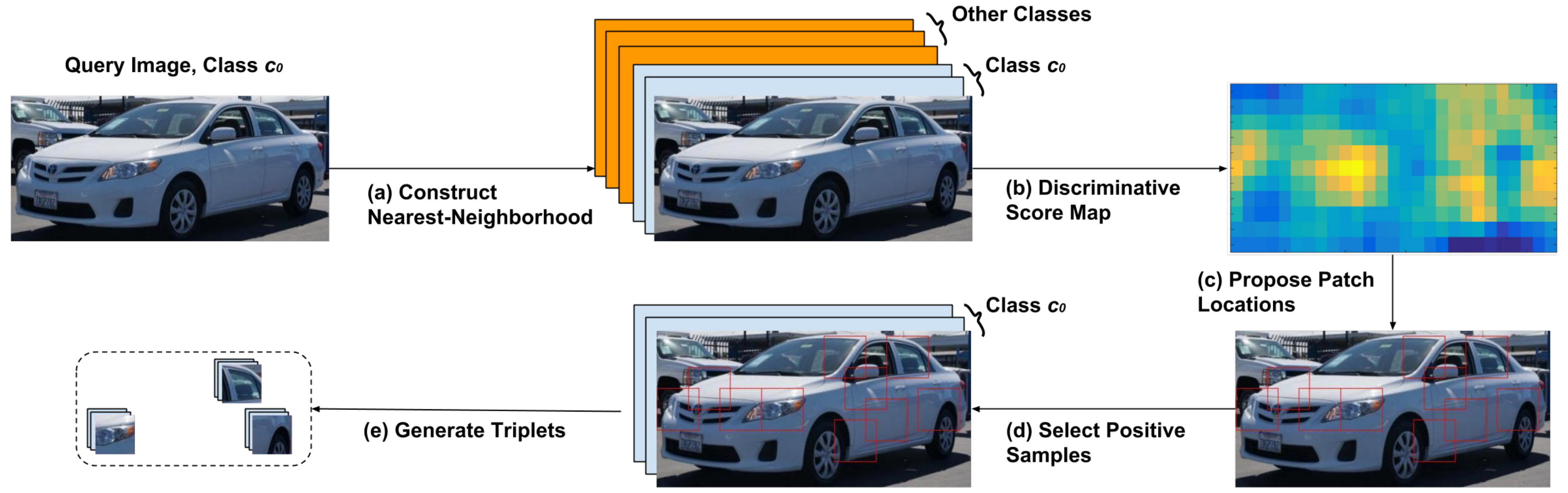
3. Mining Discriminative Triplets



Triplet Initialization → Triplet Mining → Mid-Level Representation → SVM

3.1 Triplet Initialization

- Nearest-neighbor based **local** initialization


Query Image, Class c_0
Other Classes
Class c_0
(a) Construct Nearest-Neighborhood
(b) Discriminative Score Map
(c) Propose Patch Locations
(d) Select Positive Samples
Class c_0
(e) Generate Triplets

3.2 Triplet Mining

- **Global** discovery using entropy score

$$H(\mathrm{c} \mid \mathbf{T}) = \sum_{c} p(c \mid \mathbf{T}) \log p(c \mid \mathbf{T})$$

3.3 Mid-Level Image Representation

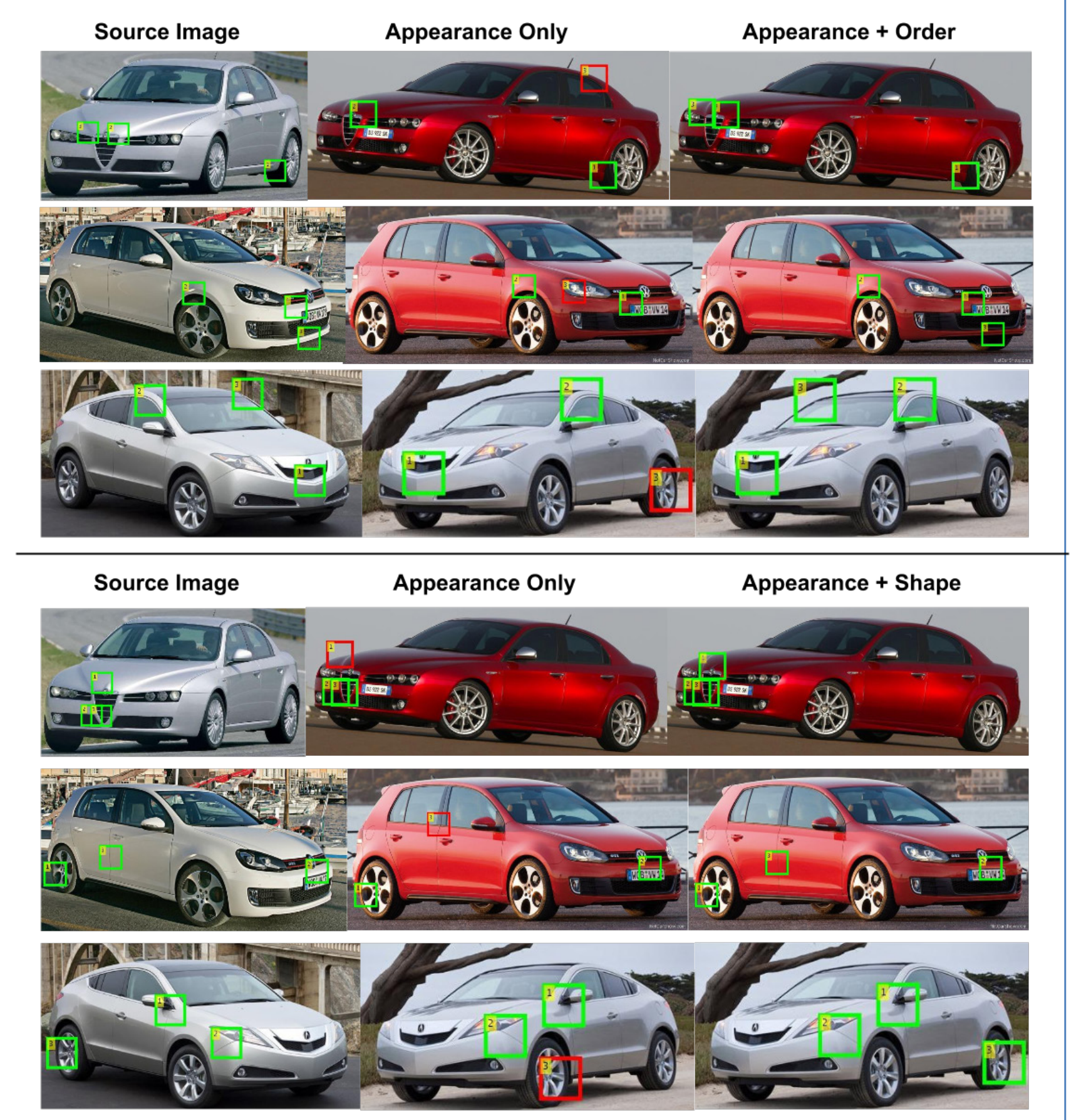
- Maximum responses of mined triplets: *Bag of Triplets* (BoT)

4. Experiments

4.1 Triplet Localization

- **FG3DCar Dataset**

Method	Localization Accuracy (%)	Improvement Over Baseline (%)
Appearance Only	24.9	-
Order Constraint	27.7	11.2
Shape Constraint	34.4	38.2
Combined	35.3	41.9



4.2 Fine-Grained Classification

- **14-Class BMVC Cars**

Method	Accuracy (%)
LLC [41]	84.5
PHOW [38]	89.0
FV [33]	93.9
structDPM [37]	93.5
BB-3D-G [25]	94.5
BoT (HOG Without Geo)	94.1
BoT (HOG With Geo)	96.6

- **100-Class FGVC-Aircraft**

Method	Accuracy (%)
Symbiotic [5]	75.9
Fine-tuned AlexNet [19]	78.9
Fisher Vector [19]	81.5
B-CNN [28]	84.1
BoT (CNN without Geo)	86.7
BoT (CNN with Geo)	88.4

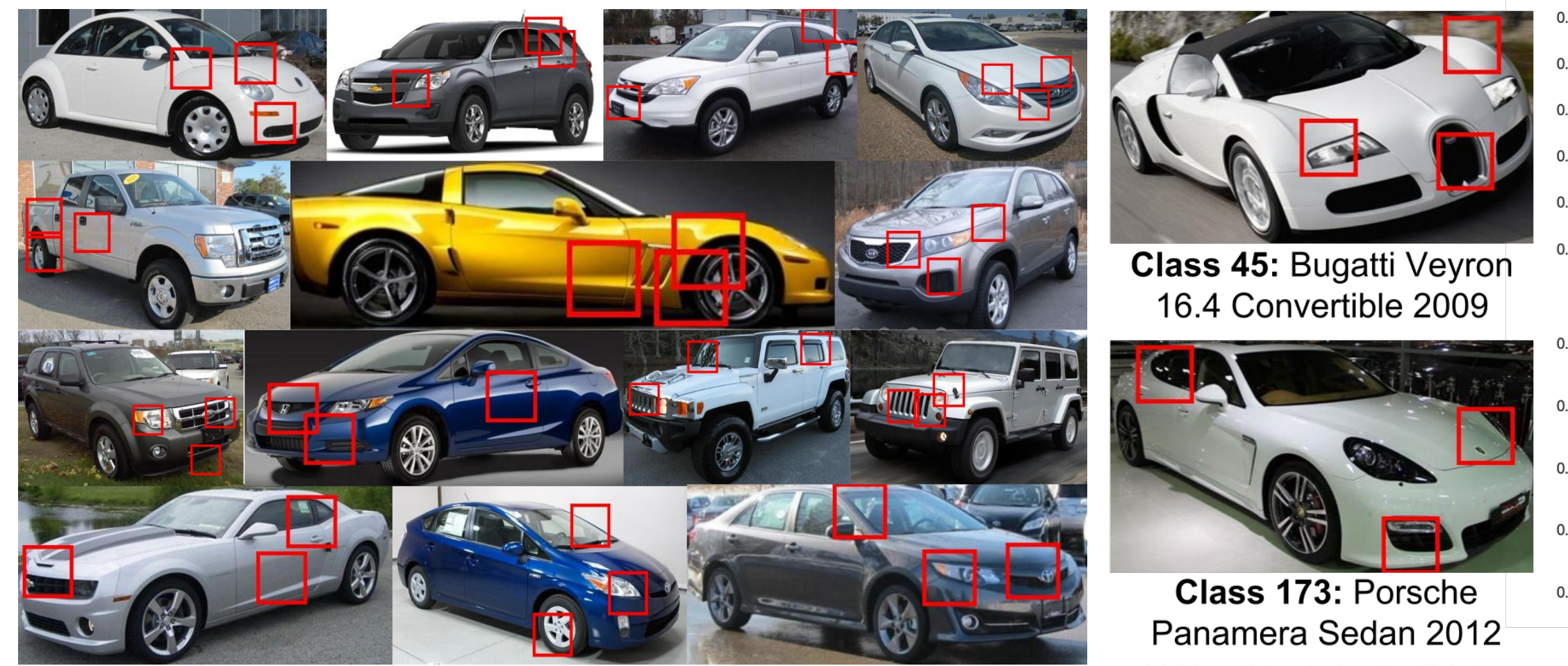
HOG: Represent patches using HOG features

- **196-Class Stanford Cars**

Method	Accuracy (%)
LLC*[41]	69.5
BB-3D-G [25]	67.6
ELLF* [23]	73.9
AlexNet From Scratch [23]	70.5
AlexNet Finetuned [43]	83.1
FT-HAR-CNN [43]	86.3
B-CNN [28]	91.3
Best Result in [24]	92.8
BoT(HOG Without Geo)*	84.6
BoT(HOG With Geo)*	85.7
BoT(CNN Without Geo)	91.2
BoT(CNN With Geo)	92.5

CNN: Represent patches using VGG-16 pool$_4$ features

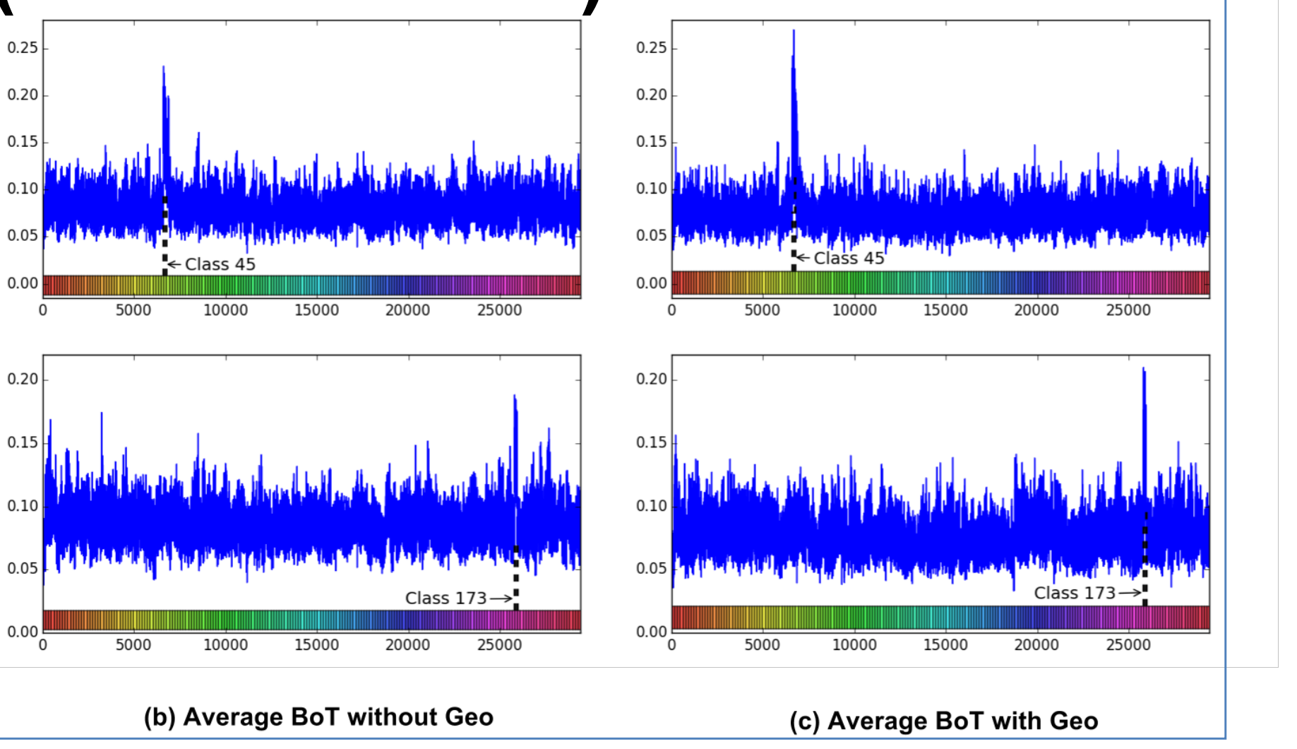
- **Most Discriminative Triplets (BMVC Cars)**
- **Descriptor Visualization (Stanford Cars)**



Class 45: Bugatti Veyron 16.4 Convertible 2009

Class 173: Porsche Panamera Sedan 2012

(a) Most Discriminative Triplet



(b) Average BoT without Geo

(c) Average BoT with Geo