DATA-DRIVEN COMPUTER VISION

FOR SCIENCE AND THE HUMANITIES

Stefan Lee

Submitted to the faculty of the University Graduate School

in partial fulfillment of the requirements for the degree

Doctor of Philosophy

in the School of Informatics and Computing,

Indiana University

August 2016

Accepted by the Graduate Faculty, Indiana University,

in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Doctoral Committee

_____

David Crandall, PhD

_____

Michael Ryoo, PhD

_____

Predrag Radivojac, PhD

_____

Chunfeng Huang, PhD

July 20th, 2016

To the cabin!

ACKNOWLEDGMENTS

First and foremost, I would like to thank my partner Brenda Peters for her constant support, for enduring the patient years we spent apart, and for providing me with the courage to try. I can't imagine I would have ever gotten here without you.

I owe a great deal of gratitude to Sven Bambach. Your companionship has been an anchor throughout these long years. We've shared many things over our time at Indiana University: classes, lab space, flights, and (for a while) a home. Through this time you've challenged me to be better and encouraged me through my failures. I'm proud to say that you are the best friend I've ever had.

Special thanks to my advisor David Crandall for his guidance, patience, and innumerable reassurances throughout this journey. It is hard to foresee the long reaching effects your involvement will have on my life; however, I can say with certainty that the course of it has been permanently diverted by your influence. I can't thank you enough for the years of understanding and effort you've consistently shown to me and all of your students.

I would like to thank my research committee members Michael Ryoo, Predrag Radivojac, and Chunfeng Huang, and all the other faculty and staff of Indiana University

who have graciously shared their expertise and time. Additional thanks to Josef Sivic, Alexei Efros, and Dhruv Batra for their time spent hosting and mentoring me at their institutions.

This thesis is based on work supported in part by the National Science Foundation (IIS-1253549, CNS-0723054, and OCI-0636361), the National Institutes of Health (R01-HD074601 and R21-EY017843), the Intelligence Advanced Research Projects Activity (IARPA) via Air Force Research Laboratory (FA8650-12-C-7212), the European Research Council (LEAP no. 336845), the Agence Nationale de la Recherche (Semapolis project, ANR-13-CORD-0003), the INRIA CityLab IPL, and the Indiana University Vice President for Research through an IU Collaborative Research Grant. Additionally, computing resources used as part of this thesis are supported in part by NSF (ACI-0910812 and CNS-0521433), the Lily Endowment, Inc., and the Indiana METACyt Initiative. Any opinions, findings, and conclusions or recommendations expressed are those of the author and do not necessarily reflect the views of the sponsoring institution.

Stefan Lee

DATA-DRIVEN COMPUTER VISION FOR SCIENCE AND THE HUMANITIES

The rate at which humanity is producing visual data from both large-scale scientific imaging and consumer photography has been greatly accelerating in the past decade. This thesis is motivated by the hypothesis that this trend will necessarily change the face of observational science and the humanities, requiring the development of automated methods capable of distilling vast image collections to produce meaningful analyses. Such methods are needed to empower novel science both by improving throughput in traditionally quantitative disciplines and by developing new techniques to study culture through large scale image datasets.

When computer vision or machine learning in general is leveraged to aid academic inquiry, it is important to consider the impact of erroneous solutions produced by implicit ambiguity or model approximations. To that end, we argue for the importance of algorithms that are capable of generating multiple solutions and producing measures of confidence. In addition to providing solutions to a number of multi-disciplinary problems, this thesis develops techniques to address these overarching themes of confidence estimation and solution diversity.

This thesis investigates a diverse set of problems across a broad range of studies including glaciology, developmental psychology, architectural history, and demography to develop and adapt computer vision algorithms to solve these domain-specific applications. We begin by proposing vision techniques for automatically analyzing

aerial radar imagery of polar ice sheets while simultaneously providing glaciologists with point-wise estimates of solution confidence. We then move to psychology, introducing novel recognition techniques to produce robust hand localizations and segmentations in egocentric video to empower psychologists studying child development with automated annotations of grasping behaviors integral to learning. We then investigate novel large-scale analysis for architectural history, leveraging tens of thousands of publicly available images to identify and track distinctive architectural elements. Finally, we show how rich estimates of demographic and geographic properties can be predicted from a single photograph.

David Crandall, PhD

Michael Ryoo, PhD

Predrag Radivojac, PhD

Chunfeng Huang, PhD

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 THESIS OVERVIEW

It is well-accepted and perhaps often over-stated that we have entered into the era of "big data"; however, what is typically left unsaid is the fact that a large fraction of this data takes the form of images and videos. Social media, photo-sharing websites, and large-scale scientific imaging are producing tremendous quantities of visual data, introducing new opportunities and challenges for many academic and scientific disciplines. When faced with the overwhelming flood of available images, traditionally manual methods prevalent in many disciplines are insufficient in either throughput or sensitivity to capture patterns in the data. The development of new computer vision approaches informed by domain-specific knowledge will be an integral part of overcoming these challenges and opening up novel avenues of data-driven academic inquiry fueled by visual data.

In the following sections, we will argue for the need for automated visual analysis in sciences and the humanities, discuss the taxonomy of tasks addressed in this thesis, and then summarize our contributions and the structure for the remaining chapters.

1.2 THE NEED FOR AUTOMATED VISUAL ANALYSIS

Recent work has shown great success using large-scale data analysis to enable academic inquiries; however, these methods have typically been based on non-visual information. For example, collaborations between sociologists and computer scientists are using social network data to measure human behavior at unprecedented scales [72], while work in health informatics is using online data to monitor outbreaks of diseases [46] and to predict their spread [111] (albeit with some controversy [110] and missteps [71]), and geologists have applied machine learning to predict the magnitude of upcoming earthquakes [1]. In the humanities, analysis of data has given insight into historical legal records [67] and the dynamics of cultural history [112]. Large-scale analysis of digitized books through several centuries has been used to quantify changes in linguistic and cultural phenomena over time [91]. The success of these methods is encouraging and points to the potential of harnessing image data to enable similar pursuits in both traditionally qualitative and quantitative domains.

In many academic domains, the introduction, development, and wide-spread use of digital imaging has greatly increased the rate at which data can be extracted from experiments; however, traditional means of analysis are not feasible at these greater throughputs. In medical imaging and biology, high-speed microscopy has opened new and promising avenues of research by capturing terabytes of images [64]. Satellites pointed towards the cosmos and back at Earth are constantly capturing images of our universe at high resolution [131]. Archaeologists and art historians are documenting artifacts and structures not only with traditional imagery but also using reconstructions made from huge numbers of individual images [33, 69]. Social

media platforms which share visual media such as Youtube, Instagram, Facebook, and Flickr house billions of images and gain many millions more a day [15]. In order to harness these new data sources and not find ourselves surrounded by mountains of undecipherable data, we need to develop automated approaches to organize, annotate, and analyze visual data.

Some work has begun to fill this space and demonstrate how computer vision techniques can enhance existing methodologies. Automated processing of medical fMRI images has helped to identify brain abnormalities due to prenatal cocaine exposure [35] and blood vessel detection in images of patients' inner eyes has automated early diagnosis of many diseases including diabetes [38]. Methods have been developed in biology to track the movement of both fine-grained bodies in microscopy imaging [21] as well as whole animals in behavioral studies [29], helping to improve our understanding of living organisms. Citizen science applications have used publicly shared photographs to estimate ecological phenomena [134, 144], and automated animal detection in trail cameras has helped to track biodiversity metrics for ecological studies [141]. Vision techniques have begun to assist in sociological studies by automatically estimating signs of hostility or rapport between interacting subjects [19]. Other work has estimated geospatially distributed statistics such as crime rate [7], neighborhood safety, uniqueness, and wealth [98]. Work in the humanities has investigated using vision to organize and navigate historical images [8, 113] and to discover hidden features in artwork [65, 123].

In an ideal world, academics working in visual domains would be able to utilize computer vision techniques similarly to other established tools; however, general purpose computer vision remains limited in its usefulness and applicability. Achieving

suitable solutions for many tasks requires tailored models and techniques that consider the characteristics of the source domains. Other problems can be reduced to common frameworks; however, the difficulty in parameterizing these problems such that they map to established techniques is still a substantial hurdle for non-experts. Moreover, the latency in integrating novel methods into publicly available software packages useful to non-experts often leaves results well behind state-of-the-art.

1.3 ANNOTATION AND DISCOVERY

This thesis contends that computer vision techniques will be necessary to enable novel science in many academic disciplines and that computer vision may already be powerful enough to meaningfully impact some domains. To this end, we develop novel techniques that demonstrate the ability of computer vision to help answer an exemplar set of four research questions related to diverse academic domains:

1) How well can ice layers be automatically found in polar radar imagery and to what extent can solution confidence be estimated?
2) To what degree can spatial biases be leveraged to improve hand detection and disambiguation in egocentric videos of social interaction?
3) Can architectural trends be discovered with large-scale visual pattern mining?
4) How informative are visual attributes of consumer photographs for predicting high-level demographic and geographic attributes of a scene and to what extent can they be recognized automatically?

In answering these questions, we demonstrate the usefulness of computer vision to academic inquiry and show how common frameworks can be applied to seemingly disjoint problems. These problems are both representative of the space of scientific

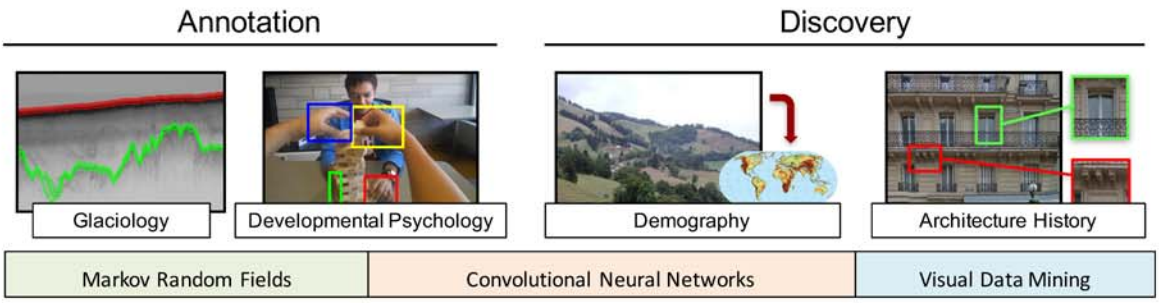


Figure 1.1: We address four problems from diverse academic domains in this thesis. These tasks are exemplars of both annotation problems where the goal is to identify particular structures for which appearance and characteristics are known in advance and discovery tasks where new information is to be found from large image collections. We apply three main techniques - Markov Random Fields, Convolutional Neural Networks, and Visual Data Mining - to provide solutions to these problems.

tasks and encompass many core computer vision areas including visual data mining, structured prediction, image classification, semantic segmentation, and object detection.

Broadly speaking, the visual academic problems presented here can be divided into two classes, annotation and discovery, which cover the space of visual scientific tasks. Ice layer tracing in polar radar images and hand detection in egocentric video are both annotation tasks and exemplars of a wide range of problems where the goal is simply to extract some obvious visual feature from data for further analysis. Annotation problems are often complicated by adverse imaging conditions (such as noise in radar images or frequent occlusions in egocentric video). Humans are typically adept at providing reasonable annotations even in these challenging conditions, but doing so tends to be extremely laborious and time consuming. Computer vision solutions in this space of problems can be viewed as automated annotation to replace human labor for visual structure localization.

In discovery tasks such as demographic attribute prediction and architectural trend discovery, the amount of data is often so large and the signals are so weak that they are unlikely to be found without years of human effort. Compared to annotation tasks, discovery tasks do not start with a previously known characterization of the visual elements important to the task. Instead, the relationships between higher-level concepts and visual features must be discovered in these tasks. For example, a training set for finding hands provides direct visual representations of hands, whereas a set of image-level labels of poverty does not directly describe the visual evidence of poverty. Automated solutions in this space give researchers enhanced analytical abilities to discover relationships in large-scale data.

1.4 SUMMARY AND THESIS OUTLINE

In this thesis, we propose computer vision approaches to solve a diverse set of problems arising in glaciology, developmental psychology, architectural history, and demography. Common to many of these applications is the need for multiple good solutions and confidence estimations. Systems which provide scientists and researchers interpretable results and diverse solutions in the face of ambiguity produce more valuable information for higher level analysis than "black box" solutions which simply yield a single estimate. Throughout this thesis, we develop these themes while demonstrating effective solutions to multiple academic problems. The structure of the remainder of this work is as follows:

– In **Chapter 2**, we provide background on the primary models used in this thesis.

– In **Chapter 3**, we present a novel technique that provides state-of-the-art precision

for layer tracing in aerial radar imaging of polar ice sheets while simultaneously providing glaciologists with point-wise estimates of solution confidence. To overcome the ambiguity and noise in the radar imagery and to estimate solution confidence, we model the layer finding problem as a structured prediction task over a probabilistic graphical model. We solve for layer positions through the use of a Markov Chain Monte Carlo method.

– In **Chapter 4**, we estimate robust hand localizations in egocentric video to empower developmental psychologists with automated annotations of grasping behaviors integral to studying learning. Egocentric video exhibits more extreme camera motion and much more frequent object and scene occlusions than traditional photography, but it also contains implicit spatial biases with respect to the camera wearer's body. We take advantage of these biases to provide solutions both for videos taken in tightly constrained laboratory settings and general environments. In the laboratory data, we pose hand tracking as structured prediction on a probabilistic graphical model and once again solve using a MCMC method. For general environments, we treat locating hands as an object detection task and leverage the spatial biases to direct powerful Convolutional Neural Network (CNN)-based appearance models.

– In **Chapter 5**, we enable novel large-scale analysis for architectural history by leveraging tens of thousands of publicly available images to identify and track the changes of temporally distinctive architectural elements, providing architecture historians with a large collection of relevant facade elements. We pose this as a large-scale visual data mining problem using image features to estimate real world

occurrence frequency and develop a graph-based framework to identify and track changing elements.

– In **Chapter 6** we show the effectiveness of state-of-the-art classification architectures at estimating the demographic and geographic properties of places in the world, based on single images. This enables automatic estimation of coarse demographic properties without the need for a formal survey. To discover the weak visual signals that inform these attributes, we automatically annotate a large collection of publicly available images to train powerful CNN-based image classification models.

– **Chapter 7** consists of closing statements regarding the work presented in this thesis, its place in the larger scientific community, and potential directions for future work.

CHAPTER 2

MODELING VISUAL PHENOMENON

The visual world is a landscape of colors, illuminations, and textures moving on deformable agents and objects, which dynamically interact with each other and the space around them. Consequentially, modeling it effectively is complex and many sophisticated models have been adapted from machine learning and statistics to attempt to manage the ambiguity of the visual world. In this chapter, we will provide background on the major models used in this thesis, with a special focus on how these models can be used to provide confidence estimation or be made to produce multiple high-quality solutions.

2.1 MARKOV RANDOM FIELDS

The visual world is full of many sources of ambiguity which can make reasoning about images and video difficult. Take for example the task of identifying the species of dog in a photograph. Partial occlusion, poor lighting, and even the similarity of many species of dog can easily make this seemingly simple problem extremely complex. Probability theory provides a natural foundation on which to build systems that must reason under uncertainty. Probabilistic models are flexible enough to model a wide range of problems while still being amicable to a set of standard inference

algorithms under reasonable conditions. These facts have made probabilistic models very popular in computer vision, being the de facto standard for many years. This thesis makes use of one popular class of probabilistic model, the Markov Random Field (or MRF) [66], which the remainder of this section covers.

Markov Random Fields are a class of undirected (and possibly cyclic) probabilistic graphical models, i.e. they are a collection of random variables having Markov properties described by some undirected graph. In these MRF graphs, each node corresponds to a random variable and the edges between them indicate some form of interaction; therefore, designing these graphs is equivalent to explicitly encoding independence and conditional relationships between the random variables. Because these models have undirected dependencies (in contrast to Bayesian networks [28]), these dependencies are modeled as general set functions called factors rather than by true conditional probability distributions [68]. The full joint probability distribution over an MRF can be written as the normalized product of these factors; however, this distribution is typically intractably large to compute or store in vision contexts. The flexibility offered by MRFs both in terms of encoding undirected dependencies and allowing for generalized factors between variables makes them an attractive model for vision problems, which often have complex relationships between many image regions.

Inference and Confidence Estimation in Markov Random Fields

Similar to other probabilistic graphical models, inference in MRFs can be accomplished using a variety of algorithms. Often the goal of inference is to produce the *maximum a posteriori* or MAP estimate from the model given some setting of model variables corresponding to a specific image; however, exact inference for cyclic MRFs

1: Initialize $X^{(0)} = \{x_1, ..., x_m\}$;
2: $j = 1$;
3: **while** $j < J$ **do**
4: $\quad X^{(j)} = X^{(j-1)}$;
5: $\quad$ **for all** x_i in $X^{(j)}$ **do**
6: $\quad\quad x_i^{(j)} \sim P(x_i | X^{(j)} - \{x_i^{(j)}\})$;
7: $\quad$ **end for**
8: $\quad j = j + 1$;
9: **end while**

Figure 2.1: The general algorithm for Gibbs sampling iteratively draws new values for random variables conditioned both on observed variables and other unknowns.

is NP-hard in the general case. The related problem of estimating the marginal distribution of each unobserved variable conditioned on the observed variables has efficient approximate solutions using either loopy belief propagation [101] (i.e. sum-product message passing) or sampling approaches.

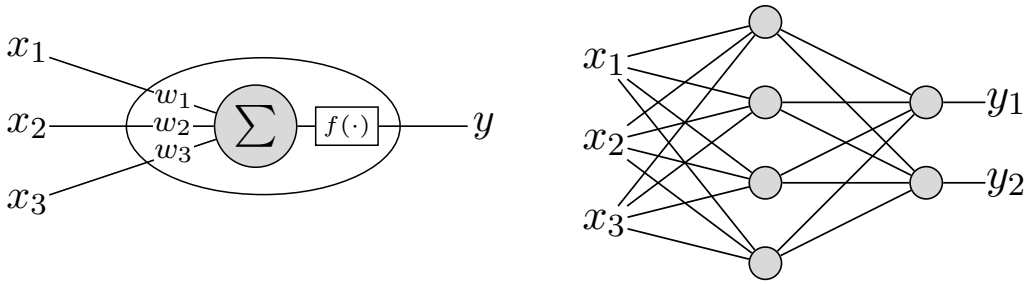
In this thesis, we primarily rely on Gibbs sampling as our inference algorithm of choice. Gibbs sampling is a Markov Chain Monte Carlo (MCMC) method which is capable of producing samples $X^{(1)}, ..., X^{(J)}$ from a distribution $f(x)$ without requiring the ability to directly sample or even know the form of $f(x)$ [16]. This is accomplished by iteratively sampling each variable conditioned on the remaining variables. Pseudo-code for Gibbs sampling is shown in Figure 2.1. This sampler provides a flexible framework for generating samples from a complex distribution, assuming samples can be taken from usually simpler full conditionals. At run-time the mJ samples must be drawn from the full conditionals where m is the number of variables and J is the number of iterations such that efficiency largely depends on the ease of sampling from the full conditionals. As the Gibbs sampler requires some initialization of the model states, many early samples may not reflect the true distribution and are

often discarded as 'burn-in' samples; however, theory guarantees that the stationary distribution of these samples is exactly the joint distribution over the model in the limit.

Armed with these samples from the joint distribution, we can easily compute functionals over the samples. For instance, the mean of the samples approximates the expectation of the joint. In Chapter 3, we use this property to estimate point-wise confidence intervals for our models prediction, producing estimates of error as part of the inference process.

2.2 CONVOLUTIONAL NEURAL NETWORKS

In recent years, the widespread use of Convolutional Neural Networks (CNNs) and other deep networks architectures has led to large performance improvements across a variety of computer vision and natural language processing tasks including image classification [55, 70, 125], object detection [97, 107], face recognition [126], pose estimation [127], semantic segmentation [20, 88], visual question answering [5], and machine translation [9, 132] to name a few. In this chapter, we will review the mechanisms of these models by tracing the evolution of CNN architectures starting from a simple artificial neural classifier and increasing in complexity until we arrive at the contemporary CNN models we use in Chapters 3 and 5. We will also discuss popular training mechanisms for these models and introduce our recent work on producing multiple solutions from structured prediction networks.



(a) Artificial Neuron (b) Feed-Forward Neural Network

Figure 2.2: **(a)** A single artificial neuron is depicted above and is comprised of linear input weights $\mathbf{w} = [w_1, \ldots, w_3]$ and an activation function $f(\cdot)$. Given input $\mathbf{x} = [x_1, \ldots, x_3]$, the output y of the neuron is the result of $f(\mathbf{x}^T\mathbf{w})$. **(b)** Arranging many of these neurons with the outputs of one layer used as input to the next results in a feed-forward neural network. A two layer feed-forward neural network with three inputs and two outputs is shown above.

Artificial Neurons and Feed-Forward Neural Networks

First proposed in 1943 by McCulloch and Pitts [89], the artificial neuron is a biologically inspired class of functions which we illustrate in Figure 2.2a. In these models, the output is computed as a function of the weighted sum of the input. More formally, given a neuron with weights $\mathbf{w} = [w_1, \ldots, w_k]$ and activation function $f(\cdot)$, the output is defined as $y = f(\mathbf{w}^T\mathbf{x})$ for input $\mathbf{x} = [x_1, \ldots, x_k]$. Depending on the form of the activation function $f(\cdot)$, this simple schema can be used to model regression as well as classification tasks. One canonical form of artificial neuron is the *perception* binary classifier introduced in 1957 by Frank Rosenblatt [108] that sets $f(\cdot)$ to be the $sign(\cdot)$ function. Like other single neuron classifiers, the perceptron is limited in its expressibility, modeling only linearly separable tasks effectively (without the use of kernels [56] which transform input non-linearly before training).

Given the interconnected structure of neurons in the brain, one natural remedy to this limited representational power of single artificial neurons is to connect many into

a single neural network. If such a network contains no loops, it is called a *feed-forward neural network*. We present an example feed-forward neural network in Figure 2.2b. In these models, neurons are arranged into layers with the outputs of each layer being used as inputs to the next. These models have been shown to be incredibly powerful universal function approximators given a sufficient number of neurons and layers [57].

The most widely used approach to train feed-forward neural networks is the gradient descent-based *backpropagation* algorithm [136]. Given a differential loss (for example, the $\mathcal{L}_2$ norm for a multi-variate regression), backpropagation computes the gradient of the loss function with respect to all weights in the network, utilizing the chain rule to remove redundant computation by operating in a sequential output-to-input order. Most commonly, backpropagation is used stochastically with small batches of data being processed at a time and the parameters being updated with respect to the loss on those examples only. The batch is processed by the network in what is called the forward pass, and then the gradient information is computed with backpropagation in the backward pass. This approach is simple to implement and computationally efficient to the extent that it has seen nearly complete adoption; however, backpropagation and feed-forward neural networks have historically had difficulties scaling to high-dimension problems.

Modern Convolutional Architectures

Despite their usefulness to other domains, feed-forward neural networks have historically had limited success on computer vision tasks for two major reasons: the exponential scaling of the number of between-layer connections, and the difficulty in training large (and especially deep) neural networks that are capable of modeling

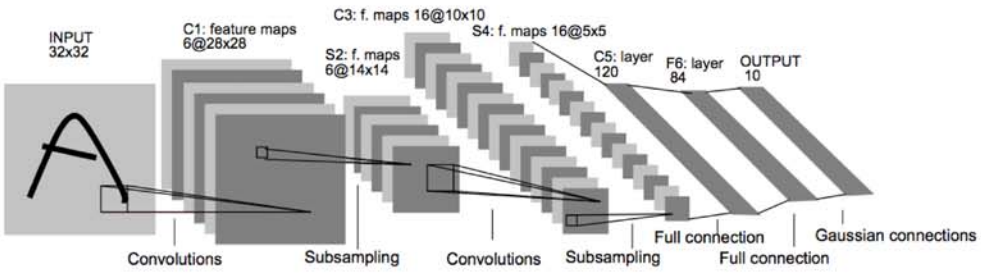


Figure 2.3: The LeNet Convolutional Neural Network architecture [73] consists of alternating convolution and sub-sampling layers followed by multiple fully-connected layers to perform classification. Each convolutional layer consists of multiple filters.

complex visual phenomena. Many techniques have been developed to address these problems and we will focus specifically on the key components of modern Convolutional Neural Networks. CNNs typically consist of multiple convolutional layers, each of which is analogous to a sparsely connected feed-forward neural network and dimensionality-reducing pooling layer. For example, Figure 2.3 shows a simple CNN for handwritten character recognition [73]. Like many other architectures, the network shown consists of alternating convolutional and subsampling layers followed by multiple fully-connected layers (identical to the feed-forward neural networks discussed above) which produce the final classification output as scores for each class. CNN training typically relies on certain forms of activation function and regularization. This section will cover the motivation and function of key components of Convolutional Neural Networks design and effective training.

For standard feed-forward neural networks, the number of weights between two layers is exponential in the number of neurons in each. For high dimensional problems such as image classification, this would result in the number of free parameters of the network scaling exponentially with the number of pixels. To remedy this, Convolutional Neural Networks introduce a spatial weight-sharing schema that re-

sembles convolutional filtering. Figure 2.4 shows the shared weight structure for a one-dimensional Convolutional Neural Network with a filter size of three. Given a set of weights represented as a filter $\mathbf{W}$, activation function $f(\cdot)$, and concatenated outputs of the previous layer $\mathbf{L}_{i-1}$, the output of all neurons in layer L_i can be computed as $f(\mathbf{W} \star \mathbf{L}_{i-1})$ where $f(\cdot)$ is applied to each element of the discrete convolution between $\mathbf{L}_{i-1}$ and $\mathbf{W}$. Since color images are three-dimensional matrices ($R \times C \times 3$), filters for image tasks are also three-dimensional and their outputs are two dimensional matrices.

This design greatly decreases the number of weights to be learned while adding a number of additional positive qualities. Learning relevant image structures or objects using convolutional filters removes (usually irrelevant) spatial dependencies that would exist in a standard fully-connected architecture (i.e. learning the representation of a cat in a traditional feed-forward neural network would require a neuron to represent how a cat looks at every point in the image rather than how a cat looks in general). Additionally, both the forward pass and backward pass for these filters in CNNs can be computed using efficient matrix libraries on powerful GPUs, making training on large datasets with batch computation feasible [22]. Finally, the hierarchical learning of later filters based on the output of previous filters results in learning a pyramid of increasingly complex features [143] that spans the space from simple Gabor filters to full or partial objects such as wheels or eyes. This hierarchical approach has been shown to be advantageous [93]. In practice, a single filter for each layer is insufficient to model the diversity of the visual world so each *convolutional layer* often contains multiple filters such that the concatenation of the outputs remains a three dimensional matrix.

Despite the massive reduction in the number of edge weights achieved through the use of convolution filters, training deep Convolutional Neural Networks (i.e. those with many layers) is still difficult using standard practices from traditional feed-forward neural networks. The primary algorithmic advancements that allow the training and regularization of extremely deep networks are the introduction of the rectified linear unit (ReLU) activation function [48] and the dropout regularization method [121]; however, it is worth mentioning that increasingly large scale datasets [30, 86], the widespread adoption of GPUs for scientific computation, and use of dataset augmentation techniques [18] has also contributed significantly to the rise in popularity and effectiveness of Convolutional Neural Networks.

The ReLU is a piecewise linear function $f(x) = max(0, x)$ which has been shown to be biologically plausible [48]. One the greatest challenges with using backpropagation for deep neural networks is that the magnitude of the corrective signal (i.e. the gradient of the loss) diminishes greatly as it is diluted through many layers. ReLUs both sparsify this dilution process by restricting negative activations, and reduce the decay of the gradient by maintaining a linear relationship between inputs and activated outputs. Compared to standard feed-forward neural network activation functions such as the sigmoid or hyperbolic tangent functions, networks using RELU activations have shown to be significantly easier to train, especially when architectures become increasingly deep (which is shown to improve performance for many problems).

Deep neural networks have a tendency to over-fit to training data due to the large number of parameters and representative power of these models, and standard regularizers such as penalizing the magnitude of weights are not effective at improving

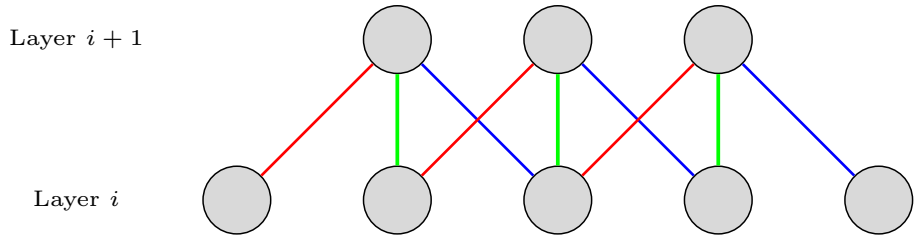


Figure 2.4: Two layers in a one-dimensional Convolutional Neural Network are shown above. Weights are color-coded to indicate weight sharing between neurons (i.e. all red edges share the same weight). Note that weights are shared spatially, similar to a convolutional filter.

generalization. The dropout regularization technique randomly retains or drops individual neurons for each example with a probability α during training. At test time, the outgoing weights from each neuron are scaled by α to approximately perform model averaging. This approximation compares favorably to Monte-Carlo model averaging at substantially reduced cost [121]. In effect, dropout temporarily creates a new, thinned network for each example which forces neuronal activations to behave more independently. Dropout is typically applied to the fully connected layers and not the convolutional layers of CNNs.

We note that Convolutional Neural Networks for classification tasks are typically trained to output a probability distribution over the classes, providing an implicit estimate of confidence; however, methods to produce multiple diverse outputs from CNNs have not been developed. Our ongoing work on this topic is presented in Section 7.1 as future work.

While there have been further advancements beyond those described here (such as batch normalization [60] and skip-connections [55]), which improve the stability of network training, the architectures and training procedures described here form the basis for the modern Convolutional Neural Networks used in this thesis.

CHAPTER 3

ICE LAYER BOUNDARY ESTIMATION FOR GLACIOLOGY

A straight forward annotation task typical of many problems in the physical sciences is to identify simple structures in an image in order to aid larger scientific inquiry and increase the throughput of analysis. These images are often the product of advanced sensing devices such as radar or microscopes and as a result tend to have substantial image noise. Common in-domain approaches to these problems typically involve pipelines of filtering that rely on fine-tuned thresholds. While these methods are successful enough to be somewhat useful, modern computer vision approaches could allow more principled solutions. In this chapter, we develop a computer vision system to solve one of these problems in a holistic manner, simultaneously estimating solutions and response confidence. We consider one particular domain of radar layer-finding, but the same technique could be applied to many other structured segmentations problems.

3.1 LAYER-FINDING IN RADAR SOUNDING IMAGES

Observing the structure and dynamics of polar ice sheets is critical for developing accurate climate models. Glaciologists have traditionally had to drill ice cores in order to observe the subterranean structure of an ice sheet, which is a slow and labor-

intensive process. Fortunately, ground-penetrating radar systems have matured in the last few years to allow surveying large areas of ice from aerial and ground vehicles with minimal human intervention [3]; however, identifying the depth of the ice sheet and the topography of the underlying bedrock is often a manual task requiring tremendous human effort.

Figure 3.1 (right) presents an example of an echogram produced by the multi-channel coherent radar depth sounder system of the Center for Remote Sensing of Ice Sheets (CReSIS) [3]. This echogram is a virtual cross-section of the ice, where the horizontal axis is distance along a flight path of the aircraft-based radar system and the vertical axis is vertical distance from the plane (i.e. depth). The echograms capture the radar signal's scattering properties and can be used to estimate an ice sheet's depth and the topography of the bedrock beneath the ice (the dark erratic line near the middle of the figure). These observations are used in models to forecast ice sheet behavior.

In this chapter, we pose identifying the surface and bedrock layers in these echogram images as an inference problem on a probabilistic graphical model [74], building on the approach introduced by Crandall *et al.* [26]. Our probabilistic framework allows for multiple sources of evidence to be integrated to determine layer boundary estimates. We introduce several important contributions to improve both the accuracy and utility of the approach. Our technical innovation uses Gibbs sampling to perform inference instead of the dynamic programming-based solver of [26]. This allows us to strengthen the underlying model to solve for both layer boundaries simultaneously, yielding automatic layer detection results that are significantly better than prior approaches. Moreover, the Gibbs sampler produces explicit confidence intervals, thus

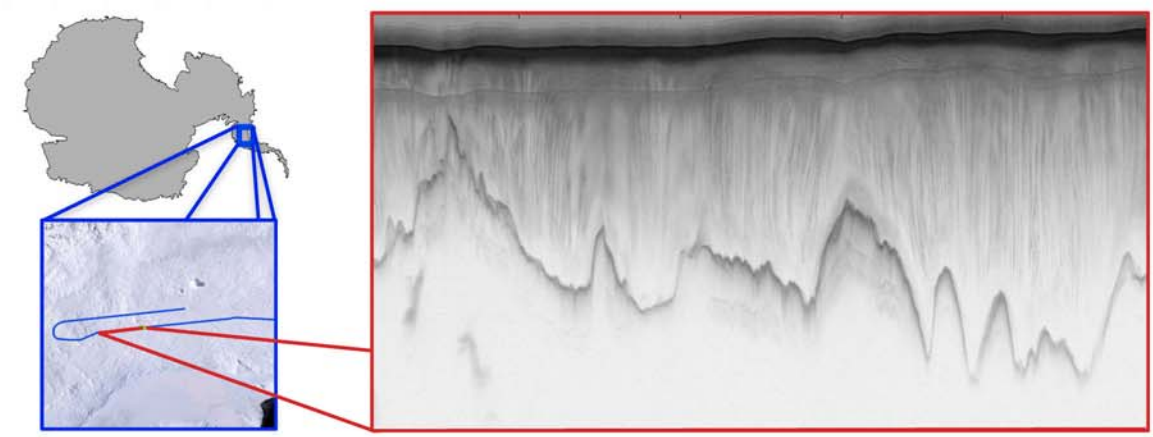


Figure 3.1: Using an aircraft-mounted radar system, glaciologists capture virtual cross-sections of the ice sheets called echograms. A sample echogram (red) is shown here where the horizontal axis is distance along a flight path (blue) and the vertical axis is vertical distance (depth) from the aircraft. The surface (very dark line near the top) and bedrock (dark erratic line near the middle) layer boundaries are visible along with weaker signals from secondary reflections and the contiguous layers of ice between the two.

estimating bands of uncertainty in the layer boundary locations. Since noise and ambiguity in radar echograms are inevitable, we believe this ability to estimate confidence is crucial for downstream applications (e.g. when used in glaciological models).

Several other semi-automated and automated methods for identifying subsurface features of ice have been introduced previously in the literature. The most related papers to our work have focused on automated detection in terrestrial echograms. Freeman et al. [43] address the problem of near-surface layer detection in Martian ice layers from orbital radar data. Using the observation that near-surface layers tend to be parallel, Freeman et al. [43] identify the easily detected top-layer and estimate a global transformation to linearize the layers, afterwards applying simple filtering and morphological operations to extract layer boundaries. Similarly, Gifford et al. [45] model Martian ice layers as linear elements and use a Steger filter [122] to identify these structures. These methods rely on filtering and hard thresholds to identify lay-

ers, which can limit applicability due to the need to fine-tune these parameters. Ilisei et al. [59] developed a two-phase technique to exploit the properties of a radar signal to generate a statistical map and then apply a segmentation algorithm. Although our application focuses on detecting bedrock and surface layers, other studies use similar techniques to identify internal layers in radar imagery [**?**, 34, 92, 99, 118].

3.2 A MARKOV CHAIN MONTE CARLO APPROACH

An echogram is a 2D matrix which represents the scattering properties of the subsurface at each along-track coordinate of the radar platform. Our task is to find two key features in these echograms: the ice surface boundary (the strong reflector near the top) and the bedrock boundary (the dark reflector near the middle of the image).

We want to estimate the location of layer boundaries by determining their paths through the image. Assume that an echogram has k layer boundaries (with k=2 in our case). Given an echogram I of dimension $M \times N$, we wish to estimate unknown variables $L = \{L_1, ..., L_k\}$, where $L_i = \{l_{i1}, ..., l_{iN}\}$ and l_{ij} denotes the row coordinate of layer i in column j.

We take advantage of the structure of this problem by posing it as a grid-shaped probabilistic graphical model. In this framework, we are interested in estimating $P(L_1, ..., L_k|I)$, the joint probability over the layer boundaries given the echogram. Given this distribution, we could apply any function over the set of possible layers including computing moments, drawing multiple highly likely solutions, and providing estimates of variance. Unfortunately, this distribution has an alarming dimension of order $O\left(M^{kN}\right)$ such that computation and storage are intractable. To address this problem, we make two assumptions,

(1) image characteristics are determined by local layer boundaries, and

(2) variables in L exhibit a Markov property with respect to their local neighbors, which greatly simplify the model and enable finding an efficient solution.

Under Bayes' Law we can decompose the joint distribution into a product of two intuitive distributions as

$$P(L_1, ..., L_k|I) \propto P(I|L_1, ..., L_k)P(L_1, ..., L_k). \tag{3.1}$$

The image likelihood term $P(I|L_1, ..., L_k)$ captures how well the image data can be explained by a set of layers and the layer likelihood $P(L_1, ..., L_k)$ captures prior knowledge about the boundaries, such as that they are smooth and do not intersect.

Our first assumption implies that image pixels not near the layer boundaries are generated by noise, so we need only model pixels near boundaries. Under this assumption, we can factor $P(I|L_1, ..., L_k)$ into a product over layer positions,

$$P(I|L_1, ...L_k) = \prod_{i=1}^{k} \prod_{j=1}^{n} P(I|l_{i,j}). \tag{3.2}$$

Since boundaries are dark edges, we model the right hand term as a product of gradient magnitude and image intensity,

$$P(I|l_{i,j}) \propto |\nabla I(l_{i,j}, j)| \cdot (1 - I(l_{i,j}, j)), \tag{3.3}$$

where $|\nabla I(x, y)|$ is the gradient magnitude at coordinate (x, y) of the image and pixel values have been scaled such that $I(x, y) \in [0, 1]$. We approximate gradient magnitude through finite differences on a 5×5 window.

The second assumption simplifies the problem by imbuing the graphical model with the property that each node $l_{i,j}$ is independent of the remaining variables in L given its immediate neighbors in the graph. Under this assumption, we have

$$P(L_1, ..., L_k) \propto \prod_{i=1}^{k} \prod_{j=1}^{n} P(l_{i,j}|N(l_{i,j})), \tag{3.4}$$

where $N(l_{i,j})$ is the set of directly connected nodes in the graph (i.e. $N(l_{i,j}) = \{l_{a,b} \,|\, 1 = |a - i| \text{ and } 1 = |b - j|\}$). We define $P(l_{i,j}|N(l_{i,j}))$ as the product of independent vertical and horizontal components. Along the same layer, the l_i's are encouraged to be smooth by a zero-mean Gaussian which is truncated to zero outside a fixed interval, and distance between layers is penalized as a step function to encourage layers not to overlap.

Figure 3.2 shows a graphical representation of our two-layer model for an echogram of width N. For each layer, each column n of the image is associated with random variables $l_{1,n}$ and $l_{2,n}$ corresponding to the positions of each layer in that column. These variables are connected via dependency relationships to the image I and their immediate neighbors (both along a layer and between layers). In Figure 3.2, the node representing $l_{1,1}$ is highlighted and the conditional distributions with respect to $l_{1,1}$ are written.

This model is similar to [26] but with important improvements. In [26], the vertical pairwise potentials are zero at and above intersection points and uniform elsewhere. But it is common in this data to see radar reflections of the surface layer directly below the actual surface, so we add a fixed-width low probability region directly below them to reduce false bedrock detections on these reflections. Perhaps more

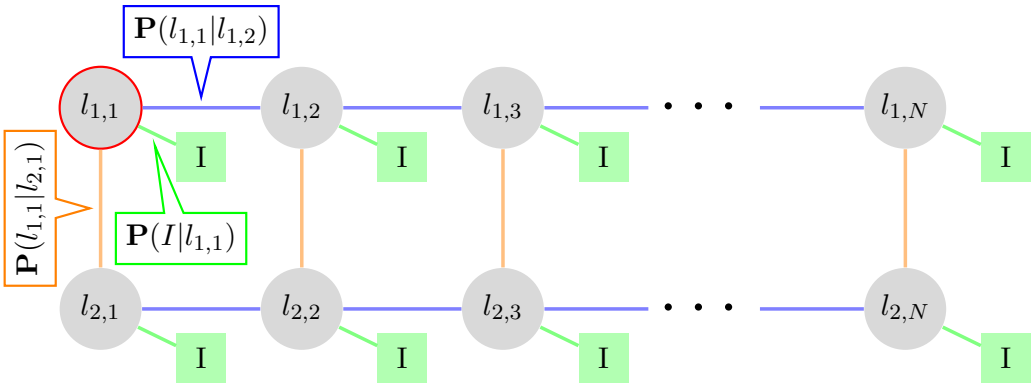


Figure 3.2: Graphical depiction of our model for two layers, where each row represents a layer through the image. **Along-layer links** enforce pixel-wise smoothness of the layers, the **between-layer links** model repulsiveness between layers, and the **image-likelihood** links condition layer locations on local image characteristics. These conditional distributions are written out for the $l_{1,1}$ node outlined in red.

importantly, the model in [26] breaks these vertical constraints in order to simplify inference by greedily solving each layer conditioned on the previous one. We avoid doing this, and our experiments show that this holistic inference approach offers substantial improvements in accuracy.

Statistical inference

The model defined by equations (3.1), (3.2), and (3.4) is a first-order Markov Random Field. Unfortunately, finding the values of L that maximizes equation (3.1) is NP-hard in the general case [68]. Rather than trying to solve this as an optimization problem, we instead attempt to estimate functionals of the full joint distribution via Gibbs sampling (a Markov Chain Monte Carlo technique discussed in Section 2.1).

It can be shown via Bayes Law and the independence assumptions in equations (3.2) and (3.4) that the full conditionals for each l_{ij} can be computed easily

as

$$P(l_{ij}|L, I) = P(l_{ij}|I, N(l_{ij})) = P(I|l_{ij})P(l_{ij}|N(l_{ij})). \tag{3.5}$$

As the domain of l_{ij} is discrete and finite, sampling from this conditional is simple. As an additional optimization, we make use of the vertical and horizontal thresholds in the layer conditionals to sparsify the computation of $P(l_{ij}|N(l_{ij}))$, as most entries are known to be near zero. We apply the Gibbs sampler to generate a sequence of samples $L^{(B)}, ..., L^{(J)}$ where B is a burn-in time during which samples are discarded. This is a common practice with MCMC methods to reduce sensitivity to initial values. To predict the layer locations we take the mean of our $J-B$ samples, which approximates the expectation of the joint distribution. We also utilize these samples to produce point-wise 95% confidence intervals by taking the 0.025 and 0.975 quantiles of these approximate marginal distributions.

3.3 LAYER FINDING PERFORMANCE

We tested our layer-finding approach using a set of 826 publicly-available radar echograms from the 2009 NASA Operation Ice Bridge program, collected with the airborne Multichannel Coherent Radar Depth Sounder system [3]. The images have 'ground truth' labels produced by human annotators; however, these labellings are often quite noisy. For instance, sometimes the annotators could not find a reasonable layer boundary and did not mark anything at all. To decouple the error in the ground truth from the method evaluation, we removed images with obviously incomplete ground truth (including those with partially defined layers and those with fewer than two layer boundaries). We ran our method on the remaining 560 images.

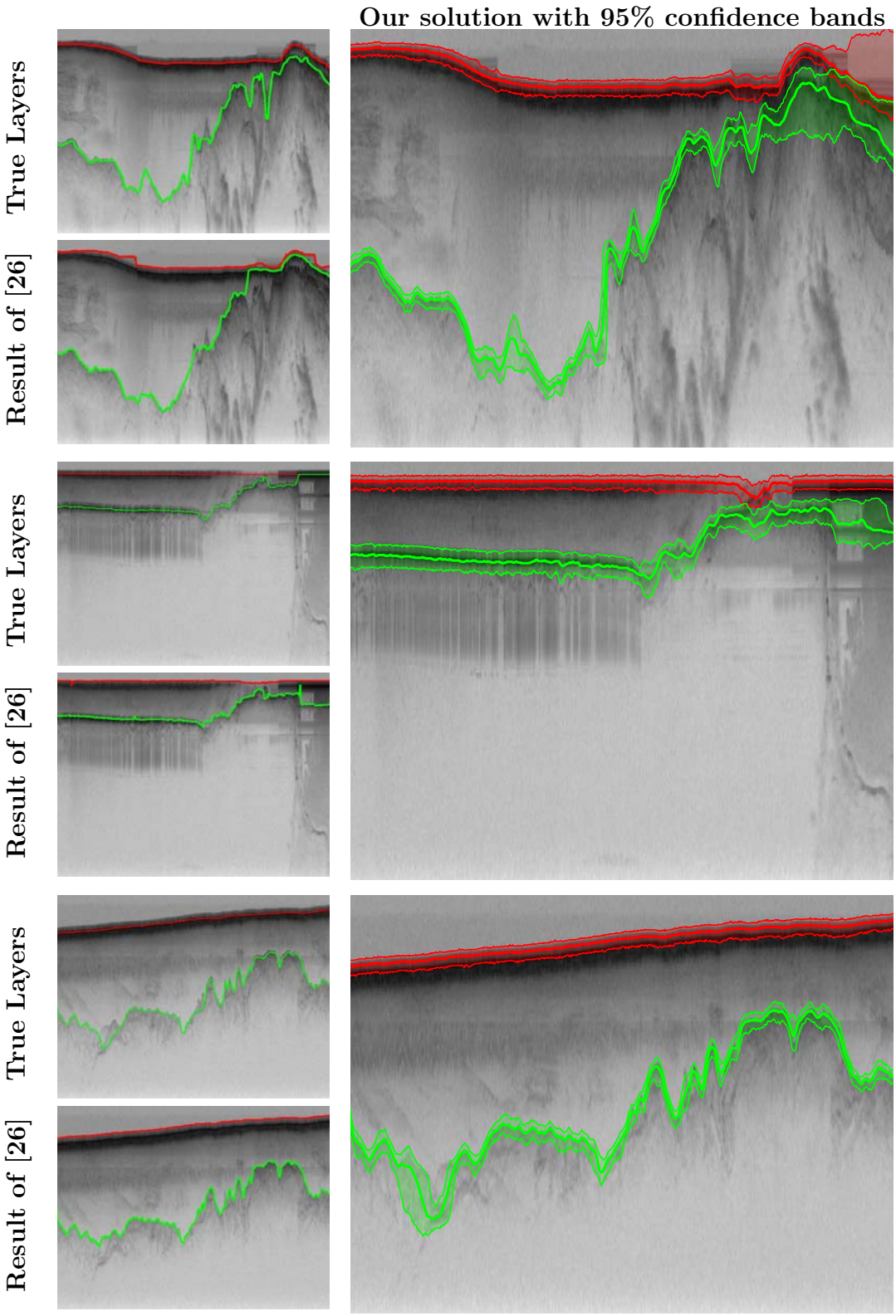


Figure 3.3: Results on three sample echograms. Each pane includes the hand-labeled image *(top-left)*, the output of [26] *(bottom-left)*, and our output with 95% confidence intervals shaded *(right)*. Best viewed in color.

For each image, we collected 10,000 samples (after a burn-in period of B=20,000 iterations to overcome initialization) and took the mean as our output.

We measure accuracy by viewing ground truth and estimated layer boundaries as 1-D signals and computing the mean absolute errors across images. We compare with [26] as it is most similar to our technique. For this experiment we find we outperform the method of [26] significantly, reducing the mean absolute error by 44.3% for surface boundaries and 48.3% for bedrock. Figure 3.3 shows results on three sample echograms, presenting the output of our technique (including the estimated confidence interval) as well as the ground truth and the baseline technique of [26]. Our method not only tends to find better layers but also captures confusion in an interpretable way (e.g. the closeness between layers in the first example coupled with the artifacts in the top-right corner causes our model to be uncertain about that area.). We find that 94.7% of the surface boundary points and 78.1% of the bedrock boundary points are within their respective confidence intervals.

3.4 CONCLUSION AND FUTURE WORK

In this chapter, we proposed an automated approach to estimate the locations of bedrock and surface layers in multichannel coherent radar imagery and demonstrated its effectiveness on a real-world dataset against the state-of-the-art. Our method not only outperforms existing approaches, but also provides point-wise confidence intervals for identified layers. This is important because ice sheet depth maps computed from radar images are used in downstream glaciology models to predict phenomena of global concern including the effect of global warming on sea level rise. Given the seriousness of these topics, it is important for automated approaches not only to have

low error in the general case, but also to be able to identify when and where errors occur through robust confidence estimation. In future work, extending this model to a Reversible-Jump MCMC [49] framework such that the number and form of layers are both found via the model would allow broader application, such as to the task of annotating annual snow-accumulation layers in near-surface radar imagery.

This simple annotation task is representative of a larger class of visual analysis problems in the physical sciences that require identifying simple structures in images. Other examples include tracking cells [21] in biology and blood vessel detection in inner-eye images to automate disease diagnosis [38]. Typically these problems have been addressed by domain specialists using filtering approaches that are often years behind analogous work in computer vision; however, due to the simplicity of the structures to be identified, these naïve approaches often perform well enough to be useful. In the following chapter, we will show an annotation task that is not amicable to simple filter based approaches due to the complexity of the structures to be identified, and we present powerful computer vision models to provide more robust annotations.

CHAPTER 4

DETECTING HANDS FOR DEVELOPMENTAL PSYCHOLOGY

Many academic disciplines are interested in studying the interaction of organisms with each other and with the world. Traditionally, research in these domains has been dominated by the careful manual observation and recording of the behavior of subjects (visual or otherwise) both in constrained settings and in the wild. With the increasing ubiquity of video recording devices, data collection for many of these studies can be at least partially automated; however, coding of the recorded video data is still typically a manual procedure. Unlike the examples in the previous chapter, the structures and events to be annotated in this imagery are substantially more complex, often being highly-deformable body parts. As the subjects move around, the environment changes in imaging conditions (sudden change of illumination exiting a house) and frequent occlusions increase the difficulty of annotation. Despite this increase in complexity, humans are still highly accurate annotators, but the time consumed in manual annotation is prohibitive for large scale analysis. In this chapter, we will take hand detection in egocentric video from child development studies as an example for the potential of powerful computer vision models to provide quality annotations in even these adverse conditions.

4.1 EGOCENTRIC HAND DETECTION

Head-mounted cameras capture video that is fundamentally different from traditional hand-held consumer cameras. Instead of capturing posed and intentionally framed moments, egocentric (head-mounted) cameras continuously record an approximation of a person's field of view during everyday life. This technology is empowering novel applications in cognitive research, for instance by recording fine-grained information about people's activities and interactions [10, 42]. However, these applications create huge volumes of video, so automated techniques are needed to process and understand them.

This chapter is motivated by recent psychology experiments conducted by the Computational Cognition and Learning Laboratory [17] at Indiana University as part of a multi-sensory approach to study embodied attention and statistical learning in toddlers. These experiments use egocentric cameras to study how young children and adults interact with one another and how children coordinate their hands, head turns, and gaze in order to manipulate objects [10, 41, 42, 106]. In these experiments, a parent and child play with toys on a table and frequently point to, reach for, and exchange toys with their hands. The child's view is extremely dynamic: the hands of both the child and parent frequently disappear and reappear or are partly occluded. Manually labeling hand positions in these large-scale datasets is slow and tedious, so a main motivation of our work is to develop a technique that can perform the labeling automatically. In the line of work presented in this chapter, we develop methods for tracking hands in egocentric video to enable automatic hand annotation for both laboratory and unconstrained egocentric videos.

General hand detection in egocentric video also has applications outside of studying behavior. Hands are perhaps the most frequent objects in egocentric video and are arguably also the most important, since they are the primary way that humans physically interact with the world. In fact, much work in egocentric activity recognition assumes that activities can be characterized by the in-hand manipulation of certain objects [37,103]. Other work on egocentric hand detection is motivated by the idea that hands are important for understanding complex object manipulations, gestures, and motor skills [82,83,115,142]. Much of the existing hand detection literature assumes that only the camera wearer's hands are visible in the scene, even though real-world egocentric video includes frequent interactions with other people [36]. Recognizing gestures, handled objects, and activities in practice will thus require distinguishing the camera owner's hands from others that occur in the scene.

In this chapter, we present two pieces of related work. In the first, we use strong spatiotemporal priors and probabilistic inference to track hands in carefully collected laboratory data [75]. In the second, we learn to disambiguate hands in less constrained data using powerful appearance models guided by spatial biases [12].

4.2 A PROBABILISTIC APPROACH FOR EGOCENTRIC VIDEOS IN CONSTRAINED SETTINGS

Given an egocentric video of the interaction between two people, we would like to estimate the locations of the observer's hands and the other person's hands. This task is difficult because these parts frequently enter and leave the frame and there is erratic camera motion caused by head motion of the camera wearer.

More formally, given a video sequence of n frames, each with $r \times c$ pixels, our

goal is to estimate the position of each of a set of parts $\mathcal{P}$ in each frame. In this section, we consider five parts in particular, $\mathcal{P} = \{\mathsf{yh}, \mathsf{yl}, \mathsf{yr}, \mathsf{ml}, \mathsf{mr}\}$, corresponding to the other person's head, hands ('your left/right') and the camera wearer's hands ('my left/right'), respectively. Although our framework is general enough to handle any set of parts (e.g. in the case of more than two interacting people), for clarity we discuss these five specific parts in particular.

We denote the latent 2-D projected position of part $p \in \mathcal{P}$ in frame i as L_p^i and define L^i to be the full configuration of parts within the frame, $L^i = \{L_p^i\}_{p \in \mathcal{P}}$. Because of the dynamic nature of egocentric video, hands often enter and exit the frame, due both to motion of the hands and motion of the head-mounted camera. To address the possible absence of any given part, we augment the domain of L_p^i with an additional state $\emptyset$ indicating that the part is not visible in the frame, i.e. $L_p^i \in \{\emptyset\} \cup ([1, r] \times [1, c])$.

In addition to part position, we also explicitly model global motion caused by head movements by introducing random variables $G = (G^1, \ldots, G^{n-1})$, where G^i is an estimate of the two dimensional global coordinate shift between frame i and frame $i + 1$. In this way, we assume the world has uniform depth such that a change of viewing angle would have the same effect on all points in the 2-D projection of the environment. This assumption is reasonable given that the distances involved in a paired interaction are relatively constrained.

4.2.1 MODELING HANDS IN EGOCENTRIC VIDEO.

We use a graphical model framework to model and estimate the locations of parts across all video frames jointly. We design our model to include known biases in

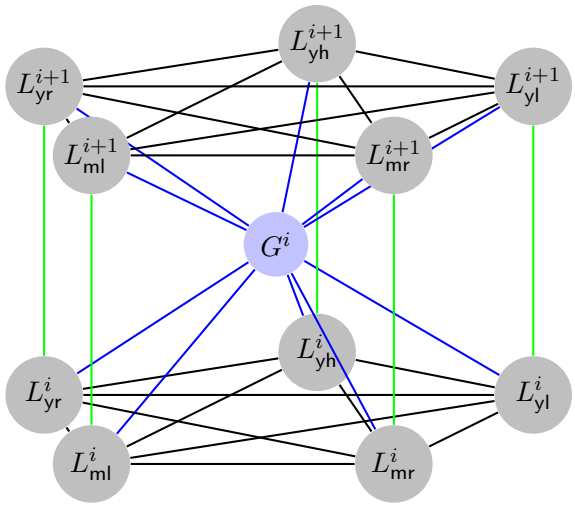


Figure 4.1: Graphical depiction of our model for two frames, where the bottom five nodes represent the locations of the head and hands in one frame, and the top five nodes represent the locations in the next frame. **Between-frame links** enforce temporal smoothness, **shift links** model global shifts in the field of view, and **in-frame links** constrain the spatial configuration of the body parts.

egocentric video and hand positions. Our model is visualized in Figure 4.1 for a two-frame video. The connections within a frame (in black) form a complete graph over the five part nodes and capture the pairwise correlations between spatial locations of the parts. The green edges between each part and its corresponding variable in the next frame enforce temporal smoothness. Finally, the global shift variable is influenced by all pairs of corresponding parts such that a similar motion in all part pairs is likely to indicate a global shift. More completely our model consists of four components:

1. **In-Frame Constraints** – In a given frame, the positions of hands show a strong correlation (e.g. my left hand is typically to the left of my right hand). We model these correlations as $P(L^i_p|L^i_q)$, defined as Gaussian distributions over relative part positions.
2. **Between-Frame Constraints** – Between adjacent frames, hands are unlikely to

change positions drastically except due to sudden camera motion. We encode this assumption as between-frame constraints $P(I^i, I^{i+1}|G^i)$ that encourage smooth tracking trajectories. These distributions are defined as Gaussian distributions over relative part locations between frames with mean G^i. G^i itself is estimated with respect to sequential frames as $P(I^i, I^{i+1}|G^i)$ defined as a Gaussian centered around the average pixel displacement between frames I^i and I^{i+1} derived from optical flow [40].

3. **Appearance Models** – Weak appearance models $P(I^i|L_p^i)$ are incorporated to identify image regions likely to represent each part. For head priors we use detections of the simple Viola-Jones face detector [133]. For hands, we consider two factors. First we compute the likelihood of a pixel being skin from a learned data-driven model. Specifically, we estimate the color distribution of skin in the HSV color space using ground truth segmentations and use this non-parametric distribution to estimate skin likelihood for a given pixel. Additionally, as arms are typically visible for the activity partner, we use simple edge cues to estimate whether each skin blob is attached to an arm and adjust the probability of that blob belonging to the activity partner with probabilities estimated from the training data.
4. **Absolute Spatial Priors** – Finally, hands have a tendency to occur in certain locations within an egocentric video frame due largely to anatomy and grasping behaviors. We model these biases as absolute spatial priors $P(L_p^i)$, defined as Gaussian distributions on absolute spatial location.

Placing these (soft) constraints into an undirected graphical model yields a joint distribution over all the latent variables $L = (L^1, ..., L^n)$ and G, conditioned on the

image sequence $I = (I^1, ..., I^n)$. The complete model can be written as

$$P(L, G|I) \propto \prod_{i=1}^{n} \left[P(I^i, I^{i+1}|G^i) \prod_{(p,q)\in\mathcal{E}} P(L_p^i|L_q^i) \prod_{p\in\mathcal{P}} P(I^i|L_p^i) P(L_p^{i+1}|L_p^i, G^i) P(L_p^i) \right], \tag{4.1}$$

where I=$(I^1, ..., I^n)$ is the image sequence and $\mathcal{E} \subset \mathcal{P}^2$ is the set of undirected edges in the complete graph over $\mathcal{P}$.

As all of our distributions are Gaussian, producing the probability of any given in-frame configuration of parts is trivial; however, a major complication in this problem is the need to model the possibility of a body part being out of the field of view. As an example, we can start by considering a 2-D isotropic Gaussian function (such as those defined over relative part positions),

$$f_{\mu,\Sigma}(x, y) = \mathcal{N}(x; \mu_1, \Sigma_{11})\mathcal{N}(y; \mu_2, \Sigma_{22}),$$

parameterized by $\mu = [\mu_1\ \mu_2]^T$ and $\Sigma = \text{diag}(\Sigma_{11}, \Sigma_{22})$. If this function represents a probability distribution over the location of a given part, then calculating the probability that the part is 'out' of a frame is equal to one minus the probability of being within the frame, $1 - F_{\mu,\Sigma}([1, c], [1, r])$, with

$$\begin{aligned} F_{\mu,\Sigma}\left([x_1, x_2], [y_1, y_2]\right) &= \int_{x_1}^{x_2}\int_{y_1}^{y_2} f_{\mu,\Sigma}(x, y)\, dy\, dx \\ &= \left[\Phi(x_2; \mu_1, \Sigma_{11}) - \Phi(x_1; \mu_1, \Sigma_{11})\right] * \left[\Phi(y_2; \mu_2, \Sigma_{22}) - \Phi(y_1; \mu_2, \Sigma_{22})\right], \end{aligned} \tag{4.2}$$

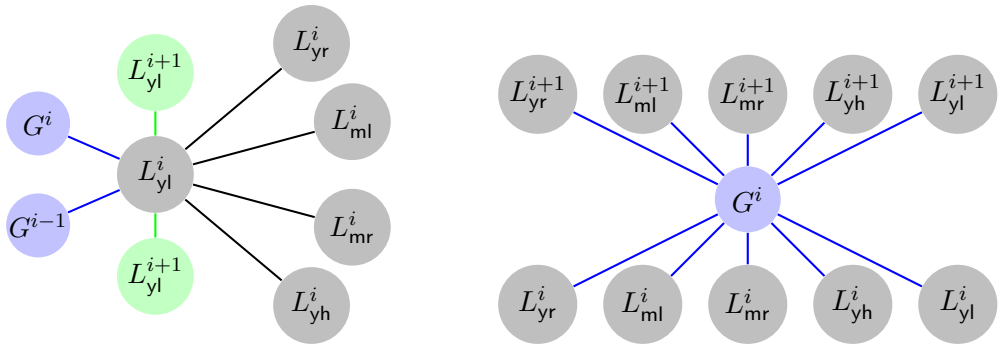


Figure 4.2: Components of the full conditional in our five-part case, for *(left)* part node L^i_{yl}, and *(right)* shift node G^i.

where $\Phi(\cdot)$ is the normal cumulative density function. We employ this technique to compute the probability of the out state $\emptyset$ using our models.

Inference

We can solve the part-tracking problem for an entire video I by maximizing Equation 4.1. Unfortunately, finding the global maximum is intractable. We thus settle for approximate inference using Gibbs sampling (see Section 2.1). As all of our model factorizes well and is made of relatively easily manipulated distributions, we can derive the full conditionals for each variable. Figure 4.2 shows an example of the dependencies for full conditionals for a part node and a shift node under our independence assumptions.

4.2.2 EVALUATION ON LABORATORY DATA

We evaluate our approach on video recorded in the Computational Cognition and Learning Laboratory [17] at Indiana University as part of a multi-sensory approach to study embodied attention and statistical learning in toddlers. In these experiments, a child and parent sit at a table and face one another with each wearing head-mounted

cameras. Parents are told to engage their child with the three toys on the table and interact as naturally as possible. To try to limit distractions, the walls of the lab are colored white, and participants wear white coats. We use the video from the child's camera such that the other person in view is always the adult. The video is captured at 30Hz with 480×720 pixel resolution. We use video data from five parent-child dyads over four play sessions. The trials had an average length of 1.5 minutes, leading to a total of 20 videos containing 56,535 frames (about 31 minutes) of social interaction from the children's perspective.

To evaluate our approach, we manually annotate part bounding boxes for 2,400 random frames from the laboratory dataset or about one frame for every second. For each frame, our system estimates the location of each of the five body parts, by either providing a coordinate or indicating that it is outside the frame. We evaluate the accuracy of our method as the fraction of true positives (i.e. cases where we correctly estimate a position inside the ground-truth bounding box) and true negatives (i.e. where we correctly predict the part to be outside the frame). We also evaluate the percentage of "perfect" frames, those in which all five parts are predicted correctly.

We are particularly interested in errors made when disambiguating the observer's hands from the partner's hands. We consider a ground-truth hand to be a disambiguation error if it is either unlabeled, labeled as the wrong person's hand, or is marked with multiple labels of different people (falsely estimating that hands overlap). The disambiguation error rate is the fraction of incorrectly disambiguated hands over all frames.

We first present qualitative results on the lab dataset. Figure 4.3 shows some sample frames, where rectangles depict the ground-truth bounding boxes and dots

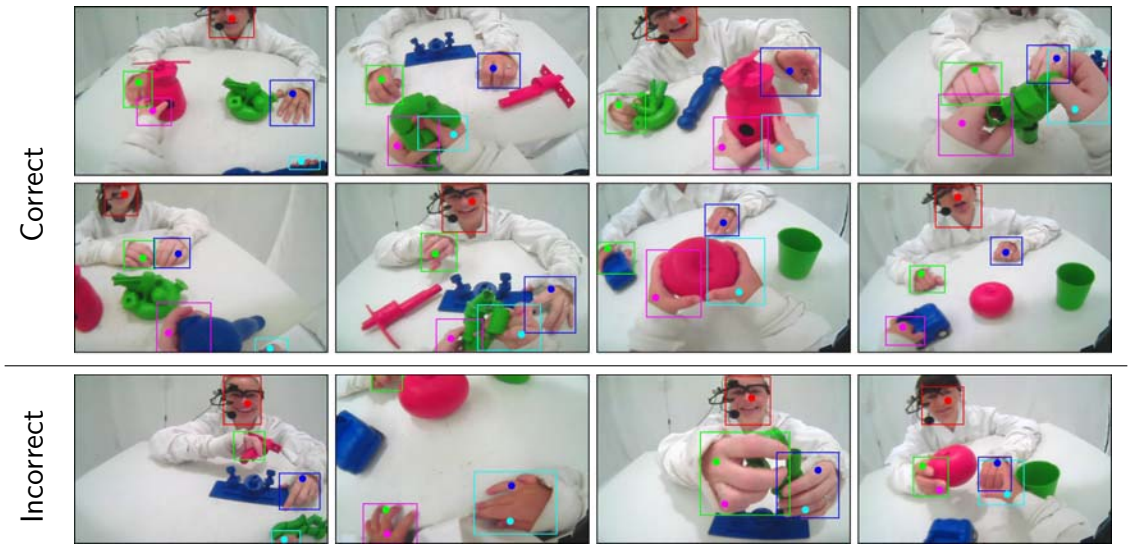


Figure 4.3: Sample frames from our results, with rectangles showing ground truth bounding boxes and dots showing predicted part positions. (red = your head, blue = your left hand, green = your right hand, magenta = my left hand, cyan = my right hand). The first two rows show our robustness with respect to partial occlusions and changes in hand configurations, while the bottom row shows failure cases.

mark our predicted position. Part identities are represented by color, so that dots inside boxes of the same color indicate correct estimates. The first two rows show perfect frames, while the last row shows errors. Common failures include incorrectly estimating a hand to be out of frame (e.g. leftmost image) or falsely estimating overlapping hands. This can be caused by hands that are closer to the observer than expected and thus too big (e.g. in the middle two images), or because one hand is farther away from the other than usual (e.g. wrong prediction for 'my left hand' in the right image).

We present detailed quantitative results in Table 4.1. Our overall detection accuracy across the five subjects of the lab data set is 68.4%. The technique generalizes well between different subjects, as evidenced by a low standard deviation across videos ($\sigma = 3.0$). Accuracies between different hands are also fairly stable, ranging

	Overall	**Observer**		**Partner**				**% Perfect**	**Disambiguation**
	Accuracy	right	left	right	left	head	head^{VJ}	**Frames**	**Error Rate**
Subject 1	64.1	50.3	60.2	68.0	54.2	87.7	86.2	14.8	37.8
Subject 2	72.6	78.5	63.3	63.8	79.7	77.5	55.5	22.8	27.4
Subject 3	70.1	64.2	66.7	60.5	68.8	90.0	85.5	24.7	34.5
Subject 4	67.3	88.0	54.7	59.5	59.3	75.2	66.0	15.5	33.1
Subject 5	68.1	72.5	61.0	66.2	60.5	80.2	69.0	17.7	30.5
Average	68.4	70.7	61.2	63.6	64.5	82.1	72.4	19.1	32.7

Table 4.1: Detection accuracies of our approach as well as a breakdown into different hands. We also compare our head-detection accuracy with the accuracy of the raw Viola-Jones detector (head^{VJ}). The second to last column shows the percentage of frames in which all five predictions were correct and the last column shows the error we made when differentiating the observer's hands and the partner's hands.

from 61.2% for "my left hand" to 70.7% for "my right hand." Overall, our approach perfectly predicts 19.1% of frames, and for Subject 3 achieves a 24.7% perfect detection rate. Although our main purpose is to detect hands, the temporal and spatial constraints in our model also improves face detection by 10 percentage points over the raw Viola-Jones detector (column head^{VJ})used as a prior.

Comparing to Baselines

We compared our model to three baselines of increasing complexity. First, we tried a random predictor which places each part marker randomly by first sampling a binary variable to decide whether the part is in the frame, and if so, assigning it to a random position. Second, we added the skin likelihood by repeating the same process but limiting the space of possible positions to be in skin patches. Finally, we build a more sensible baseline, clustering the detected skin pixels into hand-sized patches using Mean Shift [25]. Then, we greedily assign each part the position of the closest cluster centroid based on distance between centroid and part-wise absolute spatial priors.

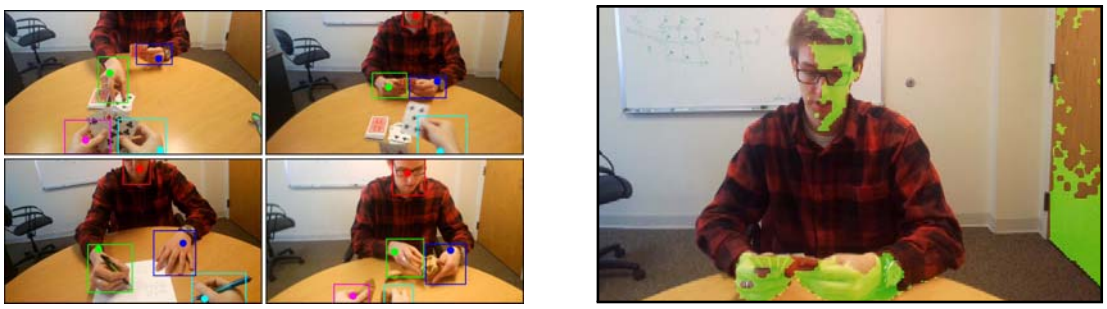
Table 4.2 shows the results of these experiments. Naturally the two random baselines perform poorly. The mean-shift based method performs better than these; however, it is still over 10 percentage-points less accurate than our approach. We also tested a simplified version of our model composed of only the spatial priors. This achieved 59.1% accuracy, comparable to the mean-shift based baseline which similarly does not impose temporal or relative spatial constraints.

Method	**Overall Accuracy**	**% Perfect Frames**	**Disambiguation Error Rate**
random	17.0	0.1	95.1
random (skin)	27.3	4.3	72.0
skin clusters	58.1	14.4	36.0
ours (likelihood + spatial prior)	59.1	9.2	44.5
our method (full)	**68.4**	**19.1**	**32.7**

Table 4.2: Comparison of our model's results to baselines, in terms of overall accuracy, percentage of perfect frames, and hand disambiguation error rate.

4.2.3 GENERALIZING TO NATURALISTIC VIDEOS

Our method relies on relatively weak part appearance models which work well in our constrained laboratory settings. To evaluate how well the model performs on more naturalistic video lacking tight constraints on background and participant attire, we recorded an additional small dataset. We used Google Glass to record a small set of egocentric videos containing two adults engaged in three kinds of social interactions: playing cards, playing tic-tac-toe, and solving a 3-D puzzle. Each video is 90 seconds long, for a total of 4.5 minutes (8,100 frames), and was captured at 30Hz with a resolution of 1280×720. We again manually annotated hand bound boxes for sample



(a) Example naturalistic frames (b) Skin detection performs poorly.

Figure 4.4: **(a)** Sample results for naturalistic video, in which two people played cards (top), tic-tac-toe, and puzzles (bottom), while one wore Google Glass. (See Fig. 4.3 caption for color legend.) **(b)** Skin detection in naturalistic environment performs poorly.

frames.

As expected, accuracy was lower for the naturalistic videos at 50.7% overall. Some example frames are shown in Figure 4.4a. We attribute this drop in accuracy largely to the weakness of our appearance models. As shown in Figure 4.4b, our skin color model fails in more natural environments, firing on near-skin tone colors (e.g. the wooden door). In the lab videos, 97% of detected skin pixels are located inside ground-truth bounding boxes; however, this figure drops to only 70% for the natural videos. Interestingly, we can still retain a relatively low disambiguation error rate in the naturalistic videos (35.6% versus 32.7%), showing that our model can compensate for noisy likelihoods.

4.3 DEEP LEARNING FOR GENERAL HAND DETECTION

While laboratory settings are useful for studying human behavior under tightly controlled conditions, observation in natural settings provide additional, more ecologically valid insights. Hand tracking in unconstrained environments is more challenging

than in laboratory data due to increased background clutter, a wider range of illuminations, and more diverse interactions. We combine the spatial biases of egocentric video discussed in the previous section with powerful learned appearance models to accurately detect hands in natural videos. To assess the performance of our method, we collected a large dataset of paired participants engaging in tabletop activities within naturalistic environments, while wearing egocentric cameras.

4.3.1 DESIGNING A STATE-OF-THE-ART HAND DETECTOR

In principle, finding hands in first-person video frames is simply an instantiation of one particular object detection task, for which we could apply any general object detection algorithm. But in practice, detecting hands requires some special considerations. Hands are highly flexible objects whose appearance and position can vary dramatically, but nonetheless we need models that are strong enough to discriminate between hand types (i.e., left vs. right hands and the camera wearer's own hands vs. their social partner's).

Convolutional Neural Networks (CNNs), discussed in Section 2.2, offer very good performance for classification tasks [70]. For object detection, the now-standard approach is to divide an image into candidate windows, rescale each window to a fixed size, fine-tune a CNN for window classification [47, 124], and then perform non-maximum suppression to combine the output of the region-level classifier into object detection results. Of course, the space of possible proposal windows is enormous, so it is important to propose regions that capture as many objects as possible in the fewest number of proposals.

In the context of detecting hands in egocentric views, there are strong spatial

biases to hand location and size [11, 76], because of the way people coordinate head and hand movements. For example, people are likely to center their active hand in or near their visual field as they perform a task. We thus propose a simple approach to candidate window sampling that combines spatial biases and appearance models.

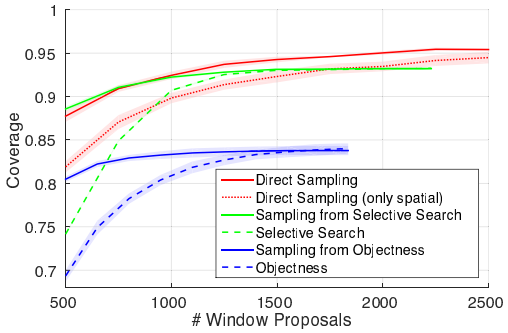
Generating Proposals Efficiently

Our primary motivation is to model the probability that an object O appears in a region R of image I. More concretely, we wish to estimate

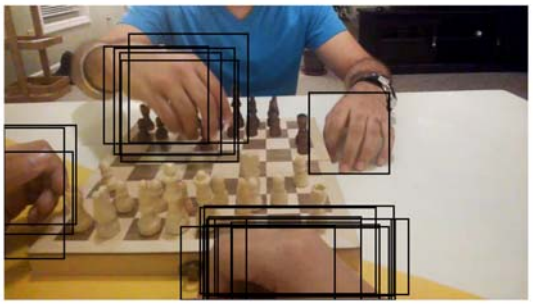
$$P(O|R, I) \propto P(I|R, O)P(R|O)P(O),$$

where $P(O)$ is the prior object occurrence probability, $P(R|O)$ is the prior distribution over the size, shape, and position of regions containing O, and $P(I|R, O)$ is an appearance model evaluated at R for O. Given a parameterization that allows for easy sampling, high quality regions can then be drawn from this distribution directly.

Here we assume regions are rectangular, so they are parameterized by an image coordinate and width and height. For each of the four types of hands, we can then estimate $P(O)$ directly from the training data. We fit $P(R|O)$ as a four-dimensional Gaussian kernel density estimator [58] (KDE) again using the ground truth annotations from the training set. For the appearance model $P(I|R, O)$ we define a simple model that estimates the probability that the central pixel of R is skin, based on the non-parametric skin color model described in the previous section. While simple, this model lets us sample very efficiently, by drawing a hand type O, and then sampling a bounding box from the KDE of $P(R|O)$, with the kernel weights adjusted by



(a) Coverage vs. Number of Proposals



(b) Example frame with top 20 IoU boxes

Figure 4.5: **(a)** Hand coverage versus number of proposals per frame, for various proposal methods. The mean and standard deviation (shaded) across five trials are shown. **(b)** An example frame with the twenty highest quality proposal boxes shown. Notice how hands are well covered by our method.

$P(I|R, O)$.

To evaluate this candidate generation technique, we measured its *coverage* — the percentage of ground truth objects that have a high enough overlap (intersection over union) with a proposed window to be counted as positive during detection. This is an important measure because it is an upper-bound on recall. Figure 4.5a shows coverage as a function of the number of proposed windows per frame for our method and two other popular window proposal methods: selective search [129] (which is the basis of the popular R-CNN detector [47]) and objectness [2]. The baselines were run using those authors' code, with parameters tuned for best results on our data (for selective search, we used the "fast" settings given by the authors but with k set to 50; for objectness, we used the standard hyper-parameters provided by the authors and retrained the object-specific weights on our dataset). As shown in the figure, our direct sampling technique (red solid line) significantly outperforms either baseline (dashed green and blue lines) at the same number of candidates per frame. Surprisingly, even our direct sampling without the appearance model (red dotted line)

performed significantly better than objectness and about the same as selective search.

To further investigate the strength of the spatial consistencies of egocentric interaction, we also subsampled the baseline proposals biased by our learned model $P(O|R, I)$. For both baselines, incorporating our learned distribution improved results significantly (solid blue and green lines), to the extent that biased sampling from selective search performs as well as our direct sampling for lower numbers of proposals. However, our full technique offers a dramatic speedup, producing 1500 windows per frame in just 0.078 seconds versus 4.38 and 7.22 seconds for selective search and objectness. All coverage experiments were performed on a machine with a 2.50GHz Intel Xeon processor.

Window Classification using CNNs

Given our accurate, efficient window proposal technique, we can now use a standard CNN classification framework to classify each proposal (after resizing to the fixed-sized input of the CNN). We used CaffeNet from the Caffe software package [62] which is a slightly modified form of AlexNet [70]. We also experimented with other network designs such as GoogLeNet [124], but found that when combined with our window proposal method, detection results were practically identical.

We found that certain adjustments to the default Caffe training procedure were important both to convergence and the performance of our networks. Only 3% of our proposed windows are positive so to avoid converging to the trivial majority classifier, we construct each training batch to contain an equal number of samples from each class. We also note that the typical data augmentation routine of mirroring examples is not used as it leads to classifiers that confuse left and right hands.

The full detection pipeline consists of generating spatially sampled window proposals, classifying the window crops with the fine-tuned CNN, and performing per-class non-maximum suppression for each test frame. Each of these components has a number of free parameters that must be learned. For our window proposal method, we estimate the spatial and appearance distributions from ground truth annotations in the training set and sample 2,500 windows per frame to provide a high coverage. The CNN weights are initialized from CaffeNet excluding the final fully-connected layer which is set by sampling weights independently from a zero-mean Gaussian. We then fine-tune the network using stochastic gradient descent with a learning rate of 0.001 and momentum of 0.999. The network was trained until the validation set error converged. The overlap thresholds for non-max suppression were optimized for each class based on average precision on the validation set. To keep our technique as general as possible, we do not take advantage of the constraint that each hand type should appear at most once in a given frame, although this is an interesting direction for future work.

4.3.2 PERFORMANCE IN NATURAL PAIRED INTERACTIONS

We collected and annotated a large dataset of paired interaction between two participants each wearing an egocentric camera. The dataset consists of four actors performing four tabletop activities at three different locations. In total, the dataset contains over 1.2 hours of video with over 15,000 pixel-wise hand annotations. We randomly split the videos into train, validation, and test sets ensuring as even a distribution of actor, activity, and location as possible. We refer to this division as the "main split."

We evaluate the effectiveness of our detection pipeline in two contexts: detecting hands of any type, and then detecting hands of specific types ("own left," "own right," etc.). In both cases, we use the PASCAL VOC criteria for scoring detections (that the intersection over union between the ground truth bounding box and detected bounding box is greater than 0.5). Figure 4.7 shows precision-recall curves for both tasks, applied to the "main split." For the general hand detection task (top-left), we obtain an average precision (AP) of 0.807 using our candidate window sampling approach, which is significantly higher than the 0.763 for selective search and 0.568 for objectness. The bottom-left pane of Figure 4.7 shows Precision-Recall curves for distinguishing between the four hand types.

There is a curious asymmetry in our hand type detections, with our approach achieving significantly better results for the social partner's hands versus the camera owner's. Figure 4.6 gives insight on why this may be, presenting detection results from randomly-chosen frames of the test set. Hands of the camera wearer tend to have many more duplicate detections on subparts of the hands (e.g. in row 2, column 2 of the figure). We attribute this tendency to how frequently "own" hands are truncated by the frame boundaries and thus appear as single or only a few fingers in the dataset. Including these partial detections alongside fully visible hands during training encourages the network to model both appearances to minimize error. While this does result in a loss of precision, the system gains the ability to robustly detect hands that are occluded or only partially in the frame (e.g. row 3, column 3) which is often the case for egocentric video.

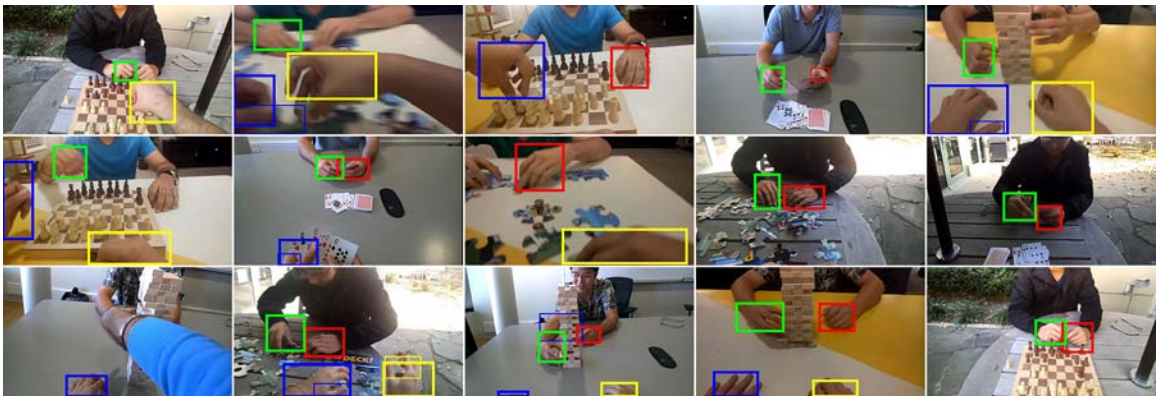


Figure 4.6: Randomly-chosen frames with hand detection results, for **own left**, **own right**, **other left**, and **other right** hands, at a detection threshold where recall was 0.7. Thick and thin rectangles denote true and false positives, respectively.

Error Analysis

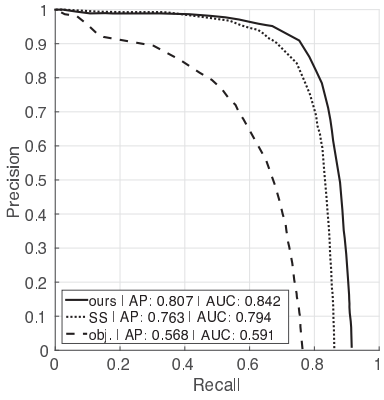
A related question is whether the errors are primarily caused by failure to detect hands of different types or confusion between hand types once a hand is detected. An analysis of the per-window classifications showed that only 2% of hand windows are mislabelled as other hands. Similarly for detection, 99% of undetected hands at a recall of 70% are due to confusion with the background class. Generally, our predictions tend to be nearly uniform for windows with ambiguous hand types, which are then removed by reasonable decision thresholds and non-max suppression. The qualitative results in Figure 4.6 also suggest that there is little confusion between different hand types.

Generalizing Across Actors, Activities, and Locations

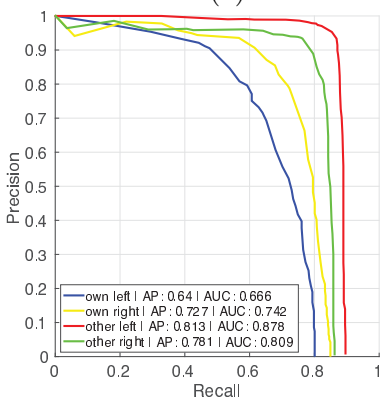
We next tested how well our classifiers generalize across different activities, different people, and different locations. To do this, we generated three different types of partitionings of our dataset across each dimension, where each split leaves out all videos containing a specific (first-person) actor, activity, or location during training,

and tests only on the held-out videos. We also split on actor pairs and activities jointly, creating 18 divisions (as not all pairs did all activities). This stricter task requires the method to detect hands of people it has never seen doing activities it has never seen.

Table 4.7c summarizes our results, again in terms of average precision (AP), with averages across splits weighted by the number of hand instances. The table shows that the detector generalizes robustly across actors, with APs in a tight range from 0.790 to 0.826 no matter which actor was held out. This suggests that our classifier may have learned general characteristics of human hands instead of specific properties of our particular participants, although our sample size of four people is small and includes limited diversity (representing three different ethnicities but all were male). For locations, the courtyard and office environments were robust, but AP dropped to 0.648 when testing on the home data. A possible explanation is that the viewpoint of participants in this location is significantly different, because they were seated on the floor around a low table instead of sitting in chairs. For activities, three of the four (cards, puzzle, and chess) show about the same precision when held out, but Jenga had significantly lower AP (0.665). The Jenga videos contain frequent partial occlusions, and the tower itself is prone to be mistaken for hands that it occludes (e.g. row 3, column 3 of Figure 4.6). Finally, splitting across actor pairs and activities results in a sharper decrease in AP, although they are still quite reasonable given the much smaller (about 6x) training sets caused by this strict partitioning of the data.



(a)



(b)

	All hands	Own hands		Other hands	
		Left	Right	Left	Right
Main split	0.807	0.640	0.727	0.813	0.781
All activities but:					
cards	0.768	0.606	0.776	0.708	0.732
chess	0.851	0.712	0.788	0.821	0.808
Jenga	0.665	0.644	0.693	0.583	0.502
puzzle	0.803	0.747	0.813	0.675	0.681
weighted average	0.772	0.675	0.768	0.699	0.686
All actors but:					
B	0.799	0.669	0.773	0.779	0.796
H	0.816	0.718	0.772	0.756	0.740
S	0.790	0.709	0.798	0.799	0.696
T	0.826	0.689	0.783	0.770	0.789
weighted average	0.807	0.700	0.782	0.776	0.756
All locations but:					
courtyard	0.790	0.702	0.785	0.755	0.755
office	0.772	0.659	0.757	0.794	0.687
home	0.648	0.558	0.703	0.538	0.591
weighted average	0.737	0.639	0.748	0.698	0.678
Split across actor pairs and activities					
weighted average	0.627	0.492	0.598	0.513	0.542

(c)

Figure 4.7: **(a)** General hand detection results with other window-proposal methods as baselines. **(b)** Results for detecting four different hand types compared with Lee *et al.* [76]. **(c)** Hand detection accuracy when holding out individual activities, participants, and locations, in terms of average precision. For example, the training set for *all activities but cards* included all videos *not* containing card playing, while the test set consisted *only* of card playing videos.

4.3.3 HAND SEGMENTATION AND ACTIVITY RECOGNITION

While detecting hands may be sufficient for some applications, pixel-wise segmentation is often more useful, especially for applications related to hand pose recognition and in-hand object detection [87]. Once we have accurately localized hands using the above approach, segmentation is relatively straightforward, as we show in this section. We use our detector both to focus segmentation on local image regions, and to provide semantic labels for the segments. Additionally, the location and pose of hands also have a natural correlation with the activity they are performing. In this section we explore hand segmentation and the influence of hand location and pose on object-independent activity recognition.

Semantic Segmentation

Our goal in this section is to label each pixel as belonging either to the background or to a specific hand class. We assume most pixels inside a box produced by our detector correspond with a hand, albeit with a significant number of background pixels caused both by detector error and the general difference in shape between hands and rectangular bounding boxes. We can presume that within a bounding box there are two generative distributions from which each pixel is drawn, skin color and background. A similar model has been developed previously for segmentation as the well-known semi-supervised segmentation algorithm, GrabCut [109] which we adapt for use in our unsupervised context.

Given an approximate foreground mask, GrabCut improves the segmentation in an iterative manner similar to expectation maximization methods. Starting from the color distributions modeled from the initial foreground and background masks,

a binary label (either foreground or background) is estimated for each pixel. This is done by modeling the region as a grid graph Markov Random Field with factors that encourage neighboring pixels to share the same label. After inference on this graph is complete, the color models are recomputed using the new label assignment. This process alternates between fitting the color models and updating the labels until convergence or a fixed number of iterations.

In more detail, for each detected hand bounding box, we use the simple color skin model described in Section 4.3.1 to estimate an initial foreground mask. We use an aggressive threshold so that all pixels within the box are marked foreground except those having very low probability of being skin. Note that we avoid running GrabCut on the entire image because arms, faces, and other hands would confuse the background color model. Instead, we use a padded region around the bounding box, ensuring that only local background content is modeled. We take the union of the output masks for all detected boxes as the final segmentation.

Using the skin color model learned for the training set, we detected hands and produced segmentations for each frame in our test set. To put our results in context, we ran the publicly-available pixel-wise hand detector of Li et al. [83], which was designed for first-person data. Their model learns a pixel-wise skin classifier based on local color features for each individual frame of the training data. At test time, each input frame is compared to the training frames and the output of the learned classifiers for the nearest k training frames based on global features are averaged. The scores are thresholded and connected components are extracted in a post-processing step. We trained their technique with 900 randomly-sampled frames from our training set. As previously mentioned, that paper defines "hand" to include any skin regions

connected to a hand, including the entire arm if it is exposed. To enable a direct comparison to our more literal definition of hand detection, we took the intersection between its output and our padded bounding boxes (i.e. we compared only on regions where both methods could produce output).

Table 4.3 presents segmentation accuracy, in terms of pixel-wise intersection over union between the estimated segmentation mask and the ground truth annotations. Our technique achieves significantly better accuracy than the baseline of [83] (0.556 versus 0.478). A similar trend is present across the stricter actor pair and activity data splits. Figure 4.8 shows our segmentations on some randomly-sampled test frames. Examining the differences between our approach and the baseline lends some insight. Our GrabCut-based approach looks only at local image color distributions and leans heavily on the quality of our detections. The baseline method, however, learns classifiers that must perform well across an entire frame which is complicated by the close visual similarity between hands and other visible skin.

Our method has two main possible failure modes: failure to properly detect hand bounding boxes, and inaccuracy in distinguishing hand pixels from background within the boxes. To analyze the influence of each, we perform an ablation study based on the ground truth annotations. Applying our segmentation approach to the ground truth detection boxes instead of the output of the hand classifier, our results rose from 0.556 to 0.73. On the other hand, taking the output of our hand detector but using the ground truth segmentation masks (by taking the intersection with the detected boxes) achieved 0.76. Each of the studies improve over our fully automatic approach by roughly 30-35%, indicating that neither detection nor segmentation is individually to blame for the decrease in accuracy, and that there is room for future work to

improve upon both.

	Own hands		Other hands		
	Left	Right	Left	Right	Average
Main split					
Ours	0.515	0.579	0.560	0.569	0.556
Li et al. [83]	0.395	0.478	0.534	0.505	0.478
Split across actor pairs and activities					
Ours	0.357	0.477	0.367	0.398	0.400
Li et al.	0.243	0.420	0.361	0.387	0.353

Table 4.3: Hand segmentation results in intersection over union with annotations.

Figure 4.8: Segmentation results on random hand regions.

Hand-based Activity Recognition

We now investigate one particular application of hand detection and segmentation in first-person video: activity recognition. Interacting with different objects affords different types of hand grasps, the taxonomies of which have been thoroughly studied

[94]. Moreover, when multiple actors are interacting, it seems likely that the absolute and relative position of hands within in the field of view also reveals evidence about the activity that the actors are performing. An interesting question is whether activities can be detected based on hand pose information alone, without using any information about the appearance or identity of handled objects or the rest of the scene. Aside from academic interest, focusing on hands independently of scene could be valuable in recognition systems: while it may be impossible to model or anticipate every single handled object or visual environment, we have shown that it is very possible to accurately detect and segment hands. To what extent could hand pose alone solve activity recognition in first-person views?

To address this question, we fine-tuned another CNN to classify whole frames as one of our four different activities. To prevent the classifier from seeing any information other than hands, we used the ground truth segmentation to mask out all non-hand background. The network saw 900 frames per activity across 36 videos during training and 100 per activity across four videos for validation. The classifier achieved 66.4% per-frame classification accuracy, or roughly 2.7 times random chance, on our test dataset with non-hand regions blacked out. While these results are not perfect, they do confirm a strong connection between activities and hand location and pose.

To evaluate how well the technique would work in an automated system, we reran the above experiment using the output of our segmentation instead of the ground truth for the test set. The per-frame activity classification accuracy falls from 66.4% to 50.9%, but this is still roughly twice random chance.

This decline is caused by two types of errors, of course: incorrect information

about the spatial configuration of the hands due to imperfect detection, and incorrect hand pose information due to imperfect segmentation. We once again investigated the relative effect of these errors, similar to the ablation study as before, and found that replacing either detection or segmentation with ground truth increased the fully automatic performance by about nine percentage points. This suggests that capturing the spatial arrangement of hands and correctly predicting their pose are equally important to per-frame activity recognition using only hand information.

So far we have considered each frame independently, but of course much information about activity lies in the temporal dynamics of the hands over time. We tried a simple voting-based approach to incorporate some of this temporal structure: we classify each individual frame in the context of a fixed-size temporal window centered on the frame. Scores across the window are summed, and the frame is labeled as the highest scoring class. To again compare with the ground truth informed upper bound, we only consider labeled frames, so a window of k frames spans approximately k seconds.

Table 4.4 presents the results. Temporal information increases activity recognition accuracy significantly, with even a window of five frames improving results from 0.664 to 0.764 when using ground truth segmentations, and from 0.509 to 0.618 using the fully automatic system. Accuracy continues to improve with increasing window size, with 50 frames achieving 0.929 with the ground truth and 0.734 for the automatic segmentations. This improvement is likely due to two factors: certain hand poses may be more distinctive than others, and segmentation errors in any given frame can be disregarded as outliers.

We also show results averaged over stricter splits, such that any actor seen in

	Window size (k)				
	1	5	15	30	50
Main split					
Segmentation mask	0.509	0.618	0.680	0.724	0.734
Ground truth mask	0.664	0.764	0.851	0.900	0.929
Split across actor pairs (average)					
Segmentation mask	0.570	0.639	0.679	0.687	0.671
Ground truth mask	0.661	0.742	0.790	0.814	0.847

Table 4.4: Activity recognition accuracy from hand masks, using a temporal window of k frames. See text for details.

testing is not seen in training. This partitioning reduces the number of splits with enough test data to two, since not all pairs performed all activities. Though limited in scope, the results of this strict task are similar to the "main split."

Our results suggest that hand segmentation could deliver high activity recognition accuracy without the need to recognize objects or backgrounds; however, our experiments also show that automated approaches would benefit from increased segmentation accuracy.

4.4 CONCLUSION AND FUTURE WORK

In this chapter, we developed powerful hand detection methods which can accurately locate and disambiguate hands in egocentric videos, both in constrained and unconstrained settings. Our models make use of probabilistic graphical models as well as powerful Convolutional Neural Networks to provide high quality annotations to developmental and social psychologists studying how we as humans interact with the world and each other in paired interactions. We further showed that these detections

could also be used to yield state-of-the-art hand pose segmentation and we explored the potential of these segmentations by showing that activities can be successfully recognized in our first-person dataset based on the configuration and pose of hands alone. In future work, we would like to generalize our dataset to more complex social interactions and improve our models to directly produce semantic segmentations of hands rather then rely on a pipeline approach like we have presented here.

In this and the previous chapter, we have demonstrated the effectiveness of computer vision techniques for annotating both simple and complex structures in images and videos. While we have presented our results on specific tasks, the models we present can be generalized to other tasks. For instance, similar models could be leveraged to improve interaction detection for automated animal behavior studies [29] or to identify animals in trail cameras [141] to track biodiversity metrics for ecological studies.

The following two chapters address discovery tasks which seek to produce new information from visual data rather than to simply locate specific visual elements within images. In contrast to the annotation tasks presented thus far, manual solutions to this class of problem are either intractable at the necessary scale, require considerable time from human experts, or both. In these tasks, the amount of data is too large and the signals are often too weak to be found by human effort without years of study.

CHAPTER 5

AUTOMATICALLY EXTRACTING ARCHITECTURAL TRENDS

For many academic pursuits, the analysis of imagery is less concerned with locating specific image structures for which the characteristics and appearance are known in advance as in the previous chapters, but rather to *discover* visual elements which correlate to higher level semantic concepts. Examples of this sort of analysis can be readily found in the study of art or architecture, wherein a researcher might seek to identify the characteristic aesthetics of painters or architects that produce work at a certain time and within certain cultures. In this chapter, we develop a general method of identifying image substructures that correlate with image-level labels. We ground our data-mining approach and its motivations within the context of the highly visual subject of architectural history, identifying architectural elements that typify different time periods of construction.

5.1 DISCOVERING ARCHITECTURAL TRENDS

As a matter of course, architectural styles change over time, reflecting the evolving artistic design, social and cultural attitudes, and technological and socioeconomic conditions of the peoples that built them. Studying features of buildings gives a window into the past, letting us observe properties of style and design at the time

they were built. To study such phenomena manually would require tremendous efforts in data collection, annotation, and analysis. In this chapter, we present a fully automatic method that harnesses tens of thousands of images to discover architectural elements that are temporally distinctive [77], i.e. they are indicative of certain periods of construction. We also track the evolution of these elements over time to identify how style has changed over the last two centuries. We use the city of Paris as a proof-of-concept and apply our method to produce novel observations.

While there is existing work in computer vision that has considered architectural applications such as classifying between different architectural styles or parsing building facades into predefined components [14, 116, 117, 119, 135, 139], these methods are constrained to small datasets and do not attempt to identify how architecture changes over time. Perhaps most related to our work in both method and scale is work in mid-level visual mining that tries to find discriminative image patches. Doersch *et al.* [31, 32] discover patches that discriminate between different cities using similar data as used here. Like us, Lee *et al.* [80] considers the temporal domain, finding style-independent classifiers of style-discriminative elements present throughout multiple time periods (e.g. automotive headlights, which have been on cars for fifty years but whose style has changed dramatically over time). Our work also finds elements with similar semantics through time, but we handle the additional challenge that elements in architecture are much more dynamic, with certain elements such as window shutters rising to prominence for decades only to fall out of favor later.

Other recent work has used Google Street View, but for other applications than ours. Arietta *et al.* [7] use regressors based on mid-level patches to predict geospatially distributed statistics such as crime rate and wealth. Ordonez and Berg [98]

predict attributes of neighborhood safety, uniqueness, and wealth. Zhou *et al.* [146] demonstrate that the frequency with which Street View images contain certain attributes such as green space, tall buildings, water, and social activities can be used to identify a particular city.

5.2 GENERATING AN ARCHITECTURAL DATASET

Rather than manually photographing buildings and researching their construction dates, we employ a noisy but automated process combining images from Google Street View (GSV) and a cadastre map (a survey of real estate boundaries) labeled with coarse construction period. We focus here on Paris as it is one of the world's best-known cities and because data on building construction dates is available from the Paris Urban Planning Agency (Atelier Parisien d'Urbanisme) [6].

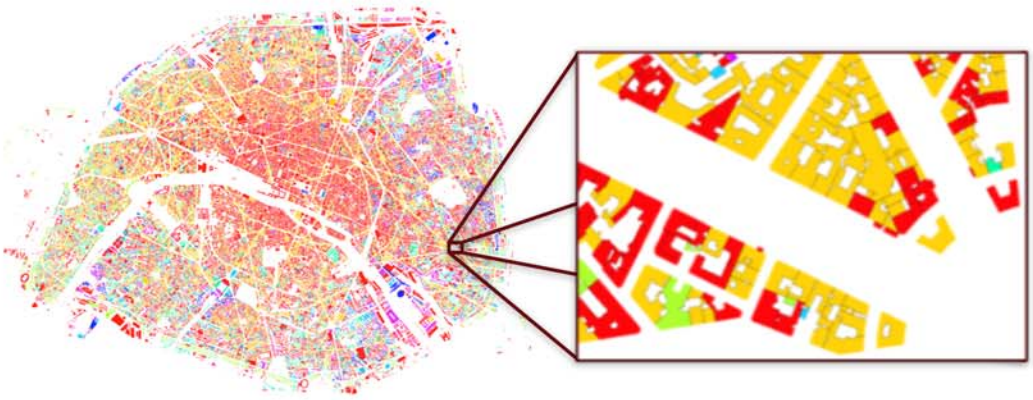


Figure 5.1: The cadastral map of Paris provided by the Paris Urban Planning Agency (Atelier Parisien d'Urbanisme) [6] provides fine-grained 2D building geometry and construction period annotations. The periods are color coded as pre-1800, 1801-1850, 1851-1914, 1915-1939, 1940-1967, 1968-1975, 1976-1981, 1982-1989, 1990-1999, and post-2000.

Our dataset generation method relies on three key data sources: fine-grained building geometry, construction period annotations, and a large collection of photos

taken at known positions. We use a digital cadastre of Paris to retrieve detailed building geometry and construction dates. The cadastre (shown in Figure 5.1) was provided by the Paris Urban Planning Agency (Atelier Parisien d'Urbanisme) [6] and includes over 120,000 buildings. Almost all of the buildings have a label indicating their coarse construction period, either pre-1800, 1801-1850, 1851-1914, 1915-1939, 1940-1967, 1968-1975, 1976-1981, 1982-1989, 1990-1999, or post-2000. To provide geotagged image data, we collected every current GSV image and associated location metadata taken within the Paris city limits, yielding about 145,000 panoramas. Each image is captured using arrays of 9 to 15 cameras on Google's custom Street View vehicles [4].

Our dataset generation process is outlined in Figure 5.2. To connect Street View images with specific building information, we need to align the images with the cadastre map and identify which building facades have been imaged. Each GSV image has a GPS coordinate such that placement in the cadastre map is a simple task, but we must still decide which buildings an image has captured and how to crop the panoramas to extract individual facades. To do this, for each panorama we look up the Street View vehicle's heading from the metadata and cast rays in 160° cones from each side of the vehicle. The rays are cast at 1° intervals and are 30 meters long, which is sufficient to reach the buildings on even the larger Parisian thoroughfares (see Figure 5.2a). We compute the first facade encountered by each ray, and select the pair of rays from each facade with the greatest angular difference (see Figure 5.2b). We then crop and warp the panoramas to produce multiple facade images per Street View panorama. After discarding facade images that are overly-skewed or very narrow images, our method results in 70,000 nearly-planar facade images. We sample

20,000 facades for analysis, evenly distributed among the construction periods.

(a) (b) (c)

Figure 5.2: Overview of data generation from Google Street View (GSV) images and cadastre maps. We first **(a)** cast rays to the sides of each GSV capture location at 1° intervals, then **(b)** compute intersections with facades and select the widest view, and **(c)** project onto the panoramas and crop and warp the facade images according to GSV metadata.

5.3 MINING TEMPORALLY DISTINCTIVE ELEMENTS

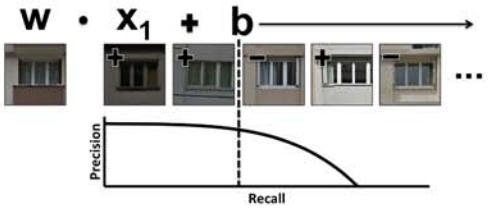
We start with the natural intuition that a specific building element is discriminative of a time period if it occurs in that time period far more often than in others. Following this idea, we develop a mining approach that approximates the occurrence frequency of a large randomly selected set of candidate patches. Figure 5.3 shows an overview of this process.

We begin with a mining approach similar to that of [32], generating a large set of candidate visual elements by sampling 25 patches at different resolutions from a held out set of images selected uniformly across periods. We represent each patch in Whitened HOG (WHO) space [44, 52], a feature sensitive to image gradient orientations. For each of these patches, we build a set of initial "detections" by finding the closest match in each of the remaining images. This results in a set of nearly 1 billion associations between patches which we use to estimate period-wise occurrence frequencies.

For each patch C_i, we label the closest 200 patch matches as positive if they come

(a) Sample candidate patches from a held out image set at multiple scales.

(b) Find nearest neighbor for each candidate in each other image across multiple scales.



(c) Take each candidate representation as the weights of a linear classifier, treating patches from the same period as positives.

(d) Rank the candidates by the quality of the classifier produced in step (c) and remove near duplicate candidates.

Figure 5.3: Overview of the period-specific element mining process.

from the same period as C_i and negative otherwise. Treating C_i's vectorized WHO representation $\hat{C}_i$ as the weights of a simple linear classifier $f_i(x) = \text{sign}(\hat{C}_i \cdot \hat{x} + b_i)$, we can quantify to what extent being visually similar to patch C_i predicts construction period. This formulation can be interpreted as a classifier trained with the square loss with the candidate C_i as a positive example and a large number of negative data points [8, 44, 52]. We apply $f_i(x)$ to each of the patch matches and vary b_i to produce a precision-recall curve. We rank the candidates by area under this curve (AUC).

At this point in the procedure, we have many candidates that are indicative of their periods; however, the candidates suffer from a lack of diversity, with many of the highest ranked elements being redundant. This is a natural effect of the initial patch sampling capturing different instances of the same frequently occurring and highly discriminative facade elements within a period. To improve the representativeness of

Figure 5.4: The top nine most discriminative patches from each period are shown above. Each 3x3 grid provides a good representation of the style of architecture for that period.

our final candidates, we prune this set by greedily removing weaker candidates that have many overlapping detections with stronger candidates.

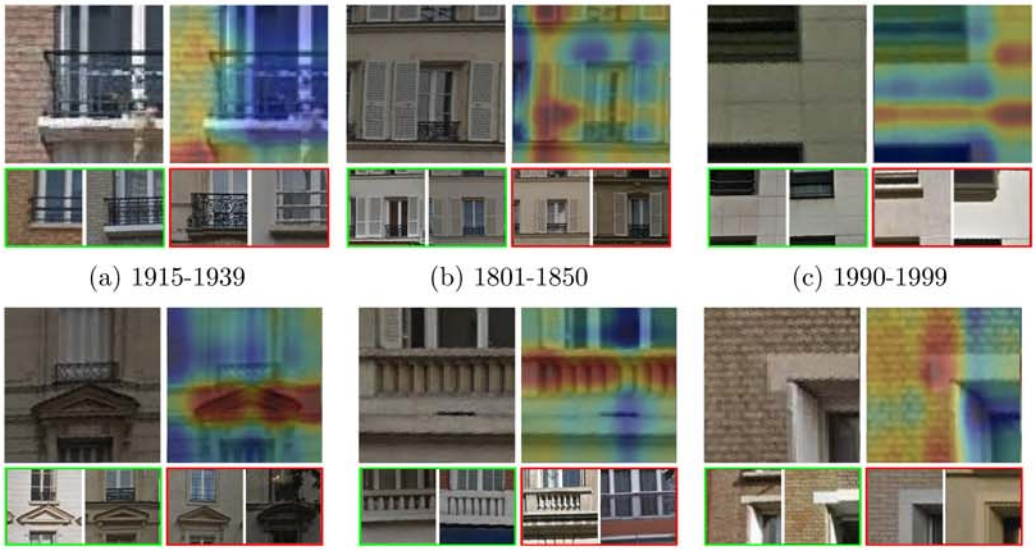
This procedure finds visual elements whose appearance correlates with building age, and ignores other common elements found throughout the city (like pavement, signs, bus stops, etc.) due to their very poor ranking; Figure 5.4 shows top patches for each period. Our larger goal is to find patches that are relevant and useful to studies of architecture. This is difficult to quantify, so we showed our discovered patches to an expert on Parisian architecture and asked for feedback [128]. They informed us that many of the patches did capture key elements known to be prevalent in their respective periods.

Fine-Grained Analysis

Given the facade level patches, we also took a finer-grained perspective, looking for the most discriminative substructures within each patch. For this analysis, we repeat our classification and ranking procedure while masking out regions of the candidate patch. By observing relative changes in the AUC, we can note which spatial cells are most discriminative, which we visualize as heatmaps in Figure 5.5. The 1915-1939 period is characterized by raw brick facades, highlighted in Figure 5.5a. Figures 5.5b and 5.5c suggest details not identified by our expert. In Figure 5.5b, the spacing between adjacent window shutters appears to be influential. In Figure 5.5c the additional horizontal line is missing in many similar pre-1990 facades. The cap in Figure 5.5d is highlighted as well. Interestingly the highlight extends off the right-hand side indicating that the continued horizontal may also be important. The railing in Figure 5.5e sets itself apart from other similar elements by the plainness of its columns as compared to close negative patches. Figure 5.5f is unique among the examples because the map highlights an area because of what is *not* present: in the close negative examples, the white trim extends down the side of the window.

Facade-Level Analysis

Another way to evaluate the usefulness of our discovered patches it to use them to evaluate the 'periodness' of whole facades. For each facade, we found the top 100 detected patches. We sum the AUC of the detected patches for each period in a facade to produce an unnormalized distribution over how well each period's patches fit the given facade. In Figure 5.6 we show the highest likelihood facade for each period. Each image is accompanied by an over-painting of patch detections with colors cor-



(a) 1915-1939 (b) 1801-1850 (c) 1990-1999

(d) 1851-1914 (e) 1915-1939 (f) 1915-1939

Figure 5.5: Sample discriminative elements. Each figure shows a patch (top-left), a fine-grained importance map (top-right), close examples from the same (bottom-left) and other (bottom-right) time periods. Best viewed in color.

responding to source period and a "reconstruction" of the image made by averaging these detections. For instance, notice how the 1851-1914 facade demonstrates the similarity in the periods spanning pre-1800 to 1914; its overpainting has colors corresponding to patch detections from pre-1800 (red) and 1801-1850 (orange) in addition to its own (yellow). The figure presents a sense of the progression in style and types of buildings constructed in Paris over the last two centuries, as modern materials gradually overtake old. The confusion about later periods is again seen here, with later periods exhibiting higher degrees of confusion (indicated by more mixture of color).

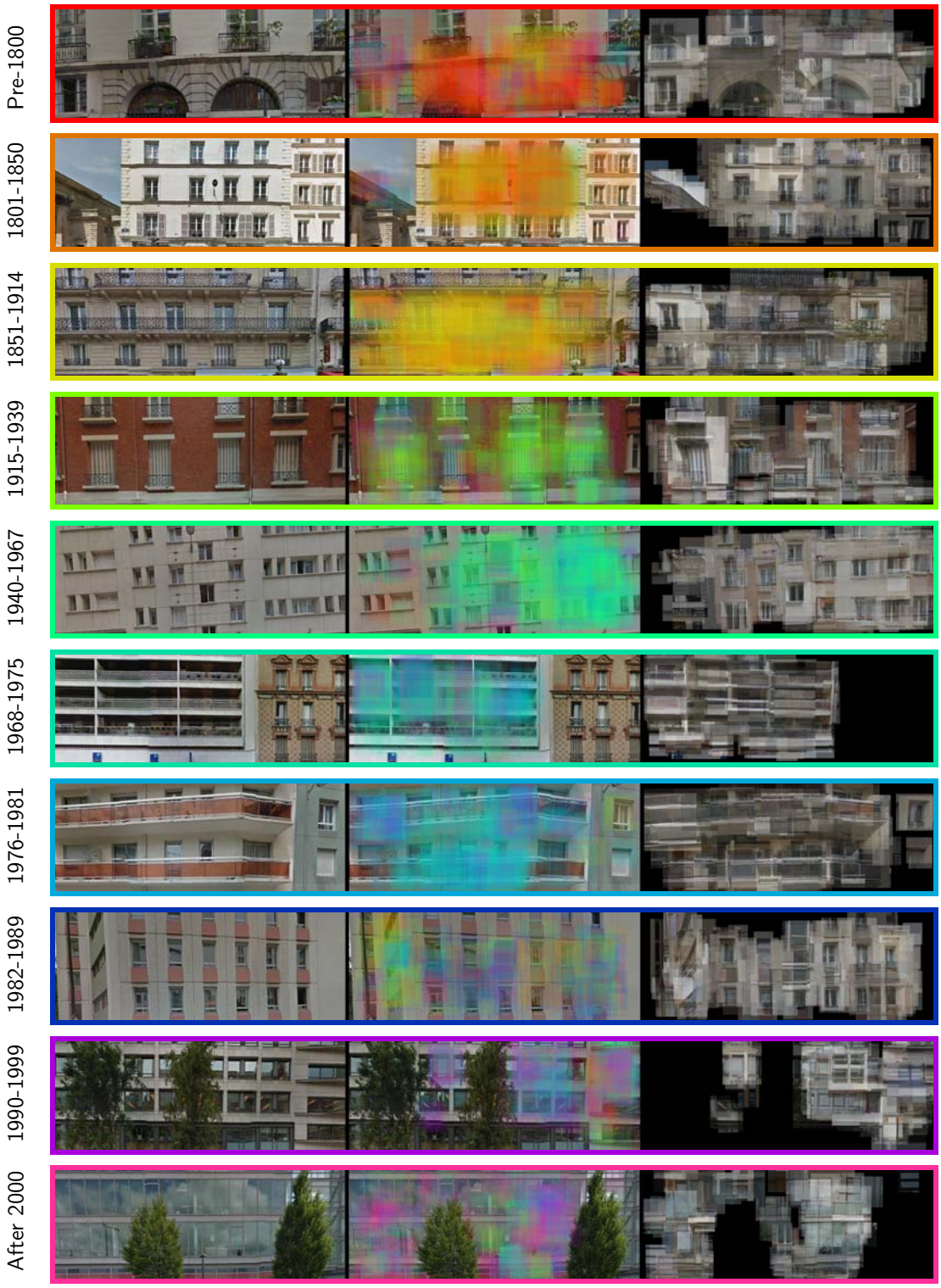


Figure 5.6: In each row we show the original facade (left), the original overpainted with the periods of the top 100 patches (middle), and a facade reconstruction where the period patches are replaced by their average images (right).

5.4 LINKING ELEMENTS THROUGH TIME

Functionally-identical elements of buildings can change substantially over time; for example, the styles of windows, doors, balconies, etc. vary dramatically across different architectural periods. We automatically identify these evolutions by looking for "chains" of elements that are discriminative to their particular time period, but are still coarsely similar in appearance to elements in sequential periods. We cannot fix the length of the chain or the beginning or ending periods in advance, as elements may appear or disappear over time. This problem is reminiscent of multiple-target tracking [13], in which detections of an object from sequential frames of video are stitched together to form trajectories, except that we are "tracking" patches over sets of images from different time periods.

Given a set of candidates C, we define a directed acyclic graph $G = \{\mathcal{V}, \mathcal{E}\}$ such that $\mathcal{V} = \{s, t\} \cup \mathcal{C}$ where s and t are special source and sink nodes. The graph forms a trellis, such that each patch in any given time period has an outgoing edge to every patch in the next period, while the source and sink connect to all nodes of the graph. Figure 5.7 presents a sample graph with four periods and three patches per period. Intuitively, the inter-period connections provide possible evolutions of corresponding elements. The source and sink nodes are added to determine the start and end of a chain, with weights such that if many matches for a patch are from the future, it is likely to be a starting point; otherwise, it tends to be an ending point.

For the edge weights, we need a measure of similarity that will connect patches likely to correspond to the same functional elements (e.g. windows, balconies, etc). We could use distance in WHO space; however, misalignments in the initial patches

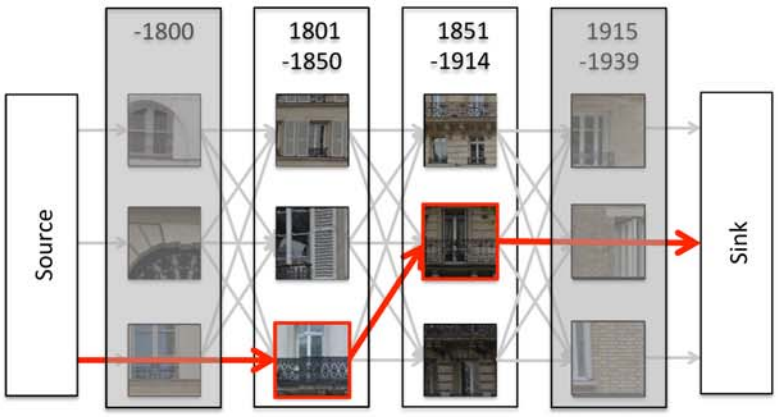


Figure 5.7: Elements in adjacent periods are fully connected with weights depending on their co-occurrence, while the source and sink connect to every node with weights that penalize the number of skipped periods. Here, the shortest path (in red) skips pre-1800 and 1915-1939 because they lack the long balconies of the other periods. For clarity, this visualization shows only four periods.

could lead to large differences in the WHO features of two nearly identical elements. Based on this observation we use relationships between images and candidate patches to estimate the spatial consistency of overlapping detections as a measure of similarity. Specifically we use the trimmed-mean deviation between co-occurring detections weighted by how fully the detections overlap.

The source and sink nodes are attached to every other node with weights dependent on each candidate's closest detections. In particular, the weight from source to a node C_i is defined as $n_< * \beta * f_<$ where $f_<$ is the fraction of the top 200 detections for candidate C_i that are from previous periods and $n_<$ is the number of previous periods. The weights to the sink are defined similarly, considering periods following that of C_i. Generally these sink and source weights will be low when a patch lacks many detections in earlier or later periods respectively. We set β empirically as a typical edge cost in a high quality chain, so that these weights balance the total cost of continuing the chain, amortized by how likely the chain should continue based on

Figure 5.8: Sample chains of architectural elements across time periods, showing how our technique discovers functionally-similar elements whose appearance has evolved over time.

the frequency analysis.

To generate chains, we greedily find the shortest path from source to sink, remove it, and repeat. Figure 5.8 shows sample chains of varying length and differing elements. Figures 5.8a and 5.8g show increasingly ornate window dressings starting from very plain structures before 1800, to multiple decorative structures in the 1851–1914 period. Figure 5.8c shows the long window balconies of the 1850s to 1940s, while Figure 5.8d shows an evolution of short balconies. Many similar chains are produced as there is a great deal of variety in balcony shapes over time. Some chains show consistent directions of change, for instance Figure 5.8e demonstrates the increasing depth of windows. The last chain in Figure 5.8f highlights railings for large buildings after 1940, with the railings transitioning to glass in 1982–1989 and into metal in the 1990s.

5.5 CONCLUSION AND FUTURE WORK

In this chapter, we presented simple but effective methods to automatically discover and track visually important architectural elements using an automatically annotated collection of thousands of street-level images of Paris. The images are mapped to buildings in a fine-grain urban planning model that annotates each with a rough construction date. Using these combined data sources, we mine for period specific stylistic elements, analyze facade-level architectural influences, and find evolutions of elements across times. Moreover, our methodology provides a general framework for identifying substructures in images that correlate with image-level labels and is a step towards developing automatic techniques to mine large-scale image collections to discover meaningful visual patterns.

In the next chapter, we will show how characteristics of whole images (including their substructures) can be used to predict high level semantic concepts by implicitly discovering features and objects in images that are predictive of the higher level concepts. Unlike in this task, in the next chapter the primary goal is not to identify the elements themselves, but rather to use them to provide an automated way of predicting other attributes of the image content.

CHAPTER 6

PREDICTING DEMOGRAPHIC AND GEOGRAPHIC ATTRIBUTES FROM IMAGES

To a human observer, details and objects in an image can easily point to higher level concepts about the place the image was taken. These concepts can range from simple features like mountains or cliffs indicating the rough elevation of a place, to subtler clues like bars on windows suggesting how much crime to expect in a region. Many of these high-level visually inferable attributes are of interest to scientists studying geography, ecology, and demography. In automatically identifying these high-level attributes, a system must discover related visual elements it can use to make high quality predictions. For modern pixel-to-prediction models like deep networks, this process of discovering salient objects, patterns, and image statistics is an implicit part of training to perform well at the task. In this chapter, we use powerful deep models to predict a wide range of over a dozen geographic and demographic properties from the content of single images alone.

6.1 PREDICTING ATTRIBUTES DIRECTLY FROM CONSUMER PHOTOGRAPHY

Demography is the study of statistics concerning births, deaths, immigration, and other socioeconomic factors that measure the changing structure of human populations. Many subfields exist within demography which study social and economic effects through the lens of population dynamics. Geographers study the Earth and how people interact with it. In this chapter, we develop automated methods of estimating coarse demographic and geographic properties from single photos. Our technique can be viewed as a indirect observational methodology which could provide demographers with automated tools to track shifting properties without the need of formal survey. We develop methods to automatically construct large labeled datasets to train and evaluate modern machine learning approaches to predict these attributes. We demonstrate the capabilities of our approach on a wide range of demographic and geographic properties [79].

Work in scene classification has considered demographic categories, like urban versus rural [137]. Similarly, recent work by Zhou *et al.* [146] learns a suite of hand-selected scene classifiers as composites of many categories from the SUN attribute dataset [100]. The attributes capture different facets of a city including architecture, greenery, and transportation. They apply these classifiers on a dense corpus of social geo-tagged images to analyze the role of different attributes in city recognition and similarity tasks. These approaches and others like them use hand-selected categories and carefully-labeled training data, whereas we take a data-driven approach, learning over a dozen attribute classifiers using social images annotated with noisy training

labels. Newsam *et al.* [81, 138] try to reconstruct maps of land use type and "scenicness" by pooling visual features from images taken in a particular location. While similar to our approach, our scope is broader: we test over a dozen attributes at a worldwide scale, whereas they study one type of attribute for part of the United Kingdom.

We also note that geographic and demographic properties tend to be spatially distributed across the globe, such that these attributes are informative of image location. Other recent work has studied localization of individual images, typically using geo-tagged photos from photo-sharing sites like Flickr as (noisy) reference images [**?**, 51, 54, 84, 85, 102, 104, 120, 145]. Among those most related to ours, Hays and Efros [54] use their geo-locations to infer population and elevation by looking up the estimated geo-tag on a geographic information system GIS map. We also estimate population and elevation (in addition to many other attributes), but by classifying attributes directly instead of geo-locating and then looking up the corresponding attribute values.

6.2 AUTOMATING ATTRIBUTE-ANNOTATED DATASET CREATION

We assembled a large collection of about 40 million geo-tagged images from the photo-sharing website Flickr. From this set, we filtered out photos having imprecise geo-tags (those less accurate than about a city block). Unfortunately, a large fraction of these images come from a relatively small number of places due to biases inherent in consumer photography. If we simply use the whole collection (or sample uniformly from it), we risk producing classifiers that memorize the appearance of a few key landmarks

without abstracting general visual properties of places that exhibit various attributes.

We thus attempt to bias the sampling as if we were drawing uniformly at random over the surface of the globe, instead of sampling directly from the geo-spatial distribution of Flickr photos. To do this, we discretized the world into $0.01^\circ \times 0.01^\circ$ latitude-longitude bins (roughly 1 km $\times$ 1 km at the middle latitudes). We randomly sample photos one-by-one, but ignore samples from bins from which we already have 100 photos. To prevent individual highly-active users from introducing bias, we avoid sampling more than five photos from any single user. Finally, we partition the data into training and testing sets; to help prevent (nearly) identical photos from leaking across the partitions, we divide on a per-user basis (so that all photos from a single photographer are placed in one set or the other).

We collected public gridded GIS data for 15 attributes including geographic features (e.g. elevation, elevation gradient and land-use) and demographic features (e.g. population density, wealth, ethnic composition). The data came from a variety of public sources including NASA and USGS, and the granularity of the gridded data ranged from about 30 arc-seconds to up to about 15 arc-minutes (see Table 6.1a for details). We used global data when available, although some of the attributes were available only for the U.S. We avoided time-varying attributes (e.g. temporal climatic attributes like daily temperature or rainfall); these attributes could be useful if accurate timestamps are known but we do not assume that here. Many of these attributes are correlated to some degree, as visualized in Figure 6.1b, and thus we can expect some structure in the recognition accuracies across different attributes.

We automatically produce labeled datasets for each attribute, by simply examining each photo's geo-tag, looking up the value of the attribute at that location in the

gridded GIS map, and then assigning that label to the photo. Of course, this process is noisy: many geo-tags on Flickr are incorrect [53], and our worldwide attribute maps are coarse enough (about 1–10km square) that attribute values may vary dramatically even within a single bin. Attribute values are most informative if they are relatively extreme — i.e. quite high or quite low. Thus we consider a restricted classification problem in which the goal is to label an image as having a high or low value for an attribute (e.g. high population or low population). We label images as high or low by thresholding at the highest and lowest quartile (25% and 75%) of the worldwide value of each attribute.

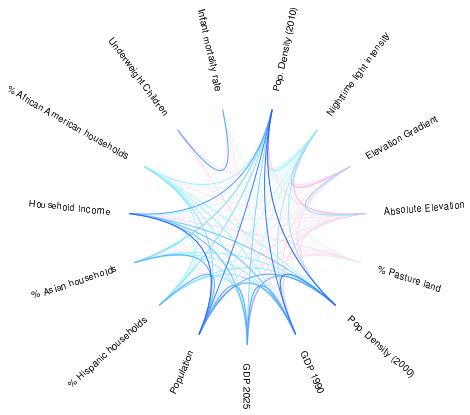
We partitioned the data into training, validation, and test sets (about 60%, 10%, 30% respectively) on a per-user basis to prevent leakage between the sets. We also ensured the class distribution in the test sets remained equal such that the random baselines are 50% for ease of interpretation.

6.3 ATTRIBUTE PREDICTION AS A CLASSIFICATION TASK

Inspired by the impressive results of convolutional deep learning on a variety of recognition problems (e.g. [70, 97, 126, 127] among others), we apply them to our problem of attribute recognition. We start from the neural network architecture proposed by Krizhevsky *et al.* [70], with five convolutional layers (C1 to C5) followed by three fully connected layers (FC6 to FC8), and the same mechanisms for contrast normalization and max pooling. In total, this model has about 60 million parameters. Our training dataset is not sufficiently large to train such a large number of parameters from scratch, so we follow Oquab *et al.* [97] and others and initialize from a model pretrained on ImageNet. We modify the final fully connected layer (FC8) for each

Attribute	Source	Year(s)	Grid size	Description
% African American Households	[114]	2000	30 arcsec	Percentage of households identifying as African American.
% Asian Households	[114]	2000	30 arcsec	Percentage of households identifying as Asian.
% Hispanic Households	[114]	2000	30 arcsec	Percentage of households identifying as Hispanic.
% Pasture	[105]	2000	0.5 arcmin	Proportion of land areas used as pasture land.
% Underweight Children	[23]	1990-2002	2.5 arcmin	Estimates of the percentage of underweight children.
Elevation	[130]	1996	30 arcsec	Elevation according to USGS's global digital elevation model.
Elevation Gradient	[130]	1996	30 arcsec	Elevation gradient according to USGS's global digital elevation model.
GDP 1990	[140]	1990	15 arcmin	GDP in millions of USD.
Infant Mortality Rate	[23]	2000	2.5 arcmin	Estimates of infant mortality rates for the year 2000.
Nighttime Light Intensity	[95]	2009	30 arcsec	Composites of nighttime lights as seen from space for calendar year.
Population Density (2000)	[24]	2000	2.5 arcmin	Population densities adjusted to match UN totals, persons per sq. km.
Population Density (2010)	[24]	2010	2.5 arcmin	Population densities (projected based on 2000 data), persons per sq. km.
Predicted GDP in 2025	[140]	2025	15 arcmin	GDP in millions of USD.
US Household Income	[140]	2000	30 arcsec	Aggregated household income in 2000, according to U.S. census.
US Population	[140]	2000	30 arcsec	U.S. population according to 2000 census.

(a) Details of the 15 attributes, showing data sources, year(s) of data collected, and grid size (GIS data resolution), followed by brief descriptions of the attributes.



(b) Correlations between attributes estimated from 100,000 image locations. Color intensity is proportional to the correlation magnitude, with positive values in blue and negative in pink.

Figure 6.1: Details of the data sources used and the correlation between attributes.

of our attribute classification problems such that it has two outputs (rather than the 1,000 of the original model) to account for our binary classification problem (estimating whether the attribute value is high or low). Additionally, the initial weights for FC8 are randomly sampled from a zero-mean normal distribution.

We used Caffe [61] for training and testing our networks, using the pretrained ImageNet network packaged with Caffe as initialization. The network for each classifier was trained independently using stochastic gradient descent with a batch size of 128 images. The learning rate was set at 0.001 and decreased by an order of magnitude every 2500 batches. The training process was allowed to continue for a maximum of 25,000 batches, but generally converged much earlier for our problems. To avoid

overfitting, the validation set was evaluated every 500 batches and the weights with the lowest validation error were used for testing.

To put the CNN results in context, we also tested classifiers using several other recognition approaches. We constructed a bag-of-words vocabulary using Histogram of Oriented Gradients (HOG) [27]; our hypothesis was that local evidence such as particular types of objects might be helpful to predict geospatial attributes. Given an image, we sample 5×5 blocks of HOG cells to produce local feature vectors in overlapping square sub-images. We use the 31-dimensional variant of HOG [39], so that the feature dimensionality of each patch is $5 \times 5 \times 31 = 775$. We represent the image in the standard bag-of-words fashion as a histogram over these features quantized to a vocabulary. In this case, we constructed a 100,000 codeword vocabulary by clustering (with k-means) over 10 million HOG features sampled from random Flickr images. We then learned a linear SVM [63] for each of our 15 attributes, to estimate a binary label indicating whether a given photo has a high or low value.

We also built simpler global scene-level features, under the hypothesis that some attributes could be inferred based on the overall appearance of a scene. We specifically used GIST [96] and spatially-pooled color histograms. For the histograms, we computed 8-bin histograms over each RGB plane within spatial regions of different sizes (specifically in a spatial pyramid with three levels of 1×1, 2×2, and 4×4 bins, yielding $(1 + 4 + 16) \times (3 \times 8) = 502$ dimensional feature vectors). As with the HOG features, we then learned linear SVMs.

The results of applying our classifiers on the 15 demographic and geographic attributes are shown in Table 6.1, where again the task is to determine whether each image was taken in a place with a high or low value of the attribute — e.g. for the

	# images	CNN	HOG	Color	GIST
Global attributes					
Elevation	14,230	**61.11**	56.50	53.34	52.92
Elevation Gradient	13,266	**60.63**	55.86	53.76	52.44
GDP, 1990 Actual	14,940	**71.33**	64.83	60.45	58.02
GDP, 2025 Predicted	14,906	**73.58**	66.05	61.79	59.26
Infant Mortality	14,634	**55.88**	52.88	52.63	50.92
Night Light Intensity	15,004	**73.61**	68.03	62.58	59.98
Population Density, 2010	14,840	**74.43**	67.48	62.05	60.11
Population Density, 2000	14,892	**72.38**	65.75	61.62	58.71
Underweight Children	1,896	**62.88**	51.72	51.72	51.55
% Pasture Land	14,972	**58.50**	54.62	54.54	52.99
U.S.-only attributes					
% African American	7,190	**65.42**	62.17	57.79	57.79
% Asian American	7,006	**63.02**	58.99	57.48	56.54
% Hispanic American	6,898	**65.65**	60.88	58.56	56.92
Household Income	6,900	**67.60**	64.55	58.12	57.61
Population	6,866	**68.10**	64.76	61.17	59.22
Average		**66.27**	61.00	57.84	56.33

Table 6.1: Classification accuracies as percentage correct for 15 geo-spatial attributes. Random baseline for all attributes is 50%; see text for human baselines.

first row of the table, whether a given photo was taken at a low or high elevation. We find that the correct classification rates vary significantly, from close to random guessing for infant mortality to nearly 75% for population density. This range reflects the difficulty of the attribute tasks we have proposed: a photo full of buildings and people is obviously probably taken in a high-population area, whereas inferring infant mortality (which is a good correlate for poverty rate) requires more subtle analysis (e.g. looking for bars on windows, or the clothes people are wearing). Some of these attributes are correlated and thus show similar performance, although we do see interesting differences amongst them: we can predict estimated GDP for 2025 more accurately than in 1990, presumably because Flickr images were mostly taken in the last 5 years, whereas the worldwide wealth distribution has changed dramatically since 1990 (e.g. China's GDP has increased by an order of magnitude). Figure 6.2 shows randomly-sampled correctly and incorrectly classified images for each attribute.

For all of the attributes, we found that the deep learning CNNs beat the other techniques by a decisive margin. The GIST and color features had an average accuracy of 56.33% and 57.84%, respectively, compared to a 50% random baseline. This confirms the hypothesis that some attributes can be (weakly) estimated based only on the overall properties of the scene. Using HOG features improved results significantly to 61.0%, suggesting that local object-level features help, but the CNNs yielded a dramatic further improvement to 66.3%. Our results thus add to the rapidly-growing evidence that deep learning can yield large improvements over traditional techniques on many vision problems.

We are not aware of other work that has studied demographic and geographic attribute classification directly, so we cannot compare against published results. Perhaps the closest work is that of Leung *et al* [81], who try to reconstruct land use maps by analyzing pools of geo-tagged photos from Flickr — a very different task than our goal of labeling images. Though not directly comparable, as a weak comparison we note that we achieve greater accuracy relative to our baseline: they report 64% accuracy versus a 61.1% random baseline for urban development classification, while we achieve 73.6% accuracy on our similar "brightness of lights at night" attribute versus 50%. Again, our tasks are very different so a direct comparison is not meaningful, but this at least suggests that our improvement over baseline is state-of-the-art.

Human Baselines

Although the automatic classifiers beat the random baseline by substantial margins, our accuracies are not near the 100% performance we might aspire to. However it is important to note that our test dataset is extremely difficult, consisting of a raw

Figure 6.2: Some correctly and incorrectly classified images. For each attribute, we show a correctly predicted high and low valued image inside the box, and an incorrectly predicted high and low valued image outside the box to the right.

set of Flickr photos; we have deliberately made no attempt to filter out difficult or noisy images (because doing so could inevitably inject biases into the dataset). Thus many of our test set photos, including indoor images and close-ups of objects, lack enough visual evidence to infer many attributes. Moreover, the ground truth labels themselves are noisy, as a significant fraction of Flickr geo-tags are wrong [53].

The sample of correctly and incorrectly classified images for each attribute shown in Figure 6.2 gives a sense for the difficulty of our dataset, and the limited amount of evidence that some images contain. For instance, in column one, row two of the figure, the classifier correctly estimates that the photo of a concert probably occurs in a city and the photo of a mountain is in a rural area. But it incorrectly decides that the fencers are in the country and the art is in the city, despite the fact that these are very reasonable decisions based on the visual evidence at hand.

To try to quantify the fraction of these difficult images, we collected annotations for three attributes (population, income, and elevation gradient) to measure human performance on these classification tasks.For each attribute, we sampled 1,000 images from our dataset such that half had a high value of the attributes and the other half had a low value according to the automatic labeling. We presented each image to two users on Mechanical Turk (restricting to "Masters" who have a long track record of quality work), asking them to classify the image into the low or high category and to provide some additional feedback.

We found that human performance ranged from 52.9% for poverty, to 60.0% for elevation gradient, to nearly 81% for population density. Our automatic classifiers actually beat human performance on poverty (taken as a proxy for infant mortality — 55.9% versus 52.9%), while achieving about the same performance on elevation gra-

dient (60.6% versus 60.0%). However the human users performed significantly better on population density (80.8% versus 73.61%). Thus while our automatic classifiers do not get near 100% accuracy, neither do humans. One reason for this is that about 28% of our dataset is indoor images, which typically have very little evidence about many of these attributes.

6.4 CONCLUSION AND FUTURE WORK

We have proposed the problem of estimating geographic and demographic attributes of the place where a photo was taken, based only on the photo's visual content. We learned Convolutional Neural Network classifiers for a wide variety of these attributes by training on large, automatically collected datasets created by combining geo-tagged Flickr photos with attribute values from GIS maps. We evaluated the performance of our models against more traditional scene-level and local features. While the CNNs give the best performance, we find that the local features outperform the simpler scene-level features by a significant degree, suggesting that the classifiers have discovered local features (like objects) that are predictive of attribute values.

CHAPTER 7

CONCLUSION AND FUTURE WORK

In this thesis, we have argued that computer vision techniques will be vital for many observational sciences to face the challenges of an increasingly visual data landscape and that computer vision techniques are already useful enough to be valuable in some academic domains. We have presented multiple lines of work developing automated image analysis techniques in both traditionally quantitative and qualitative academic contexts including glaciology, developmental psychology, architectural history, and demography. Our developed methodologies for these problems often perform better than human annotators, resulting in solutions that perform analysis at super-human accuracy, speed, and/or scale. Further, we develop on themes of diversity and confidence estimation through multiple lines of work. These successes provide additional evidence for the efficacy of computer vision techniques in the context of observational science and the humanities.

In Chapter 2, we described the models used in this thesis. In Chapter 3, we developed a holistic model for ice-layer identification which uses statistical sampling to consider thousands of potential high-probability configurations from the solution space to provide confidence intervals for layer estimates. In addition to being the first work on layer finding to introduce these notions of solution confidence, our method-

ology also achieves state-of-the-art performance. In Chapter 4, we presented two methods for hand detection and disambiguation for egocentric videos. We show that powerful spatial biases in egocentric videos can improve results in multiple settings and under multiple models. We employ a Markov Chain Monte Carlo method as well as Convolutional Neural Networks to provide high-quality hand detections in laboratory and natural environments. In Chapter 5, we harnessed tens of thousands of publicly available Google Street View images to automatically create a dataset of building facades correlated with construction period. We mined this dataset to produce hundreds of facade elements that are indicative of specific periods in Parisian architecture. The huge number of individual element instances in our dataset requires our mining approach to enforce diversity in candidate rankings. In Chapter 6, we demonstrated how large-scale analysis of consumer photography can be used to coarsely track changing demographic and geographic attributes.

Our work is part of a growing inter-disciplinary trend combining sophisticated machine learning with large-scale datasets to enable novel science. We believe that as time goes on, computer vision for large-scale analysis is likely to become an essential tool for the next generation of observational science. For tasks like ice-layer identification which have been traditionally posed as image processing tasks, introducing machine learning can have a broad impact for automated analysis, in response to growing volumes of visual data. There are also many opportunities to use machine learning to study human populations through large image collections such as studying dietary habits through shared images [90] or analyzing how our social systems affect health or educational outcomes by mining photo-sharing platforms.

Finally, as discussed in Section 2.2, no techniques have been established to produce

diverse solutions from Convolutional Neural Networks for structured prediction problems. This task is a rich line of work that warrants further investigation in order to bring greater application to these powerful learners to pipelined and user-in-the-loop systems. In the following section, we overview our recent ongoing work to address this gap and show some qualitative results.

7.1 FUTURE WORK

As discussed in Section 2.1, the most likely solution under a model may not be the lowest error solution for a test example. For large, difficult-to-optimize models like Convolutional Neural Networks (and other deep architectures), this can be especially true. Furthermore, ensembles of deep networks often converge to very similar solutions for structured problems, differing only by a few pixels for a segmentation task or a few words for image captioning. In recent work, we address the issue of learning ensembles of deep networks such that their outputs are likely to be diverse in the face of ambiguity, i.e. the loss with respect to an oracle mechanism (either a reranker or a human operator) is minimized [78]. We reproduce the methodology and some examples here as an additional contribution, though these techniques are not applied in this thesis elsewhere.

More formally, we consider the task of training an ensemble of differentiable learners that together produce a set of solutions with minimal loss with respect the an oracle that selects only the lowest-error prediction. We use $[n]$ to denote the set of $1, 2, \ldots, n$. Given a training set of input-output pairs $D = \{(x_i, y_i) \mid x_i \in \mathcal{X}, y_i \in \mathcal{Y}\}$, our goal is to learn a function $g : \mathcal{X} \rightarrow \mathcal{Y}^M$ which maps each input to M outputs. We fix the form of g to be an ensemble of M learners f such that $g(x) =$

$(f_1(x), \ldots, f_M(x))$. For some task-dependent loss $\ell(y, \hat{y})$, which measures the error between true and predicted outputs y and $\hat{y}$, we define the oracle loss of g over the dataset D as

$$\mathcal{L}_O(D) = \sum_{i=1}^{n} \min_{m \in [M]} \ell\left(y_i, f_m(x_i)\right).$$

In order to directly minimize the oracle loss for an ensemble of learners, Guzman-Rivera et al. [50] present an objective which forms a (potentially tight) upper-bound on the oracle loss. This objective replaces the minimum in the oracle loss with indicator variables $(p_{i,m})_{m=1}^{M}$, where $p_{i,m}$ is 1 if predictor m has the lowest error on example i. The resulting minimization,

$$\begin{aligned} \underset{f_m, p_{m,i}}{\text{argmin}} \quad & \sum_{i=1}^{n} \sum_{m=1}^{M} p_{i,m}\, \ell\left(y_i, f_m(x_i)\right) \\ \text{s.t.} \quad & \sum^{M} p_{i,m} = 1, \quad p_{i,m} \in \{0, 1\}, \end{aligned} \tag{7.1}$$

is a constrained joint optimization over ensemble parameters and data-point assignments. The authors propose an alternating block algorithm to approximately minimize this objective. In a manner similar to K-Means or 'hard-EM', this approach alternates between assigning examples to their min-loss predictors and training models to convergence on the partition of examples assigned to them. Note that this approach is incompatible with training deep networks, since modern architectures [55] can sometimes take weeks or months to train *a single model once*, repeatedly retraining all ensemble members may be simply infeasible.

To overcome this shortcoming, we propose a stochastic algorithm for differentiable learners which interleaves the assignment step with batch updates in stochastic gra-

dient descent (SGD). Consider the partial derivative of the objective in Eq. 7.1 with respect to the m^{th} individual learner on example x_i,

$$\frac{\partial \mathcal{L}_O}{\partial f_m(x_i)} = p_{i,m} \frac{\partial \ell(y_i, f_m(x_i))}{\partial f_m(x_i)}. \tag{7.2}$$

Notice that if f_m is the minimum error predictor for example x_i, then $p_{i,m} = 1$ and the gradient term is the same as if training a single model; otherwise, the gradient is zero. This behavior lends itself to a straightforward optimization strategy for learners trained by SGD-based solvers. For each batch, we pass the examples through the learners, calculating losses from each ensemble member for each example. During the backward pass, the gradient of the loss for each example is backpropagated only to the lowest error predictor on that example (with ties broken arbitrarily).

We call this approach Stochastic Multiple Choice Learning (sMCL). sMCL is generalizable to any learner trained by stochastic gradient descent and is thus applicable to an extensive range of modern deep networks. Unlike the iterative training schedule of MCL, sMCL ensembles need only be trained to convergence once in parallel. sMCL is also agnostic to the exact form of loss function ℓ such that it can be applied without additional effort on a variety of problems. Concretely, this can be implemented via a simple *sMCL loss layer* which can be dropped into any ensemble of any type of architecture after the final prediction layer in each member.

We find that ensembles trained with sMCL greatly reduce error with respect to an oracle for many tasks and network architectures. Figures 7.1 and 7.2 show example outputs for sMCL ensembles compared to independently trained ensembles for semantic segmentation and image captioning tasks. As the figure shows, sMCL en-

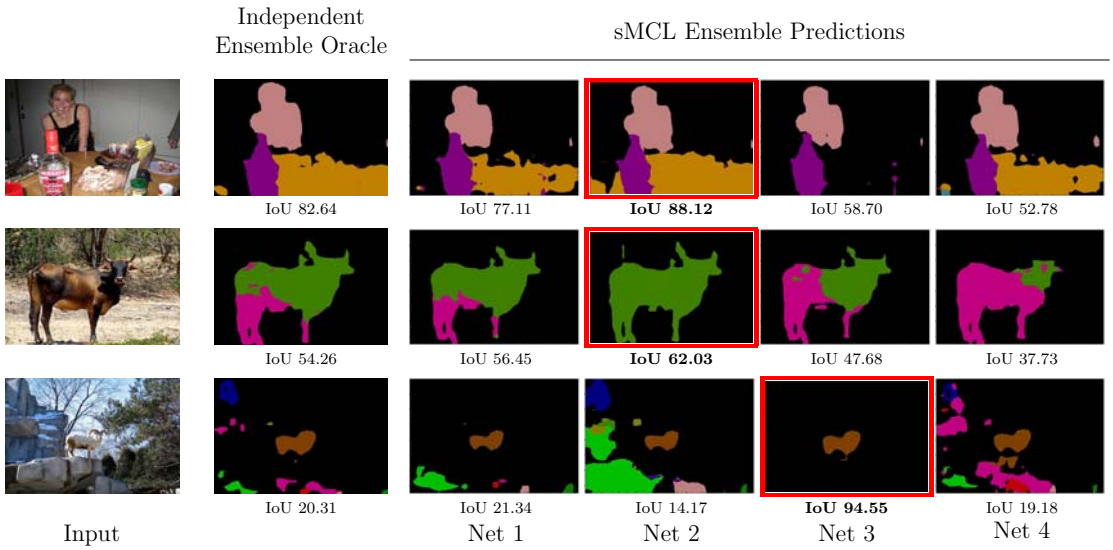


Figure 7.1: Test image and corresponding predictions obtained by each member of the sMCL ensemble as well as the top output of a classical ensemble. The outputs with minimum loss on each example are outlined in red and intersection over union is listed below each example. Notice that sMCL ensembles vary in the shape, class, and frequency of predicted segments.

sembles provide multiple reasonable hypotheses in the face of ambiguity. For example in the first row of Figure 7.1, we see the majority of the ensemble members produce dining tables of various completeness in response to the visual uncertainty caused by the clutter. Networks 2 and 3 capture this ambiguity well, producing segmentations with the dining table completely present or absent. For image captioning, we see independently trained networks producing nearly identical outputs due to the strong biases of the language model while the sMCL ensembles show diversity both in choice of language and in what parts of the image they describe.

We feel this and related work will allow deep networks to provide higher quality solution sets to both reranking systems and human operators. Our methodology is general, loss agnostic, and parameter free - allowing easy application to most modern deep architectures.

Input	Independently Trained Networks	sMCL Ensemble
	A man riding a wave on top of a surfboard. A man riding a wave on top of a surfboard. A man riding a wave on top of a surfboard. A man riding a wave on top of a surfboard.	A man riding a wave on top of a surfboard. A person on a surfboard in the water. A surfer is riding a wave in the ocean. A surfer riding a wave in the ocean.
	A kitchen with a stove and a microwave. A white refrigerator freezer sitting inside of a kitchen. A white refrigerator sitting next to a window. A white refrigerator freezer sitting in a kitchen	A cat sitting on a chair in a living room. A kitchen with a stove and a sink. A cat is sitting on top of a refrigerator. A cat sitting on top of a wooden table
	A living room filled with furniture and a flat screen tv. A living room filled with furniture and a flat screen tv. A living room filled with furniture and a window. A living room filled with furniture and a flat screen tv	A man sitting on a couch with a laptop. A living room with a couch and a table. The living room is clean and empty of people. A living room with a table and chairs
	A bird is sitting on a tree branch. A bird is perched on a branch in a tree. A bird is perched on a branch in a tree. A bird is sitting on a tree branch	A small bird perched on top of a tree branch. A couple of birds that are standing in the grass. A bird perched on top of a branch. A bird perched on a tree branch in the sky

Figure 7.2: Comparison of sentences generated by members of a standard independently trained ensemble, and an sMCL-based ensemble of size four.

7.2 SOME FINAL OVERLY-PHILOSOPHIC THOUGHTS

It seems that the information age is sparing fewer and fewer subjects, long held to be solely the domains of the human spirit, from the advance of autonomous agents. The techniques presented here are far from supplanting the expertise and intuition of those of us who dedicate our lives to singular studies, but should artificial intelligence ever complete this coup and permeate every aspect of our cultural and scientific systems, it will only be through the cooperation and curiosity of those who it would displace - a victory of obsolescence.

BIBLIOGRAPHY

[1] Hojjat Adeli and Ashif Panakkat. A probabilistic neural network for earthquake magnitude prediction. *Neural Networks*, 22(7):1018–1024, 2009.

[2] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2189–2202, 2012.

[3] C. Allen, L. Shi, R. Hale, C. Leuschen, J. Paden, B. Pazer, E. Arnold, W. Blake, F. Rodriguez-Morales, J. Ledford, et al. Antarctic ice depthsounding radar instrumentation for the NASA DC-8. *IEEE Aerospace and Electronic Systems Magazine*, 27(3):4–20, 2012.

[4] Dragomir Anguelov, Carole Dulong, Daniel Filip, Christian Frueh, Stéphane Lafon, et al. Google Street View: Capturing the world at street level. *Computer*, 43(6):32–38, 2010.

[5] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *IEEE International Conference on Computer Vision*, December 2015.

[6] Atelier Parisien d'Urbanisme. http://www.apur.org/.

[7] Sean Arietta, Alexei A. Efros, Maneesh Agrawala, and Ravi Ramamoorthi. City forensics: Using visual elements to predict non-visual city attributes. *IEEE Transactions on Visualization and Computer Graphics*, 2014.

[8] Mathieu Aubry, Bryan Russell, and Josef Sivic. Painting-to-3d model alignment via discriminative visual elements. *ACM Transactions On Graphics*, 33(2), 2014.

[9] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[10] Sven Bambach, David Crandall, and Chen Yu. Understanding embodied visual attention in child-parent interaction. In *International Conference on Development and Learning and Epigenetic Robotics*, 2013.

[11] Sven Bambach, John M Franchak, David J Crandall, and Chen Yu. Detecting hands in children's egocentric views to understand embodied attention during social interaction. In *Cognitive Science Society*, pages 134–139, 2014.

[12] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *IEEE International Conference on Computer Vision*, pages 1949–1957, 2015.

[13] Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua. Multiple object tracking using k-shortest paths optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[14] Alexander C Berg, Floraine Grabler, and Jitendra Malik. Parsing images of architectural scenes. In *IEEE International Conference on Computer Vision*, 2007.

[15] Julie Bort. Facebook stores 240 billion photos and adds 350 million more a day. In *Business Insider*, January 2013.

[16] George Casella and Edward George. Explaining the Gibbs sampler. *The American Statistician*, 46(3):167–174, 1992.

[17] Computational Cognition and Learning Laboratory. http://www.indiana.edu/ dll/.

[18] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *CoRR*, abs/1405.3531, 2014.

[19] Jixu Chen, Ming-Ching Chang, Tai-Peng Tian, Ting Yu, and Peter Tu. Bridging computer vision and social science: A multi-camera vision system for social interaction training analysis. In *IEEE International Conference on Image Processing*, pages 823–826. IEEE, 2015.

[20] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *CoRR*, abs/1412.7062, 2014.

[21] Nicolas Chenouard, Ihor Smal, Fabrice De Chaumont, Martin Maška, Ivo F Sbalzarini, Yuanhao Gong, Janick Cardinale, Craig Carthel, Stefano Coraluppi,

Mark Winter, et al. Objective comparison of particle tracking methods. *Nature methods*, 11(3):281, 2014.

[22] Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cudnn: Efficient primitives for deep learning. *CoRR*, abs/1410.0759, 2014.

[23] CIESIN/Columbia University. Poverty Mapping Project, 2005.

[24] CIESIN/Columbia University, and Centro Internacional de Agricultura Tropical (CIAT). Gridded Population of the World, Version 3 (GPWv3): Population Density Grid, 2005.

[25] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[26] David J Crandall, Geoffrey C Fox, and John D Paden. Layer-finding in radar echograms using probabilistic graphical models. In *IAPR International Conference on Pattern Recognition*, pages 1530–1533. IEEE, 2012.

[27] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[28] Adnan Darwiche. *Modeling and reasoning with Bayesian networks*. Cambridge University Press, 2009.

[29] Fabrice de Chaumont, Renata Dos-Santos Coura, Pierre Serreau, Arnaud Cressant, Jonathan Chabout, Sylvie Granon, and Jean-Christophe Olivo-Marin. Computerized video analysis of social interactions in mice. *Nature methods*, 9(4):410–417, 2012.

[30] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[31] Carl Doersch, Abhinav Gupta, and Alexei Efros. Mid-level visual element discovery as discriminative mode seeking. In *Advances in Neural Information Processing Systems*, 2013.

[32] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes Paris look like Paris? *ACM Transactions on Graphics*, 31(3), 2012.

[33] Thomas Laurence Evans. *Digital archaeology: bridging method and theory*. Psychology Press, 2006.

[34] Mark Fahnestock, Waleed Abdalati, S. Luo, and S Prasad Gogineni. Internal layer tracing and age-depth-accumulation relationships for the northern greenland ice sheet. *Journal of Geophysical Research*, 106(D24):33789–33, 2001.

[35] Yong Fan, Hengyi Rao, Hallam Hurt, Joan Giannetta, Marc Korczykowski, David Shera, Brian B Avants, James C Gee, Jiongjiong Wang, and Dinggang Shen. Multivariate examination of brain abnormality using both structural and functional mri. *NeuroImage*, 36(4):1189–1199, 2007.

[36] Alircza Fathi, Jessica K Hodgins, and James M Rehg. Social interactions: A first-person perspective. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1226–1233. IEEE, 2012.

[37] Alireza Fathi, Ali Farhadi, and James M Rehg. Understanding egocentric activities. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[38] Oliver Faust, Rajendra Acharya, EYK Ng, Kwan-Hoong Ng, and Jasjit S Suri. Algorithms for the automated detection of diabetic retinopathy using digital fundus images: a review. *Journal of medical systems*, 36(1):145–157, 2012.

[39] Pedro Felzenszwalb, Ross Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9), 2010.

[40] David Fleet and Yair Weiss. Optical flow estimation. In *Handbook of Mathematical Models in Computer Vision*, pages 237–257. Springer, 2006.

[41] John M Franchak, Kari S Kretch, Kasey C Soska, and Karen E Adolph. Head-mounted eye tracking: A new method to describe infant looking. *Child Development*, 82(6):1738–1750, 2011.

[42] Michael C. Frank. Measuring children's visual access to social information using face detection. In *Cognitive Science Society*, 2012.

[43] Greg J Freeman, Alan C Bovik, and John W Holt. Automated detection of near surface martian ice layers in orbital radar data. In *Image Analysis &*

Interpretation (SSIAI), 2010 IEEE Southwest Symposium on, pages 117–120. IEEE, 2010.

[44] Michael Gharbi, Tomasz Malisiewicz, Sylvain Paris, and Frédo Durand. A gaussian approximation of feature space for fast image similarity. *Technical report, MIT*, 2012.

[45] Christopher M Gifford, Gladys Finyom, Michael Jefferson, MyAsia Reid, Eric L Akers, and Arvin Agah. Automated polar ice thickness estimation from radar imagery. *IEEE Transactions on Image Processing*, 19(9):2456–2469, 2010.

[46] Jeremy Ginsberg, Mohebbi Mohebbi, Rajan Patel, Lynnette Brammer, Mark Smolinski, and Larry Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, 457:1012–1014, 2009.

[47] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587. IEEE, 2014.

[48] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 315–323, 2011.

[49] Peter J Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[50] Abner Guzman-Rivera, Dhruv Batra, and Pushmeet Kohli. Multiple choice learning: Learning to produce multiple structured outputs. In *Advances in Neural Information Processing Systems*, pages 1799–1807, 2012.

[51] Qiang Hao, Rui Cai, Zhiwei Li, Lei Zhang, Yanwei Pang, and Feng Wu. 3d visual phrases for landmark recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3594–3601. IEEE, 2012.

[52] Bharath Hariharan, Jitendra Malik, and Deva Ramanan. Discriminative decorrelation for clustering and classification. In *European Conference on Computer Vision*, 2012.

[53] Claudia Hauff. A study on the accuracy of Flickr's geotag data. In *Special Interest Group on Information Retrieval*, 2013.

[54] James Hays and Alexei A. Efros. IM2GPS: estimating geographic information from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[55] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[56] Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. *The annals of statistics*, pages 1171–1220, 2008.

[57] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[58] Alexander Ihler. Kernel Density Estimation (KDE) Toolbox for Matlab. http://www.ics.uci.edu/ ihler/code/kde.html.

[59] Ana-Maria Ilisei, Adamo Ferro, and Lorenzo Bruzzone. A technique for the automatic estimation of ice thickness and bedrock properties from radar sounder data acquired at antarctica. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 4457–4460. IEEE, 2012.

[60] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[61] Yangqing Jia. Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/, 2013.

[62] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[63] Thorsten Joachims. Making large-scale support vector machine learning practical. In *Advances in Kernel Methods*, 1999.

[64] Charles Kervrann, Carlos Óscar Sánchez Sorzano, Scott T Acton, Jean-Christophe Olivo-Marin, and Michael Unser. A guided tour of selected image processing and analysis methods for fluorescence and electron microscopy. *IEEE Journal of Selected Topics in Signal Processing*, 10(1):6–30, 2016.

[65] Daniel Kim, Seung Son, and Hawoong Jeong. Large-scale quantitative analysis of painting arts. *Scientific Reports*, 4(7370), 2014.

[66] Ross Kindermann, James Laurie Snell, et al. *Markov random fields and their applications*, volume 1. American Mathematical Society Providence, RI, 1980.

[67] Sara Klingenstein, Tim Hitchcock, and Simon DeDeo. The civilizing process in London's Old Bailey. *Proceedings of the National Academy of Science*, 111(26), 2014.

[68] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[69] David Koller, Bernard Frischer, and Greg Humphreys. Research challenges for digital archives of 3d cultural heritage models. *Journal on Computing and Cultural Heritage (JOCCH)*, 2(3):7, 2009.

[70] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[71] David Lazer, Ryan Kennedy, Gary King, and Alessandro Vespignani. The Parable of Google Flu: Traps in big data analysis. *Science*, 343(6176), 2014.

[72] David Lazer, Alex Pentland, Lada Adamic, Sinan Aral, Albert-Laszlo Barabasi, et al. Computational social science. *Science*, 323(5915):721–723, February 2009.

[73] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[74] Sang-Rim Lee, Jerome Mitchell, David J Crandall, and Geoffrey C Fox. Estimating bedrock and surface layer boundaries and confidence intervals in ice sheet radar imagery using mcmc. In *IEEE International Conference on Image Processing*, pages 111–115. IEEE, 2014.

[75] Stefan Lee, Sven Bambach, David Crandall, John Franchak, and Chen Yu. This hand is my hand: A probabilistic approach to hand disambiguation in egocentric video. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, pages 543–550, 2014.

[76] Stefan Lee, Sven Bambach, David J. Crandall, John M. Franchak, and Chen Yu. This hand is my hand: A probabilistic approach to hand disambiguation in egocentric video. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, June 2014.

[77] Stefan Lee, Nicolas Maisonneuve, David Crandall, Alexei A Efros, and Josef Sivic. Linking past to present: Discovering style in two centuries of architecture. In *IEEE International Conference on Image Processing*.

[78] Stefan Lee, Senthil Purushwalkam, Michael Cogswell, Viresh Ranjan, David Crandall, and Dhruv Batra. Stochastic multiple choice learning for training diverse deep ensembles.

[79] Stefan Lee, Haipeng Zhang, and David J Crandall. Predicting geo-informative attributes in large-scale image collections using convolutional neural networks. In *IEEE Winter Conference on Applications of Computer Vision*, pages 550–557. IEEE, 2015.

[80] Yong Jae Lee, Alexei A Efros, and Martial Hebert. Style-aware mid-level representation for discovering visual connections in space and time. In *IEEE International Conference on Computer Vision*, 2013.

[81] Daniel Leung and Shawn Newsam. Proximate sensing: Inferring what-is-where from georeferenced photo collections. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[82] Cheng Li and Kris M. Kitani. Model recommendation with virtual probes for egocentric hand detection. In *IEEE International Conference on Computer Vision*, 2013.

[83] Cheng Li and Kris M. Kitani. Pixel-level hand detection in ego-centric videos. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[84] Xiaowei Li, Changchang Wu, Christopher Zach, Svetlana Lazebnik, and Jan-Michael Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In *European Conference on Computer Vision*, 2008.

[85] Yunpeng Li, David J Crandall, and Daniel P Huttenlocher. Landmark classification in large-scale image collections. In *IEEE International Conference on Computer Vision*, pages 1957–1964, 2009.

[86] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

[87] Yizhou Lin, Gang Hua, and Philippos Mordohai. Egocentric object recognition leveraging the 3d shape of the grasping hand. In *European Conference on Computer Vision*, pages 746–762. Springer, 2014.

[88] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[89] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[90] Austin Meyers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin P Murphy. Im2calories: towards an automated mobile vision food diary. In *IEEE International Conference on Computer Vision*, pages 1233–1241, 2015.

[91] Jean-Baptiste Michel, Yuan Shen, Aviva Aiden, Adrian Veres, Matthew Gray, et al. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014), 2011.

[92] Jerome E Mitchell, David J Crandall, Geoffrey C Fox, and John D Paden. A semi-automatic approach for estimating near surface internal layers from snow radar imagery. In *2013 IEEE International Geoscience and Remote Sensing Symposium-IGARSS*, pages 4110–4113. IEEE, 2013.

[93] Guido F Montufar, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio. On the number of linear regions of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 2924–2932. 2014.

[94] John R Napier. The prehensile movements of the human hand. *Journal of Bone and Joint Surgery*, 38(4):902–913, 1956.

[95] NOAA National Geophysical Data Center. Version 4 DMSP-OLS nighttime lights time series.

[96] Aude Oliva and Antonio Torralba. Building the gist of a scene: the role of global image features in recognition. In *Progress in Brain Research*, 2006.

[97] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1717–1724, 2014.

[98] Vicente Ordonez and Tamara L Berg. Learning high-level judgments of urban perception. In *European Conference on Computer Vision*, 2014.

[99] Christian Panton. Automated mapping of local layer slope and tracing of internal layers in radio echograms. *Annals of Glaciology*, 55(67):71–77, 2014.

[100] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[101] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 2014.

[102] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[103] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[104] Rahul Raguram, Joseph Tighe, and Jan-Michael Frahm. Improved geometric verification for large scale landmark image collections. In *British Machine Vision Conference*, pages 1–11, 2012.

[105] Navin Ramankutty, Amato T Evan, Chad Monfreda, and Jonathan A Foley. Farming the planet: 1. geographic distribution of global agricultural lands in the year 2000. *Global Biogeochemical Cycles*, 22(1), 2008.

[106] James Rehg, Gregory Abowd, Agata Rozga, Mario Romero, Mark Clements, Stan Sclaroff, Irfan Essa, O Ousley, Yin Li, Chanho Kim, et al. Decoding children's social behavior. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3414–3421, 2013.

[107] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

[108] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[109] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004.

[110] Derek Ruths and Jürgen Pfeffer. Social media for large scale studies of behavior. *Science*, 346, 2014.

[111] Marcel Salathé, Linus Bengtsson, Todd Bodnar, Devon Brewer, John Brownstein, et al. Digital epidemiology. *PLOS Computational Biology*, 8(7), 2012.

[112] Maximilian Schich, Chaoming Song, Yong-Yeol Ahn, Alexander Mirksy, Mauro Martino, et al. A network framework of cultural history. *Science*, 345(6196), 2014.

[113] Grant Schindler and Frank Dellaert. 4d cities: analyzing, visualizing, and interacting with historical urban photo collections. *Journal of Multimedia*, 7(2):124–131, 2012.

[114] L Seirup and G Yetman. Us census grids, 2000. 2006.

[115] Giuseppe Serra, Marco Camurri, Lorenzo Baraldi, Michela Benedetti, and Rita Cucchiara. Hand segmentation for gesture recognition in ego-vision. In *Proceedings of the 3rd ACM International Workshop on Interactive mMltimedia on Mobile & Portable Devices*, pages 31–36. ACM, 2013.

[116] Gayane Shalunts, Yll Haxhimusa, and Robert Sablatnig. Architectural style classification of building facade windows. In *Advances in Visual Computing*, pages 280–289, 2011.

[117] Gayane Shalunts, Yll Haxhimusa, and Robert Sablatnig. Architectural style classification of domes. In *Advances in Visual Computing*, pages 420–429. Springer, 2012.

[118] Louise C Sime, Richard CA Hindmarsh, and Hugh Corr. Instruments and methods automated processing to derive dip angles of englacial radar reflectors in ice sheets. *Journal of Glaciology*, 57(202):260–266, 2011.

[119] Loic Simon, Olivier Teboul, Panagiotis Koutsourakis, Luc Van Gool, and Nikos Paragios. Parameter-free/pareto-driven procedural 3d reconstruction of buildings from ground-level sequences. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[120] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Modeling the World from Internet Photo Collections. *International Journal of Computer Vision*, 80(2), 2008.

[121] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[122] Carsten Steger. An unbiased detector of curvilinear structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):113–125, 1998.

[123] David Stork. Computer vision and computer graphics analysis of paintings and drawings: An introduction to the literature. In *Computer Analysis of Images and Patterns*, 2009.

[124] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.

[125] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[126] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[127] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. *arXiv preprint arXiv:1312.4659*, 2013.

[128] Yves Ubelmann. *Personal communication*, Dec 2014.

[129] Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013.

[130] United States Geological Survey. Global 30 arc-second elevation (GTOPO30).

[131] Jochem Verrelst, Gustau Camps-Valls, Jordi Muñoz-Marí, Juan Pablo Rivera, Frank Veroustraete, Jan GPW Clevers, and José Moreno. Optical remote sensing and the retrieval of terrestrial vegetation bio-geophysical properties–a review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 108:273–290, 2015.

[132] Oriol Vinyals and Quoc Le. A neural conversational model. *arXiv preprint arXiv:1506.05869*, 2015.

[133] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[134] Jingya Wang, Mohammed Korayem, and David Crandall. Observing the natural world with flickr. In *IEEE International Conference on Computer Vision Workshop*, pages 452–459, 2013.

[135] Julien Weissenberg, Hayko Riemenschneider, Mukta Prasad, and Luc Van Gool. Is there a procedural logic to architecture? In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[136] Paul J Werbos. Applications of advances in nonlinear sensitivity analysis. In *System modeling and optimization*, pages 762–770. Springer, 1982.

[137] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[138] Ling Xie and Shawn Newsam. IM2MAP: deriving maps from georeferenced community contributed photo collections. In *SIGMM International Workshop on Social Media*, 2011.

[139] Zhe Xu, Dacheng Tao, Ya Zhang, Junjie Wu, and Ah Chung Tsoi. Architectural style classification using multinomial latent logistic regression. In *European Conference on Computer Vision*, 2014.

[140] G. Yetman, S. R. Gaffin, and X. Xing. Global 15x15 Minute Grids of the Downscaled GDP Based on the SRES B2 Scenario, 2004.

[141] Xiaoyuan Yu, Jiangping Wang, Roland Kays, Patrick A Jansen, Tianjiang Wang, and Thomas Huang. Automated identification of animal species in camera trap images. *EURASIP Journal on Image and Video Processing*, 2013(1):1–10, 2013.

[142] José Zariffa and Milos R Popovic. Hand contour detection in wearable camera video using an adaptive histogram region of interest. *Journal of neuroengineering and rehabilitation*, 10(1):1, 2013.

[143] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, 2014.

[144] Haipeng Zhang, Mohammed Korayem, David J Crandall, and Gretchen LeBuhn. Mining photo-sharing websites to study ecological phenomena. In *International World Wide Web Conference*, pages 749–758. ACM, 2012.

[145] Yan Zheng, Ming Zhao, Yang Song, Hartwig Adam, Ulrich Buddemeier, Alessando Bissacco, Fernando Brucher, Tat Chua, and Hartmut Neven. Tour

the world: building a web-scale landmark recognition engine. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[146] Bolei Zhou, Liu Liu, Aude Oliva, and Antonio Torralba. Recognizing city identity via attribute analysis of geo-tagged images. In *European Conference on Computer Vision*, 2014.

CURRICULUM VITAE

Education

Indiana University, Bloomington, IN

Ph.D., Computer Science, August 2016

M.S., Computer Science, May 2013

University of West Florida, Pensacola, FL

B.S., Computer Science, August 2011

Research Positions

Machine Learning & Perception Group - Virginia Tech

Visiting researcher working under Dr. Dhruv Batra, Fall 2015.

INRIA WILLOW - L'École Normale Superiéure & UC Berkeley

Visiting researcher working under Dr. Josef Sivic and Dr. Alexia Efros, Summer 2014.

IU Computer Vision Lab - Indiana University

PhD student under Dr. David J. Crandall, Fall 2012 to Present.

Peer Reviewed Conference Papers

1) Sven Bambach, **Stefan Lee**, David Crandall, and Chen Yu, Lending A Hand: Detecting Hands and Recognizing Activities in Complex Egocentric Interactions. *IEEE International Conference on Computer Vision (ICCV)*, 2015.

2) **Stefan Lee**, Nicolas Maisonneuve, David Crandall, Josef Sivic, and Alexei A. Efros. Linking Past to Present: Discovering Style in Two Centuries of Architecture. *IEEE International Conference on Computational Photography (ICCP)*, 2015.

3) **Stefan Lee**, Haipeng Zhang, and David Crandall. Predicting Geo-informative Attributes in Large-scale Image Collections using Convolutional Neural Networks. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2015.

4) **Stefan Lee**, Sven Bambach, David Crandall, John Franchak, and Chen Yu. This Hand Is My Hand: A Probabilistic Approach to Hand Disambiguation in Egocentric Video. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Workshop on Egocentric Vision,* 2014.

5) **Stefan Lee**, Jerome Mitchell, David Crandall, and Geoffery Fox. Estimating Bedrock and Surface Layer Boundaries and Confidence Intervals in Ice Sheet Radar Imagery Using MCMC. *IEEE International Conference on Image Processing (ICIP),* 2014.

Book Chapters

1) David J. Crandall, Yunpeng Li, **Stefan Lee**, and Daniel P. Huttenlocher. "Recognizing Landmarks in Large-Scale Social Image Collections." *Large-Scale Visual Geo-Localization.* Ed. Amir R. Zamir, Asaad Hakeem, Luc Van Gool, Mubarak Shah, Richard Szeliski. Springer, 2016.

Extended Abstracts

1) Sven Bambach, **Stefan Lee**, David Crandall, John Franchak, and Chen Yu. Tracking Hands of Interacting People in Egocentric Video. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Workshop on Observing and Understanding Hands in Action*, 2015.

2) **Stefan Lee**, Haipeng Zhang, and David Crandall. Predicting Geo-informative At-

tributes in Large-scale Image Collections using Convolutional Neural Networks. *IEEE International Conference on Computer Vision (ICCV), Workshop on Web-scale Vision and Social Media (VSM)*, 2015.

3) **Stefan Lee**, Nicolas Maisonneuve, David Crandall, Josef Sivic, and Alexei A. Efros. Linking Past to Present: Discovering Style in Two Centuries of Architecture. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Workshop on Large Scale Visual Recognition and Retrieval*, 2015.
4) **Stefan Lee** and David Crandall. Learning to Identify Local Floral with Human Feedback. *IEEE Conference on Computer Vision and Pattern Recognition, Workshop on Computer Vision and Human Computation,* 2014.
5) Sven Bambach, **Stefan Lee**, David Crandall, and Chen Yu. Analyzing Hands to Recognize Social Interactions with a Large-scale Egocentric Hands Dataset, Workshop on Observing and Understanding Hands in Action, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
6) Sven Bambach, **Stefan Lee**, David Crandall, and Chen Yu. Detecting and Classifying Hands in Social and Driving Contexts, Vision for Intelligent Vehicles and Applications (VIVA) Challenge and Workshop, IEEE Intelligent Vehicles Symposium, 2015.
7) Sven Bambach, **Stefan Lee**, David Crandall, John Franchak, and Chen Yu. Tracking Hands of Interacting People in Egocentric Video, Workshop on Observing and Understanding Hands in Action, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

Technical Reports

1) **Stefan Lee**, Senthil Purushwalkam, Michael Cogswell, David J. Crandall, and Dhruv Batra. Why M Heads are Better than One: Training a Diverse Ensemble of Deep Networks. *arXiv*, 2015.

2) Bingjing Zhang, Judy Qiu, **Stefan Lee**, and David Crandall. Large-Scale Image Classification using High Performance Clustering. *Indiana University Department of Computer Science Technical Report*, 2013.

Service

1) Co-organizer for *Diversity meets Deep Networks - Inference, Ensemble Learning, and Applications* tutorial co-located with CVPR 2016 along with Alexander Kirillov, Bogdan Savchynskyy, Carsten Rother, and Dhruv Batra.

Awards

- Bradley Postdoctoral Fellowship (Virginia Tech) 2016
- CVPR HANDS Workshop Travel Award 2016
- Heidelberg Laureate Forum Acceptance (HLF Foundation) 2015
- Dissertation Development Award (Indiana University) 2015
- Doctoral Consortium Travel Award (Int'l. Conference on Computer Vision) 2015
- Intel Best Paper Award (Workshop on Egocentric Vision) 2014

Teaching Experience

- B659 - Image Processing and Recognition - Assistant Instructor Spring 2015
- I399 - Research Methods for Informatics and Computing - Mentor Fall 2013
- C211 - Introduction to Computer Science - Assistant Instructor Fall 2011 - Spring 2013