

# **High-Level, Part-Based Features for Fine-Grained Visual Categorization**

**Thomas Berg**

Submitted in partial fulfillment of the  
requirements for the degree  
of Doctor of Philosophy  
in the Graduate School of Arts and Sciences

**COLUMBIA UNIVERSITY**

2017

ProQuest Number: 10257751

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



ProQuest 10257751

Published by ProQuest LLC (2017). Copyright of the Dissertation is held by the Author.

All rights reserved.

This work is protected against unauthorized copying under Title 17, United States Code  
Microform Edition © ProQuest LLC.

ProQuest LLC.  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 – 1346

PREVIEW

©2017

Thomas Berg

All Rights Reserved

## ABSTRACT

# High-Level, Part-Based Features for Fine-Grained Visual Categorization

Thomas Berg

Object recognition—“What is in this image?”—is one of the basic problems of computer vision. Most work in this area has been on finding basic-level object categories such as *plant*, *car*, and *bird*, but recently there has been an increasing amount of work in “fine-grained” visual categorization, in which the task is to recognize subcategories of a basic-level category, such as *blue jay* and *bluebird*.

Experimental psychology has found that while basic-level categories are distinguished by the presence or absence of parts (a bird has a beak but car does not), subcategories are more often distinguished by the characteristics of their parts (a starling has a narrow, yellow beak while a cardinal has a wide, red beak). In this thesis we tackle fine-grained visual categorization, guided by this observation. We develop alignment procedures that let us compare corresponding parts, build classifiers tailored to finding the interclass differences at each part, and then combine the per-part classifiers to build subcategory classifiers.

Using this approach, we outperform previous work in several fine-grained categorization settings: bird species identification, face recognition, and face attribute classification. In addition, the construction of subcategory classifiers from part classifiers allows us to automatically determine which parts are most relevant when distinguishing between any two subcategories. We can use this to generate illustrations of the differences between subcategories. To demonstrate this, we have built a digital field guide to North American birds which includes automatically generated images highlighting the key differences between visually similar species. This guide, “Birdsnap,” also identifies bird species in users’ up-

loaded photos using our subcategory classifiers. We have released Birdsnap as a web site and iPhone application.

PREVIEW

# Table of Contents

<b>List of Figures</b>	iv
<b>List of Tables</b>	x
<b>1 Introduction</b>	1
<b>2 Prior Work</b>	6
2.1 Face Recognition . . . . .	6
2.1.1 Alignment . . . . .	6
2.1.2 Hierarchical Classifiers . . . . .	8
2.2 Fine-Grained Visual Categorization . . . . .	10
2.2.1 Part-based Features . . . . .	10
<b>3 Tom-vs-Pete Classifiers and Identity-preserving Alignment</b>	12
3.1 Reference Dataset . . . . .	15
3.2 Identity-preserving Alignment . . . . .	16
3.3 Tom-vs-Pete Classifiers and Verification . . . . .	20
3.4 Results . . . . .	22
<b>4 POOF: Part-Based One-vs-One Features</b>	26
4.1 Part-Based One-vs-One Features . . . . .	29
4.1.1 Implementation details . . . . .	32
4.2 Experiments . . . . .	33

4.2.1	Bird Species Identification . . . . .	33
4.2.2	Face Verification . . . . .	36
4.2.3	Attribute Classification . . . . .	37
<b>5</b>	<b>How Do You Tell a Blackbird from a Crow?</b>	<b>41</b>
5.1	Related Work . . . . .	44
5.2	Visual Similarity . . . . .	45
5.2.1	Finding Similar Classes . . . . .	46
5.2.2	Choosing Discriminative Features . . . . .	47
5.2.3	Visualizing the Features . . . . .	47
5.3	A Visual Field Guide to Birds . . . . .	49
5.3.1	A Tree of Visual Similarity . . . . .	52
<b>6</b>	<b>Birdsnap</b>	<b>56</b>
6.1	The Birdsnap Dataset . . . . .	59
6.2	One-vs-Most Classifiers . . . . .	61
6.3	A spatio-temporal prior for bird species . . . . .	63
6.3.1	Adaptive kernel density estimation of the spatio-temporal prior . . . . .	65
6.4	Experiments on the Birdsnap Dataset . . . . .	69
6.5	Visualizing species frequency and migration . . . . .	72
6.6	Illustrating field marks . . . . .	74
6.7	A Tour of Birdsnap . . . . .	77
6.7.1	The Birdsnap Web Site . . . . .	77
6.7.2	The Birdsnap Mobile App . . . . .	82
<b>7</b>	<b>Conclusions</b>	<b>85</b>
7.1	Recent Developments . . . . .	86
	<b>Bibliography</b>	<b>87</b>

<b>Appendix: The Birdsnap Dataset</b>	<b>100</b>
A.1 Motivation Behind the Dataset . . . . .	100
A.2 Building the Dataset . . . . .	101
A.3 Comparisons with Other Datasets . . . . .	108

PREVIEW

# List of Figures

2.1 Comparison of face alignments. Top row from left to right shows the original detected face, alignment by funnelling, and alignment by similarity. The bottom row shows affine alignment, piecewise affine alignment, and our identity-preserving warp, discussed in Chapter 3.	7
3.1 The verification system. A reference set of images is used to train a parts detector and a large number of “Tom-vs-Pete” classifiers. Then given a pair of test images, we detect the parts and used them to perform an “identity-preserving” alignment. The Tom-vs-Pete classifiers are run on the aligned images, with the results passed to a same-or-different classifier to produce a decision.	13
3.2 Labeled face parts. (a) There are fifty-five “inner” points at well-defined landmarks and (b) forty “outer” points that are less well-defined but give the general shape of the face. (c) The triangulation of the parts used to perform a piecewise affine warp.	15

3.3 Warping images to frontal. (a) Original images. (b) Aligning by an affine transformation based on the locations of the eyes, tip of the nose, and corners of the mouth does not achieve tight correspondence between the images. (c) Warping to put all 95 parts at their canonical positions gives tight correspondence, but de-identifies the face by altering its shape. (d) Warping based on genericized part locations gives tight correspondence without obscuring identity. In all methods, we ensure that the side of the face presented to the camera is on the right side of the image by performing a left-right reflection when necessary. This restricts the worst distortions to the left side of the image (shown with a gray wash here), which the classifiers can learn to weight less important than the right. . . . .	17
3.4 Finding generic parts. (a) The fiducial detector gives the inner part locations (yellow triangles) of the probe image. (b) For each reference subject, we find the image with inner parts closest, under similarity, to the detected probe parts. (c) Averaging the ( <i>inner and outer</i> ) part locations over this set of reference images gives the “generic” inner (blue circle) and outer (pink square) parts. (d) A close up of the eye shows that this subject’s eye is slightly longer (left-to-right) with less distance from eye to brow than the average eye. For clarity, only a subset of the full 95 parts are shown in this figure. . . . .	19
3.5 The top left image is produced by the alignment procedure. Each of the remaining images shows the region from which one low-level feature is extracted. SIFT descriptors are extracted from each square and concatenated. Concentric squares indicate SIFT descriptors at the same point but different scales. . . . .	20

3.6	(a) A comparison with the best published results on the LFW image-restricted benchmark, including the Associate-predict method [Yin <i>et al.</i> , 2011], Brain-inspired features [Pinto and Cox, 2011], and Cosine Similarity Metric Learning (CSML) [Nguyen and Bai, 2011], (b) The log scale highlights the performance of our method at the low-false-positive rates desired by many security applications. . . . .	23
3.7	lfw benchmark results. (a) The contribution of Tom-vs-Pete classifiers, compared to random projection or low-level features. (b) The contribution of the alignment method, compared with a piecewise affine warp using non-generic part locations or a global affine transformation. . . . .	24
4.1	Learning a Part-based One-vs-One Feature (POOF) for bird species identification. Given (a) a reference dataset of images labeled with class (species) and part locations, a POOF is defined by specifying two classes, one part for feature extraction, another part for alignment, and a low-level “base feature.” (b) Samples of the two chosen classes are taken from the dataset and (c) aligned to put the two chosen parts in fixed locations. (d) The aligned images are divided into cells at multiple scales, from which the base feature is extracted. A linear classifier is trained to distinguish the two classes, giving (e) a weight to each cell. We threshold the weights and find the maximal connected component contiguous to the chosen feature part, setting this as (f) the support region for the POOF. Finally, a classifier is trained on the base feature values from just the support region. The output of this classifier is our one-vs-one feature. . . . .	27
4.2	Bird species classification accuracy on (a) the full 200-species CUB benchmark and (b) the “birdlets” subset of 14 woodpeckers and vireos defined in [Farrell <i>et al.</i> , 2011]. . . . .	31
4.3	Face parts from the detector of [Belhumeur <i>et al.</i> , 2011]. . . . .	34

4.4	Results on the LFW benchmark. (a) POOFs and the top four previous published results. (b) Comparison of POOFs with low-level features. . . . .	36
5.1	(a) For any bird species (here the red-winged blackbird, at center), we display the other species with most similar appearance. More similar species are shown with wider spokes. (b) For each similar species (here the American crow), we generate a “visual field guide” page highlighting differences between the species. . . . .	42
5.2	A similarity tree of bird species, built from our visual similarity matrix. Species similar to the red-winged blackbird are in blue, and species similar to the Kentucky warbler are in red. . . . .	43
5.3	Visual field guide pages for the Kentucky warbler. . . . .	50
5.4	The phylogenetic “tree of life” representing evolutionary history. Species visually similar to the red-winged blackbird are in blue, and those similar to the Kentucky warbler are in red. Although the American crow and common raven are visually similar to blackbirds, they are not close in terms of evolution. . . . .	51
5.5	Similarity matrices. (a) Visual similarity. (b) Phylogenetic similarity. In both, rows/columns are in order of a depth-first traversal of the evolutionary tree, ensuring a clear structure in (b). The large dashed black box corresponds to the passerine birds (“perching birds,” mostly songbirds), while the small solid black box holds similarities between crows and ravens on the y-axis and blackbirds and cowbirds on the x-axis. . . . .	52
5.6	The top three visually similar, phylogenetically dissimilar species pairs from Table 5.1. First row: Gadwall and Pacific Loon. Second row: Hooded Merganser and Pigeon Guillemot. Third row: Red-breasted Merganser and Eared Grebe. Example images are chosen for similar pose. . . . .	54

6.1	The main, species-browsing page of the Birdsnap web site. Species can be arranged by the phylogenetic “Tree of Life” (shown), by visual similarity (as described in Section 5.3.1), by sighting frequency at the currently selected place and date (based on the spatio-temporal prior described in Section 6.3), or alphabetically. . . . .	57
6.2	The main screen of the Birdsnap iPhone app, a simpler version of the browsing wheel on the web site. . . . .	58
6.3	Sample images from the Birdsnap dataset, with bounding boxes and part annotations. The species of these samples, from left to right, are Northern Cardinal, Broad-tailed Hummingbird, Great Egret, Black-headed Grosbeak, and Nuttall’s Woodpecker. . . . .	59
6.4	One-vs-most classifiers (top) improve both overall accuracy and the consistency and “reasonableness” of classification results. . . . .	62
6.5	Fixed-time slices of our spatio-temporal prior show the Barn Swallow arriving from South America during its spring migration (left) and established in its summer grounds (right). Brighter regions indicate higher likelihood of a sighting. . . . .	64
6.6	One-vs-most accuracy omitting the $k$ most similar classes from training. As we increase $k$ , accuracy of the one-vs-most classifiers initially increases at all ranks. Results for additional values of $k$ , shown in Table 6.1, are omitted for clarity. . . . .	68
6.7	Mean visual distance between query species and returned species. One-vs-most classifiers return species that are more similar to the query species. . .	70
6.8	The one-vs-most classifiers and spatio-temporal prior each contributes significantly to overall performance. The dashed line, using labeled part locations, shows hypothetical performance with human-level part localization. .	70

6.9	Species density over time in a fixed location. The “raw density” is the estimate from Section 6.3.1. Applying a median filter and adaptive threshold lets us recognize the Wild Turkey as present year round, despite the low frequency. . . . .	73
6.10	Field marks differentiating the Great Egret and the Snowy Egret. By filtering based on Tanimoto similarity, we ensure that we find three <i>different</i> features: beak color, the extension of the mouth beneath the eye, and the long, slender neck. In contrast, the top three features found by the method of Chapter 5 without filtering all appear to relate to beak color. . . . .	75
6.11	List view of species on the Birdsnap web site, here sorted by sighting frequency at the specified date and location. . . . .	78
6.12	Detail view for the Golden-winged Warbler on the web site (where it appears as a single, scrollable column). . . . .	79
6.13	Fields marks view from the web site, showing the differences between the Golden-winged Warbler and the Chestnut-sided Warbler with illustrations generated by the process described in Section 6.6. . . . .	81
6.14	The recognition submission window on the web site, after the user has clicked on the eye and tail. . . . .	82
6.15	Screens from the Birdsnap iPhone application. . . . .	83
A.1	The Amazon Mechanical Turk interface for bounding box labeling. . . . .	102
A.2	The Amazon Mechanical Turk interface for part labeling. . . . .	103
A.3	The Amazon Mechanical Turk interface for species and sub-species class labeling. . . . .	106

# List of Tables

1.1	Winning mean average precision on the Pascal VOC Classification Challenge (Competition 1) [Everingham <i>et al.</i> , 2005 2013] . . . . .	2
1.2	Winning top-5 accuracy on the ImageNET classification challenge. For 2015 and 2016 we show the best classification accuracy on the classification and localization challenge, as the separate classification challege was discontinued after 2014. . . . .	3
4.1	Attribute classification accuracy. For each attribute, the top row is baseline accuracy using the low-level base features (color and gradient direction histograms) directly, and the bottom row is accuracy using POOFs. The more accurate is bold. The last column gives accuracy of [Kumar <i>et al.</i> , 2011] on the same test images, in bold when better than the POOF 600-sample classifier. The last row shows the average improvement using POOFs over the low-level features or [Kumar <i>et al.</i> , 2011]. As these are binary attributes, chance gives 50% accuracy. . . . .	38
5.1	Species pairs with high visual and low phylogenetic similarity. . . . .	53
6.1	Accuracy of the one-vs-most classifiers increases at all ranks as $k$ increases to 15. Beyond $k = 15$ , high-rank accuracy continues to increase, but rank-1 accuracy decreases. . . . .	67
A.1	Species of the Birdsnap dataset, with image and category counts. Part 1 of 3. 110	

# Acknowledgments

After spending a few years programming front-office systems in Tokyo, facing yet another project where the great challenge would be getting our clients to agree on the optimal tab-ordering of fields on the data entry screens, I began to wonder if learning a little computer science might put me in the way of more interesting work. So I convinced my girlfriend (now wife) to “visit” the States and headed back home to go to school. Thank you, Aya, for coming with me. I couldn’t have got through this on my own.

With no training in computer science, I didn’t interest the graduate schools that interested me, so I began taking classes at Harvard’s (open admissions) extension school. Two instructors in particular, Jamie Frankel and David Albert, fed my interest and had enough faith to write the recommendation letters I needed to get in to Columbia. Thank you, Jamie and David, for pushing me along.

At Columbia, I planned on a quick master’s degree, then back to work at a higher pay-and interest-grade. But in my first semester, I took some *very interesting classes*. Shree Nayar’s and Peter Belhumeur’s classes got me interested in computer vision and pattern recognition. Then a summer project with Peter Allen and Corey Goldfeder gave me a taste of research. Thank you, Shree and Peter and Peter and Corey, for showing me the good stuff. And especially thank you Peter, for letting me stick around when I decided I wanted to stay for a PhD.

I couldn’t seem to get anything to work my first couple years, but the advice, friendship, and examples of my fellow students kept me going and even made it fun. Thank you Neeraj Kumar for your enthusiasm and patience, Matei Ciocarlie for your welcome, Ollie Cossairt for your quiet excellence and hidden hippie outlook, Austin Reiter for your matter-of-fact

skills and speed, Jiongxin Liu for your discipline, persistence, and singing voice, Hao Dang for your cheer.

With an inexhaustible well of ideas and intuition from Peter, and his unerring nose for implementation bugs, our work began to work. Thank you Peter, for teaching me how to think clearly, how to write, and how to argue with you.

After a few years, with my graduation “imminent”, I left New York for California, remaining a Columbia student until I put the final touches on my thesis. These final touches have now taken over two years, during which Daniel Miau has leapt into the breach whenever a server went down or a disk failed. Thank you, Daniel, for being my eyes and hands on campus.

During all my time at Columbia, Anne Fleming has made the administrative bumps as smooth as they could possibly be. Daisy Nguyen has done the same for all my hardware troubles. Thank you Anne and Daisy.

This thesis is by all of us.

PREVIEW

For Aya, as everything

# Chapter 1

## Introduction

Object detection and classification are among the most-studied problems in computer vision. Where are the objects in this image, and what are they? A lot of research effort has gone toward this problem, and a lot of progress has been made.

One measure of this progress is the PASCAL Visual Object Classes (VOC) challenges [Everingham *et al.*, 2014]. In the classification challenge, as administered each year from 2007 to 2012, we are presented with an image and asked to classify it according to the most prominent object it contains: an airplane, bicycle, boat, bus, car, motorbike, train, bottle, chair, dining table, potted plant, sofa, TV / monitor, bird, cat, cow, dog, horse, sheep, or person. The training set consists of 11,540 images across these classes. As shown in Table 1.1, mean average precision of the top-performing method rose from 59.4% in 2007 to 82.2% in 2012 [Everingham *et al.*, 2005 2013]. Although the challenge officially ended in 2012, later work using the 2012 dataset has achieved mean average precision as high as 85.4%, or 94.3% when using additional training data [Everingham *et al.*, 2016].

The heir to the PASCAL VOC challenges is the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), run annually since 2010. In comparison with the PASCAL VOC, ILSVRC has one thousand categories instead of twenty and 1.2 million training images instead of twelve thousand. To account for the possibility of additional, unlabeled objects in the test images, classification is considered correct if any of five class labels pre-

Year	Winning mAP
2007	59.4%
2008	58.6%
2009	66.5%
2010	73.8%
2011	78.6%
2012	82.2%

Table 1.1: Winning mean average precision on the Pascal VOC Classification Challenge (Competition 1) [Everingham *et al.*, 2005 2013]

dicted by the algorithm match the test label. As shown in Table 1.2, this top-5 accuracy has improved each year, from 71.8% in 2010 to 97.0% in 2016 [Russakovsky *et al.*, 2015; Liu *et al.*, 2015; Liu *et al.*, 2016].

As object detection and classification improve, and further gains become harder to come by, it has become more important to explore the errors made by state-of-the-art methods and how these errors can inform further research. In an analysis of PASCAL VOC object detection results from two top methods, Hoiem *et al.* [Hoiem *et al.*, 2012] find the primary cause of false positives, to be “similar category” errors, in which a detector for one class fires on an instance of a similar class, for example, when the horse detector fires on a cow. These are the most common errors on average across all twenty classes, even though some classes (bottle, potted plant, and TV / monitor) have *no* similar classes and therefore *no* errors of this type at all. If we increase the number of categories, we expect this type of error to become more common, as each category will have more similar categories and the difference between similar categories will decrease. An analysis of the best results on the ILSVRC [Russakovsky *et al.*, 2013] confirms this, noting for example that the best method can distinguish dogs from non-dogs with 99% accuracy, but is much less reliable in distinguishing the 120 different dog breeds in the challenge from each other.

This problem of distinguishing very similar categories from each other is the problem

Year	Winning Top-5 Accuracy
2010	71.8%
2011	74.2%
2012	83.6%
2013	88.3%
2014	93.3%
2015	96.4%
2016	97.0%

Table 1.2: Winning top-5 accuracy on the ImageNET classification challenge. For 2015 and 2016 we show the best classification accuracy on the classification and localization challenge, as the separate classification challenge was discontinued after 2014.

of *fine-grained visual categorization* (FGVC) and is the topic of this thesis. Examples of fine-grained categorization include recognizing breeds of dog, species of bird, or models of automobile. This thesis focuses on the use of *parts* of the object for fine-grained recognition, for two reasons.

First, we are guided by results from the study of human perception. Psychologists use the terms “superordinate level,” “basic level,” and “subordinate level” to describe levels in the taxonomic hierarchy with which we label the objects in our environment. While not always perfectly well-defined, in general the basic level is the level at which we most readily recognize and label objects. For example, barring a reason to be more or less specific, we are more likely to say we saw a “car” (basic level) than a “vehicle” (superordinate level) or a “Toyota Camry” (subordinate level). Similarly we say “bird” rather than “animal” or “starling,” and “hammer” rather than “tool” or “ball-peen hammer.” In these terms, fine-grained categorization is the problem of distinguishing subordinate-level categories of the same basic-level category from each other.

Work in experimental psychology suggests that the basic level is often defined by the presence or absence of parts [Tversky and Hemenway, 1984]. A car has four wheels, a

bird has a beak and feathers, and a hammer has a head and a handle. Subordinate-level categories, in contrast, are often distinguished by characteristics of their parts: a starling has an orange beak and spotted feathers. If humans perform fine-grained categorization by considering characteristics of the parts, it’s appealing to have our algorithms look to the parts as well.

Second, although FGVC is relatively young as a named area of research in computer vision, it has much in common with a very well-studied *instance*-level classification problem: face recognition. Instance recognition is often considered a different problem from FGVC, at one end of a granularity spectrum with basic-level recognition at the other end and FGVC in the middle. But the difficulty in fine-grained and instance-level recognition is essentially the same: small inter-class differences often swamped by intra-class differences, so we consider face recognition (and instance recognition in general) as an example of fine-grained categorization, where the basic-level category is *face* and the subordinate-level categories are individuals. The best methods of face recognition all include finding parts of the face (eyes, nose, etc.) so that corresponding parts of faces can be compared with each other, so it’s natural to investigate whether the use of parts is important for fine-grained recognition in other domains as well.

In this thesis, we develop a set of part-based features for fine-grained visual categorization and demonstrate their application to several problems.

After discussing relevant prior work in Chapter 2, in Chapter 3 we consider the problem of face verification, matching faces by identity. Using a set of 95 parts on the face, we design a method for alignment that brings the faces into correspondence while preserving interclass differences, and a method for learning a set of stacked classifiers that distinguish one face from another. In Chapter 4, we generalize this to learn a set of part-based features we call “POOFs” to distinguish subcategories in any domain, and demonstrate the generalized method’s effectiveness at classification of human faces and birds. Subcategorization using POOFs closely follows the intuition from experimental psychology, as each feature is built to measure a characteristic of a particular part, and these part-specific features are

combined to build the subcategory classifiers.

In Chapter 5, we consider applications of part-based features beyond automatic recognition, in particular how we can use these features to develop an understanding of the visual domain defined by a basic-level category. Again taking birds as our example, we automatically determine which subcategories (species) are most similar to each other and annotate images to show the key differences that distinguish similar species from each other. Finally, in Chapter 6, we describe the application of these ideas to build Birdsnap, a publicly-available digital field guide to birds, implemented as a web site and an iPhone App. We use the methods of Chapter 5 to build the guide, illustrating the similarities and differences between species, and use the methods of Chapter 4 to perform automatic identification of the birds in users' uploaded photos. This section describes the challenges we encountered when applying our methods to build a real, useful system, and the modifications to our methods that these challenges necessitated.

# Chapter 2

## Prior Work

### 2.1 Face Recognition

The “finest” fine-grained categorization is instance recognition, where we must identify individual instances of a class, and the best-studied example of instance recognition is face recognition. So we look to prior work on faces. The full body of work on face recognition is too large to survey here, so we focus on two aspects relevant to our work, alignment and hierarchical classifiers.

#### 2.1.1 Alignment

It is well established that alignment is critical for good performance in face recognition with uncontrolled images ([Gu and Kanade, 2008; Wang *et al.*, 2006; Wolf *et al.*, 2009]). One method often applied is [Huang *et al.*, 2007a]’s “funneling,” which extends the congealing method of [Learned-Miller, 2006] to handle noisy, real-world images. These methods find transformations that minimize differences in images that are initially only roughly aligned. Another common technique is to apply a similarity or affine transformation to the images based on the locations of detected fiducial points such as the corners of the eyes and mouth. Due to both their effectiveness and the fact that pre-aligned images for the standard “Labeled Faces in the Wild” (LFW) face verification dataset are publicly avail-