1-1-2016

Monocular 3d Object Recognition

Menglong Zhu
University of Pennsylvania, menglong@cis.upenn.edu

Monocular 3d Object Recognition

Abstract
Object recognition is one of the fundamental tasks of computer vision. Recent advances in the field enable reliable 2D detections from a single cluttered image. However, many challenges still remain. Object detection needs timely response for real world applications. Moreover, we are genuinely interested in estimating the 3D pose and shape of an object or human for the sake of robotic manipulation and human-robot interaction.

In this thesis, a suite of solutions to these challenges is presented. First, Active Deformable Part Models (ADPM) is proposed for fast part-based object detection. ADPM dramatically accelerates the detection by dynamically scheduling the part evaluations and efficiently pruning the image locations. Second, we unleash the power of marrying discriminative 2D parts with an explicit 3D geometric representation. Several methods of such scheme are proposed for recovering rich 3D information of both rigid and non-rigid objects from monocular RGB images. (1) The accurate 3D pose of an object instance is recovered from cluttered images using only the CAD model. (2) A global optimal solution for simultaneous 2D part localization, 3D pose and shape estimation is obtained by optimizing a unified convex objective function. Both appearance and geometric compatibility are jointly maximized. (3) 3D human pose estimation from an image sequence is realized via an Expectation-Maximization algorithm. The 2D joint location uncertainties are marginalized out during inference and 3D pose smoothness is enforced across frames.

By bridging the gap between 2D and 3D, our methods provide an end-to-end solution to 3D object recognition from images. We demonstrate a range of interesting applications using only a single image or a monocular video, including autonomous robotic grasping with a single image, 3D object image pop-up and a monocular human MoCap system. We also show empirical start-of-art results on a number of benchmarks on 2D detection and 3D pose and shape estimation.

Degree Type
Dissertation

Degree Name
Doctor of Philosophy (PhD)

Graduate Group
Computer and Information Science

First Advisor
Kostas Daniilidis

Keywords
3D object, computer vision, object recognition

Subject Categories
Computer Sciences

MONOCULAR 3D OBJECT RECOGNITION

Menglong Zhu

A DISSERTATION

in

Computer and Information Science

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

2016

Kostas Daniilidis, Professor
Computer and Information Science
Supervisor of Dissertation

Lyle Ungar, Professor
Computer and Information Science
Graduate Group Chairperson

Dissertation Committee

Jianbo Shi, Professor
Computer and Information Science
University of Pennsylvania

Daniel Lee, Professor
Computer and Information Science
University of Pennsylvania

Camillo J. Taylor, Professor
Computer and Information Science
University of Pennsylvania

Silvio Savarese, Assistant Professor
Computer Science
Stanford University

MONOCULAR 3D OBJECT RECOGNITION

Menglong Zhu

To Mom, Dad and Lingli,

For always encouraging me to pursuit my dreams,

and for always be by my side :)

Acknowledgments

It couldn't be more enjoyable! I cherish every moment of the five-year graduate study pursuing a PhD with my advisor Kostas Daniilidis. I came to Penn with a deep fascination about computer vision and robotics. GRASP Lab turned out to be the perfect place to fulfill my dreams. Surrounded by inspiring faculty, brilliant colleagues, awesome robots, there is no better place I could imagine.

There are so many things I would like to thank my advisor Kostas Daniilidis. He means much more than a perfect advisor to me. I am still very grateful for him persuading me to pursuit a PhD at the time I was enrolled as a Masters student in the Robotics program. As I look back, this is one of those rare occasions that changes one's life, inviting me onto the journey of keeping chasing my passion in the field at the time I wasn't so sure I could do it. I would also like to thank him for being a great advisor, being knowledgeable and inspiring and super supportive and encouraging throughout the years. I learned so much about geometry from Kostas and it became the key element in this PhD thesis. I'm very grateful to have the opportunity to work with him. And I am really excited to see our group growing so much these years (Figure 1)!

I had a great time with Kosta Derpanis, when he was a post-doc in our group. He helped me kick-starting my research in the early years at Penn. Kosta is also very kind and supportive and he is more like a big brother to me. I really enjoy his accessibility for discussing detailed technical questions and I really learned a lot from him about research and paper writing. His dedication in research has always been my role model. Kosta, I still owe you a visit to Toronto.

I would like to thank Jianbo Shi for his inspiring Computer Vision course CIS581. That was my first rigorous treatment on the topic of computer vision. I learned so much in that class about vision and gaining practical skills with MATLAB. I can still vividly remember the refreshing sunrise I saw after spending the whole night at my teammate's place coding up the Canny edge detector.

I would also want to express my gratitude to the late Ben Taskar. His Machine Learning class provided detailed analysis via rigorous derivation and practical problems that still benefits me today. In addition to the knowledge about machine learning, I was more than happy to win the class final competition in image based gender classification.

Many other professors have also left me a great deal of influence through lectures and collaborations. Jean Gallier, thanks for the manifold class and the awesome jokes: I still remember you saying the way to deal with debt is to take logdet! Thank you Vijay Kumar, for the robotic class teaching so many things about planning, manipulation and quadrotor control. Thank you CJ Taylor for your comments on my WPEII and collaboration in RCTA. Thank you Dan Lee for leading the RoboCup team reaching world champion multiple times. Thank you professor Silvio Savarese, your group's research has always been inspiring.

Apart from the fantastic faculty of GRASP Lab, I have worked with so many brilliant colleagues. Xiaowei, thank you for teaching me so much about convex optimization, being a great friend to chat, to play tennis and soccer. Best of luck in your application to TsingHua University! Yida, thanks for staying late night together in lab debugging, grabbing fried chicken wings at 2AM. Weiyu, thanks for showing me tricks of fancy MATLAB plotting, biking together all the way from Princeton to Philly and always inviting me over to dinner parties. Nikolay, thanks for introducing me to active learning and the world of Texas Holdem. Ben Cohen, thank you so much for teaching me about ROS and PR2. Yash, 30 under 30, you rock! Keep your awesome work. Samarth, best wishes for your graduate study at Georgia Tech. Thank you PR2, being an awesome robot, but just remember who taught you to read.

Figure 1: The amazing Kostas' research group (with family and friends)!

Thanks to all my friends, who made my years in Philadelphia much more colorful! And special thanks to Lingli, for all the great memories of us exploring cities, climbing mountains and diving in the oceans, and most importantly for keeping me alive and smiling!

ABSTRACT

MONOCULAR 3D OBJECT RECOGNITION

Menglong Zhu

Kostas Daniilidis

Object recognition is one of the fundamental tasks of computer vision. Recent advances in the field enable reliable 2D detections from a single cluttered image. However, many challenges still remain. Object detection needs timely response for real world applications. Moreover, we are genuinely interested in estimating the 3D pose and shape of an object or human for the sake of robotic manipulation and human-robot interaction.

In this thesis, a suite of solutions to these challenges is presented. First, Active Deformable Part Models (ADPM) is proposed for fast part-based object detection. ADPM dramatically accelerates the detection by dynamically scheduling the part evaluations and efficiently pruning the image locations. Second, we unleash the power of marrying discriminative 2D parts with an explicit 3D geometric representation. Several methods of such scheme are proposed for recovering rich 3D information of both rigid and non-rigid objects from *monocular* RGB images. (1) The accurate 3D pose of an object instance is recovered from cluttered images using only the CAD model. (2) A global optimal solution for simultaneous 2D part localization, 3D pose and shape estimation is obtained by optimizing a unified convex objective function. Both appearance and geometric compatibility are jointly maximized. (3) 3D human pose estimation from an image sequence is realized via an Expectation-Maximization algorithm. The 2D joint location uncertainties are marginalized out during inference and 3D pose smoothness is enforced across frames.

By bridging the gap between 2D and 3D, our methods provide an end-to-end solution to 3D object recognition from images. We demonstrate a range of interesting applications using only a single image or a monocular video, including autonomous robotic grasping with a single image, 3D object image pop-up and a monocular human MoCap system. We also show empirical start-of-art results on a number of benchmarks on 2D detection and 3D pose and shape estimation.

Contents

List of Tables

List of Figures

Part I

Introduction and Related Work

Chapter 1

Introduction

> There is no royal road to geometry [other than the *Elements*].
>
> — Euclid, on if there is an easier way of learning geometry.

The idea of autonomous robots being able to interact with its surroundings, serving human beings in various tasks, dates back to the ancient Greek mythologies. The depiction of the mechanical servants built by the Greek god Hephaestus is one of the earliest expressions of such dream of human kind. This dream has long been fascinating and nowadays extended far beyond the form of mere humanoid robot servants but as autonomous vehicles, flying drones, Internet connected devices and smart wearable gadgets, etc. A key component of such intelligent robotic systems lies in their capacities of 3D object recognition, i.e., the ability to *localize, identify and infer the 3D geometry* of massive amount of different objects including us, human, in an unknown environment.

Object recognition is one of the holy grails of computer vision and a vital step towards true intelligence. Much has been achieved over the past decade within the computer vision community in terms of identifying objects in 2D imageries. As soon as massive image exemplars became available on the Internet and through industrious annotation (Deng et al.,

2009; Everingham et al., 2010), the community has harnessed fruitful results as the state of the art in detecting object categories has improved dramatically (Felzenszwalb et al., 2010b; Girshick et al., 2014).

However, recognition means more than just high precision and recall in detection or classification in 2D. The ultimate goal of recognition is to enable robots to take meaningful actions in 3D. Both accuracy, efficiency and the level of details captured by the vision systems are crucial. Firstly, fast and robust object recognition assures the robots to perform tasks reliably in a reasonable amount of time without appearing dumb or overthinking. In general, there exists a trade-off between efficiency and accuracy, given limited computational resource. It is meaningful to consider striking a balance between these two factors and build a fast system that only tolerances bounded error. Secondly and more importantly, the images are flat but our world is not. It is the reasoning of detailed object 3D geometry that lays the foundation of robotic interactions, such as manipulation, in the real world. The link between 2D and 3D recognition is not yet well established within the community. Designing algorithms that extend beyond recognition in 2D into estimating the 3D information of the objects from images is thus desirable.

Despite the availability of 3D sensors such as laser scanners, stereo cameras and infrared RGB-D sensors, the goal of recovering 3D geometry out of 2D RGB imaginary is still extremely appealing not only intellectually but also for practical reasons. The problem of recovering 3D from 2D is ill-posed because of the lost of dimensionality during image formation. Nonetheless, having a strong prior of the world, human are capable of perceiving the 3D geometry with only one eye. Thus, although remaining a challenging problem, it still seems viable for machines. In addition, the existing billions of images and videos created by consumer cameras and smart phones provide a valuable source of data for developing machine intelligence in preparing for real world scenarios. The exciting and perceivable future with augmented reality devices, autonomous cars and drones calls for smarter computer vision systems that can reason the 3D structure from images efficiently with the minimum sensor load.

Perhaps the most intriguing aspect of the quest is that bridging the gap between 2D and 3D recognition requires more than just learning and matching patterns but jointly reasoning with the prior knowledge of the 3D world geometry. That draws analogy to the actual human thinking process (Kahneman, 2011), in which both the instinct system – fast and mostly pattern matching – and the logical system – slow but deliberate – work cohesively together for a final decision. The hope here is to peak into the essence of true intelligence following such direction. With all the challenges present and fascinating future awaits, 3D object recognition is a key step to take along the journey towards realizing the dream of intelligent robots.

1.1 Problem Statement

Problem 1 (Monocular 3D Object Recognition)**.**

Input: *A single* RGB *image or* RGB *video sequence of arbitrary scene. The images could contain arbitrary number of objects including human.*

Output: *(1) 2D bounding boxes indicating the detected objects and human; (2) 3D orientation, translation and surface mesh model of the objects; 3D orientation, translation and skeletal pose of human.*

Complexity: *Polynomial computational time and space in the number of input pixels and number of output detections and number of object categories.*

More specifically, the problem can be split into the following two separate tasks:

- 2D Detection: identify and localize a particular object.
- 3D Recognition: estimate detailed 3D geometric information of the detected objects.

The input images can contain arbitrary scenes with or without objects. In the task of detection, the objects of interest include any categories of objects with bounding box

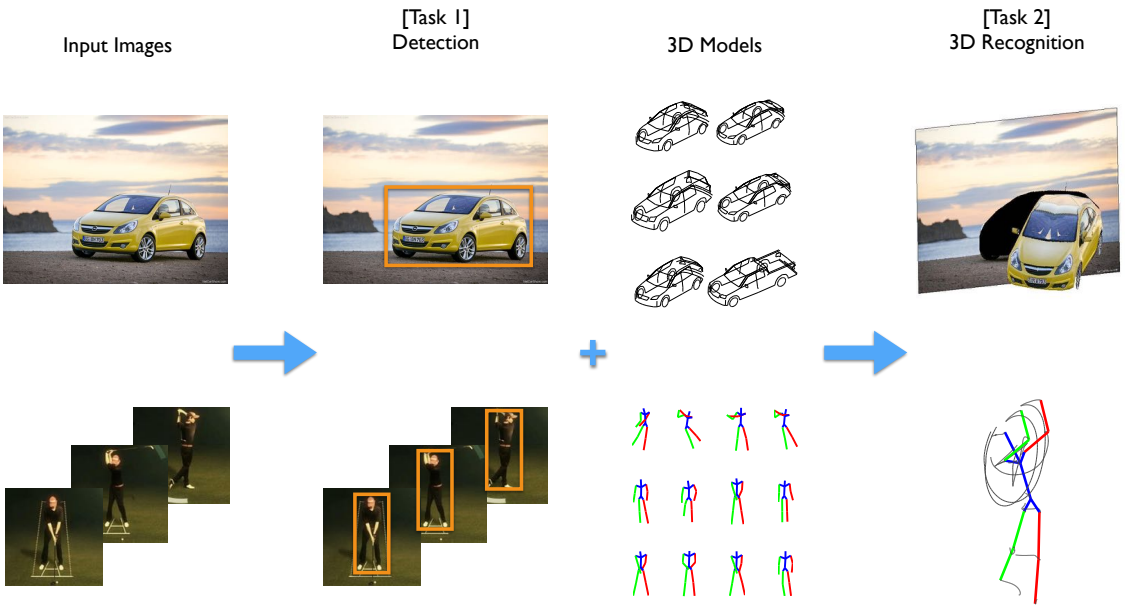


Figure 1.1: Overview of the 3D object recognition problem. Column-wise from left to right: (1) the input image or image sequence; (2) the first task of the problem is to detect the objects and output 2D bounding boxes indicating the location and the scale of the detections; (3) the 3D models of the objects are given; (4) the second task of the problem is to estimate 6-DOF pose and 3D shape mesh model for objects and 3D skeletal pose for human.

annotated training examples available. The algorithm should output the same number of detections as the objects present in the image. If there is no object of interest present, the output of the algorithm should be empty. In the task of recognition, the objects are assumed to be detected and roughly localized. It is also assumed that the 3D object CAD models are available, or the 3D skeleton models in the case of human. The goal of the algorithms is to reason about the 3D geometry and output both 6-DOF (degree of freedom) pose and 3D shape mesh of the objects or 3D skeleton joint positions for human. Figure 1.1 illustrates the problem on which this thesis focuses.

1.2 Challenges

The challenges in 3D object recognition can be roughly classified into three categories, namely the appearance variation, depth uncertainty and computational complexity. The appearance variation casts difficulties in localizing and identifying the objects. The depth uncertainty of each pixel is caused by the image formation process itself in which 3D objects are projected into 2D. The lost in depth dimension creates impediments in recovering the original 3D pose and shape of the object. Another challenge lies in the computational complexity as detectors should be efficient enough for real applications. Each challenge is discussed in detail in the following sections.

1.2.1 Appearance Variation

The appearance of an object can be affected by different aspects, such as inter class variation, lightening changes, pose variation etc. In terms of detection the following factors cast the most challenges.

Intra-class differences First, because of the hierarchical structure of object category taxonomy, object instances of the same category may belong to different sub-categories that can appear quite differently. For example, the car category can be further separated into sedan, hatchback, truck etc. The overall shape of a sedan is largely different from that of a truck. Second, even the same object instance can appear drastically different due to changes in lightening, covering material and texture etc. For example, a person wearing different clothings varies a lot in appearance.

Viewpoint and articulation change Another important factor of appearance variation is viewpoint from which the object is observed. The occlusion boundary of an object changes dramatically between different viewing angles. This effect is the inspiration of the aspect graph representation (Plantinga and Dyer, 1990) in the early days of computer vision. The challenge here is that it is hard to represent an object category with a single

rigid template. For non-rigid object such as human, self occlusion cause by the articulation of body pose also changes the appearances.

Background clutter and occlusion The surrounding background of an object can cause trouble in detecting an object. Just imagine playing the game of "Where's Waldo?". The presented noisy clutter background overwhelms the presence of the object and usually triggers false positive detections from the detector. An ideal detector should reliably tell the differences between foreground and background. In addition, occlusion by other objects alters the perceived silhouette shape of an object. It is already hard for me to find the red-white-strip shirt of Waldo among the enormous crowd, not to mention when he is partially blocked by someone else!

1.2.2 Depth Uncertainty

The biggest hurdle in 3D object recognition from 2D is the depth uncertainly raised naturally from the image formation process. In the classic pinhole camera model, each image coordinate (u, v) is associated with a ray $\lambda(X, Y, Z)$ in 3D passing through the camera center, where λ is a the positive unknown scaling factor inverse proportional to depth Z. Here we assume the focal length is f and the radial distortion is not considered for simplicity of discussion. The relation between 2D and 3D can be expressed as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim \lambda \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}.$$

Lifting 2D to 3D If there are multiple views of the same scene, depth can be estimated by matching 2D features in the images and triangulating the corresponding rays in 3D (Hartley and Zisserman, 2004). Recovering the lost depth dimension from a single viewpoint cannot be accomplished by triangulation. It requires carefully incorporating geometric prior to explain the observed scene or object.

Non-rigid human pose Comparing to rigid pose and shape estimation problem, recovering non-rigid pose or motion from images is more daunting. First, it is challenging to accurately localize the human joints due to the lack of distinctive texture, appearance variation, foreshortening and occlusion, etc. Second, each joint angle in 3D is ambiguous even given the perfectly localized 2D position. Either pointing inwards or outwards can result in the same projection. The lack of the rigidity constrains makes the problem much harder to solve.

1.2.3 Computational Issues

Computational complexity is another practical concern of the object recognition algorithms. Both the learning and inference algorithms should complete within a reasonable amount of computational time and space. Both the model and algorithmic complexity require careful design.

Efficiency and accuracy When dealing with 2D object detection task, a sliding window approach is usually adopted. The detector searches over the entire 2D image location and scale space, or the image pyramid. Each location is examined one after another with an object specific classifier to see if there exists an object or not. The computational time of such approach grows as the time for evaluating each classifier increases. This approach becomes more computationally intense as the classifier requires more processing time. On the other hand, better recognition accuracy is usually achieved with more sophisticated model which requires longer time to evaluate. The challenge here is to strike a balance between the efficiency and accuracy.

Search over rotations The 3D pose estimation problem encompasses a parameter search space over the 3D rotation group $\mathbf{SO}(3)$. $\mathbf{SO}(3)$ is the special orographic orthogonal in three dimensions. It has a natural structure as a manifold for which the group operations are smooth. And it turns out that performing efficient search over the group is rather hard.

One would either follow the geodesic on the manifold from certain initialization or resort to an exhaustive search approach. The former approach suffers poor initialization and the latter has difficulties in handling practical problems as the exhaustive search is too time consuming.

Search over instances In the task of 3D shape estimation, recovering an accurate shape is relatively easy when the instance is known. In that case, the shape estimation problem is reduced to a nearest neighbor problem given the database of known instance. The most trivial approach means searching over all the instances of an object category to find the matching one. Linearly searching over the models can be very inefficient and quickly becomes infeasible for practical problems. Therefore the model representation should be compact, i.e. sub-linear or even constant w.r.t. the number of instances, and still represent the entire object category. However, if the instance is unknown or not exactly covered by the model database, estimating the accurate shape is quite challenging.

1.3 Contributions

Deformable part model (DPM) (Felzenszwalb et al., 2010c), arguably one of the most successful object detectors in the past decade, represents the object as a union of parts connected in a star structure.DPM and its variants achieved the state-of-art object detection results on the most popular object recognition benchmarks such as PASCAL VOC (Everingham et al., 2010) and ImageNet (Deng et al., 2009). The key to such success lies in the combination between rigid parts and 2D deformation. The part appearances learned from data are discriminative enough to handle intra-class variation and the 2D deformation of the parts allow flexibility to deal with inter-class variation.

There have been several directions to which the researchers extended the powerful DPM models. One is to accelerate the detection process of DPM. By incorporating the idea of cascade (Dollár et al., 2012; Felzenszwalb et al., 2010a; Sapp et al., 2010; Weiss et al., 2012) or Branch-and-Bound (Kokkinos, 2011), those methods reduce the image locations

to actually evaluate the full model during inference; Another is to increase the model complexity to deal with structured output problems other than the binary classification problem as usually presented in object detection. For example, more parts are added to form a tree-structure model for human pose estimation (Yang and Ramanan, 2011) or facial landmarks localization (Zhu and Ramanan, 2012); Generic 3D object recognition is pioneered by Savarese and Fei-Fei (2007). More recently, researchers have been focused on combining part models with 3D geometry to build more powerful object detectors that are also able to provide 3D information such as viewpoint (Hu and Zhu, 2014; Liebelt et al., 2008b; Pepik et al., 2012b,c; Zia et al., 2013). Few efforts have been devoted to the combined estimation of pose and shape from a single image (Hejrati and Ramanan, 2012; Lin et al., 2014; Xiang and Savarese, 2012).

This thesis advances the start-of-art in the following two directions.

First, the proposed Active Deformable Part Models (ADPM) pushes the DPM acceleration further by considering part evaluation as a *planning problem*. An active inference process decides at each image location whether to evaluate more parts and in what order or to stop and predict a label. A policy is learned off-line to balance the trade-off between the acceleration gain and detection accuracy loss across the entire training set. The result policy leads to a three times speed-up versus the prior work of Cascade DPM (Felzenszwalb et al., 2010a).

Second, a suite of methods to estimate rich 3D information from 2D images are presented. The proposed methods can solve both instance and category, rigid and articulated 3D recognition problems from singe images. The methods achieve a wide range of successful applications including robot grasping with only a single image; single image pop-up via accurate 3D shape reconstruction of object categories; a monocular video MoCap system through 3D articulated human pose recovery. The essence of our approaches is marrying the power of discriminative learning with 3D geometrical constrains into a unified objective function that considers simultaneously appearance and geometric compatibility.

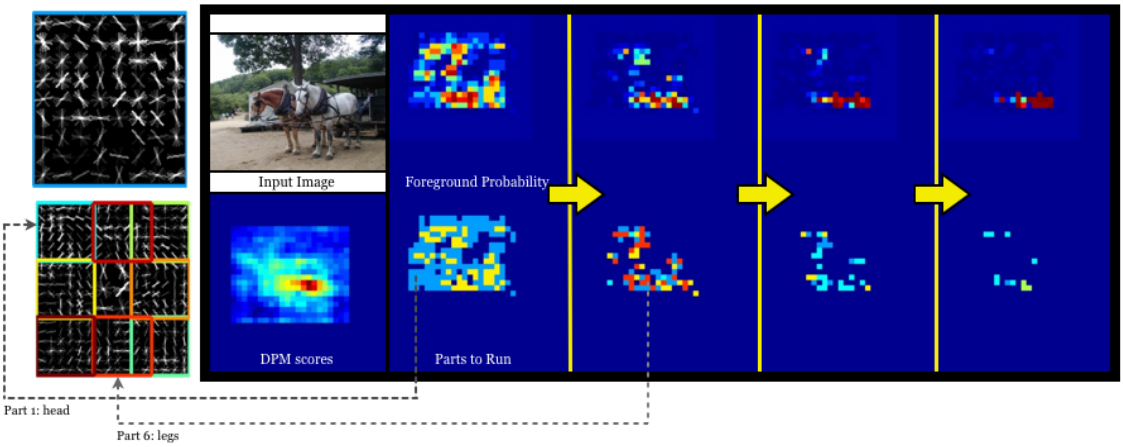


Figure 1.2: **Active DPM Overview**: A deformable part model is shown with colored root and parts in the first column. The second column contains an input image and the original DPM scores as a baseline. The rest of the columns illustrate the inference process of the Active DPM, which proceeds in rounds. The foreground probability is maintained at each image location (top row) and is updated sequentially. A policy (learned off-line) is used to select the best sequence of parts to apply at different locations. The bottom row shows the part filters applied at consecutive rounds with colors corresponding to the parts on the left. For more detailed explanation, see §5

1.3.1 Efficient 2D Object Detection, §5

At each location in the image pyramid, a part-based object detector has to make a decision: whether to evaluate more parts and in what order or to stop and predict a label. This decision can be regarded as a *planning problem*, whose state space consists of the set of previously used parts and the confidence of whether an object is present or not. While existing approaches rely on a predetermined sequence of parts, our approach optimizes the order in which to apply the part filters so that a minimal number of part evaluations provides maximal classification accuracy at each location. Our second idea is to use a decision loss in the optimization, which quantifies the trade-off between false positive and false negative mistakes, instead of the threshold-based stopping criterion utilized by most other approaches. These ideas have enabled us to propose a novel object detector,

Active Deformable Part Models (ADPM), named so because of the active part selection. The detection procedure consists of two phases: an off-line phase, which learns a part scheduling policy from the training data and an online phase (inference), which uses the policy to optimize the detection task on test images. During inference, each image location starts with equal probabilities for object and background. The probabilities are updated sequentially based on the responses of the part filters suggested by the policy. At any time, depending on the probabilities, the policy might terminate predicting either a background label (which is what most cascaded methods take advantage of) or a positive label, in which case all unused part filters are evaluated in order to obtain the complete DPM score. Fig. 1.2 exemplifies the inference process. The main contributions can be summarized as the following:

- We obtain an active part selection policy which optimizes the order of the filter evaluations and balances number of evaluations used with the classification accuracy based on the scores obtained during inference.
- The proposed detector achieves a significant speed-up versus the cascade DPM without sacrificing accuracy.
- The approach can be generalized to any detection problem, which involves a linear additive score and uses several parts (stages) even if they are just SIFT points.

1.3.2 3D Pose Estimation of Object Instances, §6

The goal is to detect and localize objects in single view RGB images of environments containing arbitrary ambient illumination and substantial clutter for the purpose of autonomous grasping. Objects can be of arbitrary color and interior texture and, thus, we assume knowledge of only their 3D model without any appearance or texture information. Using 3D models makes an object detector immune to intra-class texture variations. We further abstract the 3D model by only using its 2D silhouette and thus detection is driven by the shape of the 3D object's projected occluding boundary.

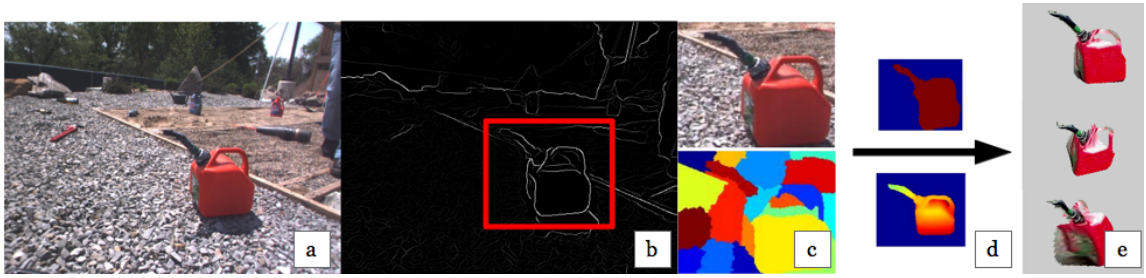


Figure 1.3: Overview of the proposed 3D pose estimation of object instances approach. From left-to-right: a) The input image. b) S-DPM detection hypothesis on the image contours. c) The hypothesis bounding box (red) is segmented into superpixels. d) The set of superpixels with the closest chordiogram distance to the model silhouette is selected. Pose is iteratively refined such that the model projection aligns well with the foreground mask silhouette. e) Three textured synthetic views of the final pose estimate are shown.

Object silhouettes with corresponding viewpoints that are tightly clustered on the viewsphere are used as positive exemplars to train the state-of-the-art Deformable Parts Model (DPM) (Felzenszwalb et al., 2010c). We term this shape-aware version S-DPM. S-DPM simultaneously detects the object and coarsely estimates the object's pose. We propose to apply an S-DPM classifier on image contours as a first high recall step yielding several bounding box hypotheses. Given these hypotheses, we solve for segmentation and localization simultaneously. After over-segmenting the hypothesis region into superpixels, the superpixels that best match a model boundary are selected using a shape-based descriptor, the chordiogram (Toshev et al., 2012). A chordiogram-based matching distance is used to compute the foreground segment and rerank the hypotheses. Finally, using the full 3D model we estimate all 6-DOF of the object by efficiently iterating on the pose and computing matches using dynamic programming.

We also demonstrate our approach with a (PR2) robot grasping 3D objects on a cluttered table based on a single view RGB image. We report 3D pose accuracy by comparing the estimated pose rendered by the proposed approach with a ground truth point cloud recovered with a RGB-D sensor. Such grasping capability with accurate pose is crucial for

robot operation, where popular RGB-D sensors cannot be used (e.g., outdoors) and stereo sensors are challenged by the uniformity of the object's appearance within their boundary. Fig. 1.3 exemplifies the inference process. The main contributions can be summarized as the following.

- In terms of assumptions, our approach is among the few in the literature that can detect 3D objects in single images of cluttered scenes independent of their appearance.
- It combines the high recall of an existing discriminative classifier with the high precision of a holistic shape descriptor achieving a simultaneous segmentation and detection reranking.
- Due to the segmentation, it selects the correct image contours to use for 3D pose refinement, a task that was previously only possible with stereo or depth sensors.

1.3.3 3D Pose and Shape Estimation of Object Categories, §7

We propose a novel approach that marries the power of discriminative parts with an explicit 3D geometric representation with the goal to infer 3D shape and continuous pose of an object (or *pop-up*) from a single image. Part descriptors are discriminatively learned in training images. Such parts are centered around projections of 3D landmarks which are given in abundance on the training 3D models. To establish a compact representation we minimize the number of needed landmarks by solving a facility-location problem.

To deal with geometric deformation, we summarize the training set of 3D models into a shape dictionary from which we can generalize by linear combination. Given a test image we detect top location hypotheses of each part. The challenge is how to fit best these parts by maximizing the geometric consistency. This entails the selection among the hypotheses of each part and the shape/pose computation. Unlike other approaches which rely on local optimization and initialize pose by DPM-based discretized pose estimation (Lin et al., 2014; Zia et al., 2013), we compute the selection as well as the shape and

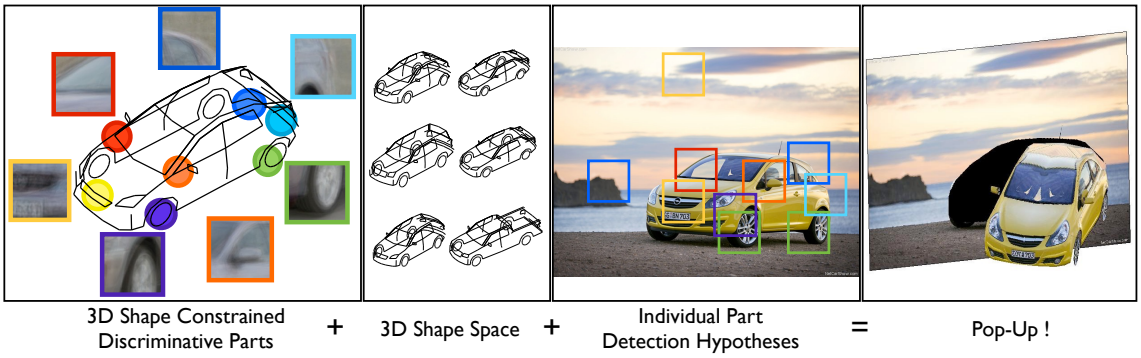


Figure 1.4: Illustrative summary of the single image popup approach: 3D Landmarks on a 3D model are associated with discriminatively learned part descriptors (left). Intra-class shape variation is captured with linear combinations of a sparse shape basis (2nd left). Learned part descriptors produce multiple maximum responses for each part in a testing image (3rd from left). The selection of the part hypotheses, 3D pose and 3D shape are simultaneously estimated and the result is illustrated through a popup (right).

pose parameters in one step using a convex program solved with the alternating direction method of multipliers (ADMM). Fig. 1.4 exemplifies the inference process. The main contributions can be summarized as the following.

- A convex optimization framework for joint landmark localization, fine grained 3D shape and continuous pose estimation from a single image.
- Our convex objective does not require viewpoint or detection initialization.
- An automatic landmark selection method considering both discriminative power in appearance and spatial coverage in geometry.

1.3.4 Articulated 3D Pose Estimation from Image Sequences, §8

This chapter is concerned with the challenge of recovering the 3D full-body human pose from a monocular RGB image sequence. Potential applications of the presented research

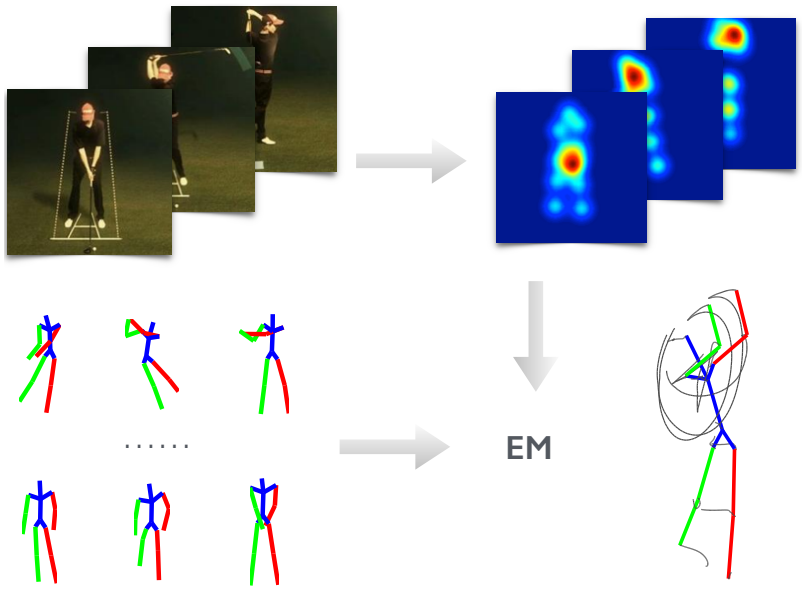


Figure 1.5: Overview of the proposed articulated 3D Pose Estimation approach. (top-left) Input image sequence, (top-right) CNN-based heat map outputs representing the soft localization of 2D joints, (bottom-left) 3D pose dictionary, and (bottom-right) the recovered 3D pose sequence reconstruction.

include human-computer interaction (cf. (Shotton et al., 2011)), surveillance, video browsing and indexing, and virtual reality.

A 3D pose recovery framework that consists of a novel synthesis between discriminative image-based and 3D reconstruction approaches is presented. In particular, the approach reasons jointly about image-based 2D part location estimates and model-based 3D pose reconstruction, so that they can benefit from each other. Further, to improve the approach's robustness against detector error, occlusion, and reconstruction ambiguity, temporal smoothness is imposed on the 3D pose and viewpoint parameters.

In the light of previous research, this chapter makes the following contributions. Given a monocular video, two novel approaches for recovering the 3D human pose sequence are presented. The first approach assumes that the 2D poses are provided and proceeds by combining a sparse representation of 3D pose with temporal smoothness in the 3D domain to estimate the 3D poses. The second approach generalizes the first approach by

relaxing the common but restrictive assumption that the 2D poses are provided or explicitly estimated (cf. (Wang et al., 2014)) and instead treats the 2D pose as a latent variable. A CNN-based body joint detector is used to learn the uncertainty map for the image location of each joint. To estimate the 3D pose, an efficient EM algorithm is proposed, where the latent joint positions are marginalized to fully account for the uncertainty in the 2D joint locations. Finally, empirical evaluation demonstrates that the proposed approaches are more accurate compared to extant approaches. In particular, in the case where 2D joint locations are provided, the proposed approach exceeds the accuracy of the state-of-the-art NRSFM baseline (Dai et al., 2012) on the Human3.6M dataset (Ionescu et al., 2014). In the case where the 2D landmarks are unknown, empirical results on the Human3.6M dataset demonstrate overall improvement over published results. Further, the proposed approach is shown to outperform a publicly available 2D pose estimation baseline on the challenging PennAction dataset (Zhang et al., 2013). Fig. 1.5 exemplifies the inference process.

1.3.5 Published Work Supporting This Thesis

Active Deformable Part Models (§5) is presented in (Zhu et al., 2014a) for fast object detection. Single Image 3D Object Pose Estimation (§6) first appeared in (Zhu et al., 2014b) that enables robots grasping different objects by only taking a single image of the scene. The method of 3D pop-up from single images (§7) extended the scope of single image grasping to deal with object categories and recover surface shapes appeared in (Zhu et al., 2015). A further extension to articulated 3D human pose estimation from monocular videos (§8) is recently accepted to CVPR 2016. In addition, the Active Deformable Part Models contribute to the success in our recent work of semantic localization (Atanasov et al., 2015).

Chapter 2

Related Work

2.1 Accelerated Object Detection

We will refer to work on object detection that optimizes the inference stage rather than the representations since our representation is identical with DPM (Felzenszwalb et al., 2010b). Our method is inspired by an acceleration of the DPM object detector, the cascade DPM (Felzenszwalb et al., 2010a). However, while the sequence of parts evaluated in the cascade DPM is pre-defined and a set of thresholds has to be determined empirically, our approach selects the part order and the stopping time at each location based on an optimization criterion. We find the next closest approach to be (Sznitman et al., 2013), which maintains a foreground probability at each stage of a multi-stage ensemble classifier and determines a stopping time based on the corresponding entropy. The difference of our approach is that it jointly optimizes the stage order and the stopping criterion. Kokkinos (Kokkinos, 2011) uses Branch-and-Bound (BB) to prioritize the search over image locations driven by an upper bound on the classification score. It is related to our approach in that object-less locations are easily detected and the search is guided in location space but with the difference that our policy proposes the next part to be tested in cases that no label can yet be given to a particular location. Earlier approaches (Lampert, 2010; Lampert et al., 2008; Lehmann et al., 2011b) relied on BB to constrain the search space of object

detectors based on a sliding window or a Hough transform but without deformable parts. Another related group of approaches focuses on learning a sequence of object template tests in position, scale, and orientation space that minimizes the total computation time through a coarse-to-fine evaluation (Fleuret and Geman, 2001; Pedersoli et al., 2011).

The classic work by Viola and Jones (Viola and Jones, 2001) introduced a cascade of classifiers whose order was determined by importance weights learned by AdaBoost. The approach was studied extensively in (Bourdev and Brandt, 2005; Brubaker et al., 2008; Gualdi et al., 2012; Lehmann et al., 2011a; Zhang et al., 2011). Recently, Dollar et al. (Dollár et al., 2012) introduced cross-talk cascades which allow detector responses to trigger or suppress the evaluation of weak classifiers in their neighborhood by exploiting the correlation of the classifier responses in the neighboring positions and scales. Weiss et al. (Weiss et al., 2012) used structured prediction cascades to optimize a function with two objectives: pose refinement and minimum filter evaluation cost. Sapp et al. (Sapp et al., 2010) learn a cascade of pictorial structures with increasing pose resolution by progressively filtering the pose state space. Its emphasis is on pre-filtering structures through max-margin scoring rather than part locations so that human poses with weak individual part appearances can still be recovered. Rahtu et al. (Rahtu et al., 2011) use general "objectness" filters to produce location proposals which are fed into a cascade, designed to maximize the quality of the locations that advance to the next stage. Our approach is also related to and could be combined with active learning using Gaussian processes for classification (Kapoor et al., 2010).

Similarly to the closest approaches above (Felzenszwalb et al., 2010a; Kokkinos, 2011; Sznitman et al., 2013), our method aims to balance the number of part filter evaluations with the classification accuracy in part-based object detection. The novelty and the main advantage of our approach is that in addition it optimizes the part filter ordering. Since our "cascades" still run only on parts, we do not expect the approach to show higher accuracy than structured prediction cascades (Sapp et al., 2010) which consider more sophisticated representations that the pictorial structures in the DPM.

2.2 Rigid Object Pose Estimation

Early approaches based on using explicit 3D models are summarized in Grimson's book (Grimson, 1990) and focus on efficient techniques for voting in pose space. Horaud (Horaud, 1987) investigated object recognition under perspective projection using a constructive algorithm for objects that contain straight contours and planar faces. Hausler (Häusler and Ritter, 1999) derived an analytical method for alignment under perspective projection using the Hough transform and global geometric constraints. Aspect graphs in their strict mathematical definition (each node sees the same set of singularities) were not considered practical enough for recognition tasks but the notion of sampling of the view-space for the purpose of recognition was introduced again in (Cyr and Kimia, 2001) which were applied in single images with no background. A Bayesian method for 3D reconstruction from a single image was proposed based on the contours of objects with sharp surface intersections (Han and Zhu, 2003). Sethi et al. (Sethi et al., 2004) compute global invariant signatures for each object from its silhouette under weak perspective projection. This approach was later extended (Lazebnik et al., 2002) to perspective projection by sampling a large set of epipoles for each image to account for a range of potential viewpoints. Liebelt et al. work with a view space of rendered models in (Liebelt et al., 2008a) and a generative geometry representation is developed in (Liebelt and Schmid, 2010). Villamizar et al. (Villamizar et al., 2011) use a shared feature database that creates pose hypotheses verified by a Random Fern pose specific classifier. In (Glasner et al., 2011a), a 3D point cloud model is extracted from multiple view exemplars for clustering pose specific appearance features. Others extend deformable part models to combine viewpoint estimates and 3D parts consistent across viewpoints, e.g., (Pepik et al., 2012a). In (Hao et al., 2013), a novel combination of local and global geometric cues was used to filter 2D image to 3D model correspondences.

Others have pursued approaches that not only segment the object and estimate the 3D pose but also adjusts the 3D shape of the object model. For instance, Gaussian Process Latent Variable Models were used for the dimensionality reduction of the manifold of

shapes and a two-step iteration optimizes over shape and pose, respectively (Prisacariu et al., 2013). The drawback of these approaches is that in the case of scene clutter they do not consider the selection of image contours. Further, in some cases tracking is used for finding the correct shape. This limits applicability to the analysis of image sequences, rather than a single image, as is the focus in the current paper.

Our approach resembles early proposals that avoid appearance cues and uses only the silhouette boundary, e.g., (Cyr and Kimia, 2001). None of the above or the exemplar-based approaches surveyed below address the amount of clutter considered here and in most cases the object of interest occupies a significant portion of the field of view.

Early view exemplar-based approaches typically assume an orthographic projection model that simplifies the analysis. Ullman (Ullman and Basri, 1991) represented a 3D object by a linear combination of a small number of images enabling an alignment of the unknown object with a model by computing the coefficients of the linear combination, and, thus, reducing the problem to 2D. In (Basri, 1993), this approach was generalized to objects bounded by smooth surfaces, under orthographic projection, based on the estimation of curvature from three or five images. Much of the multiview object detector work based on discrete 2D views (e.g., (Gu and Ren, 2010a)) has been founded on successful approaches to single view object detection, e.g., (Felzenszwalb et al., 2010c). Savarese and Fei-Fei Savarese and Fei-Fei (2007) presented an approach for object categorization that combines appearance-based descriptors including the canonical view for each part, and transformations between parts. This approach reasons about 3D surfaces based on image appearance features. In (Payet and Todorovic, 2011), detection is achieved simultaneously with contour and pose selection using convex relaxation. Hsiao et al. (Hsiao et al., 2010) also use exemplars for feature correspondences and show that ambiguity should be resolved during hypothesis testing and not at the matching phase. A drawback of these approaches is their reliance on discriminative texture-based features that are hardly present for the types of textureless objects considered in the current paper.

As far as RGB-D training and test examples are concerned, the most general and representative approach is (Lai et al., 2011). Here, an object-pose tree structure was proposed that simultaneously detects and selects the correct object category and instance, and refines the pose. In (Rusu et al., 2010), a viewpoint feature histogram is proposed for detection and pose estimation. Several similar representations are now available in the Point Cloud Library (PCL) (Rusu and Cousins, 2011). We will not delve here into approaches that extract the target objects during scene parsing in RGB-D images but refer the reader to (Koppula et al., 2011) and the citations therein.

The 2D-shape descriptor, chordiogram (Toshev et al., 2012), we use belongs to approaches based on the optimal assembly of image regions. Given an over-segmented image (i.e., superpixels), these approaches determine a subset of spatially contiguous regions whose collective shape (Toshev et al., 2012) or appearance (Vijayanarasimhan and Grauman, 2011) features optimize a particular similarity measure with respect to a given object model. An appealing property of region-based methods is that they specify the image domain where the object-related features are computed and thus avoid contaminating objected-related measurements from background clutter.

2.3 3D Shape Reconstruction

The most related work includes the family of methods that estimate an object shape by aligning a deformable shape model to image features. This idea originated from the active shape model (ASM) (Cootes et al., 1995a), which was originally proposed for segmentation and tracking based on low-level image features. Cristinacce and Cootes (Cristinacce and Cootes, 2006) proposed the constrained local models (CLM), which combined ASM with local appearance models for 2D feature localization in face images. Gu and Kanade (Gu and Kanade, 2006) presented a method to align 3D deformable models to 2D images for 3D face alignment. The similar methods were also proposed for 3D car modeling

(Hejrati and Ramanan, 2012; Hu and Zhu, 2014; Lin et al., 2014; Zia et al., 2013) and human pose estimation (Ramakrishna et al., 2012; Zhou and De la Torre, 2014). Our method differs in that we use a data-driven approach for discriminative landmark selection and we solve landmark localization and shape reconstruction in a single convex framework, which enables a global solution.

The representation of our model is inspired by recent advances in part-based modeling (Felzenszwalb et al., 2010b; Hariharan et al., 2012; Kokkinos, 2011; Singh et al., 2012), which models the appearance of object classes with a collection of mid-sized discriminative parts. Our optimization approach is related to the previous work on using convex relaxation techniques for object matching (Jiang et al., 2011; Li et al., 2011; Maciel and Costeira, 2003). These methods focused on finding the point-to-point correspondence between an object template and an image in 2D, while our method considers 3D to 2D matching as well as shape variability.

Our paper is also related to recent work on 3D pose estimation which encodes the geometric relations among local parts and achieved continuous pose estimation. Several work leveraged 3D models to warp features or parts into their canonical view (Savarese and Li, 2007; Xiang and Savarese, 2012; Yan et al., 2007). Other work rendered local appearances and depth from 3D models and subsequently encoded in a 3D voting scheme (Glasner et al., 2011b; Liebelt et al., 2008b; Sun et al., 2010). DPM was further lifted to 3D deformable models (Fidler et al., 2012; Pepik et al., 2012b) to predict continuous viewpoint. Instance models were also used to recover 3D pose of an object (Aubry et al., 2014; Lim et al., 2013). But this line of work focused on pose estimation and either used generic class models or instance-based models. Our approach differs in that we not only provide a detailed shape representation but also consider intra-class variability.

2.4 Articulated 3D Pose Estimation

Considerable research has addressed the challenge of 3D human motion capture from video (Brubaker et al., 2010; Moeslund et al., 2006; Sminchisescu, 2007). Early research on 3D monocular pose estimation in videos largely centred on incremental frame-to-frame pose tracking, e.g., (Bregler and Malik, 1998; Sigal et al., 2012; Sminchisescu and Triggs, 2003). These approaches rely on a given pose and dynamic model to constrain the pose search space. Notable drawbacks of this approach include: the requirement that the initialization be provided and their inability to recover from tracking failures. To address these limitations, more recent approaches have cast the tracking problem as one of data association across frames, i.e., "tracking-by-detection", e.g., (Andriluka et al., 2010). Here, candidate poses are first detected in each frame and subsequently a linking process attempts to establish temporally consistent poses.

Another strand of research has focused on methods that predict 3D poses by searching a database of exemplars (Jiang, 2010; Mori and Malik, 2006; Shakhnarovich et al., 2003) or via a discriminatively learned mapping from the image directly or image features to human joint locations (Agarwal and Triggs, 2006; Ionescu et al., 2014; Salzmann and Urtasun, 2010; Tekin et al., 2015; Yu et al., 2013). Recently, deep convolutional networks (CNNs) have emerged as a common element behind many state-of-the-art approaches, including human pose estimation, e.g., Li and Chan (2014); Li et al. (2015); Tompson et al. (2014); Toshev and Szegedy (2014). Here, two general approaches can be distinguished. The first approach casts the pose estimation task as a joint location regression problem from the input image (Li and Chan, 2014; Li et al., 2015; Toshev and Szegedy, 2014). The second approach uses a CNN architecture for body part detection (Chen and Yuille, 2014; Jain et al., 2014; Pfister et al., 2015; Tompson et al., 2014) and then typically enforces the 2D spatial relationship between body parts as a subsequent processing step. Similar to the latter approaches, the proposed approach uses a CNN-based architecture to regress confidence heat maps of 2D joint position predictions.

Most closely related to the present paper are generic factorization approaches for

recovering 3D non-rigid shapes from image sequences captured with a single camera (Akhter et al., 2011; Bregler et al., 2000; Cho et al., 2015; Dai et al., 2012; Zhu et al., 2014c), i.e., non-rigid structure from motion (NRSFM), and human pose recovery models based on known skeletons (Lee and Chen, 1985; Park and Sheikh, 2011; Taylor, 2000; Valmadre and Lucey, 2010) or sparse representations (Akhter and Black, 2015; Fan et al., 2014; Ramakrishna et al., 2012; Zhou et al., 2015b,c). Much of this work has been realized by assuming manually labeled 2D joint locations; however, there is some recent work that has used a 2D pose detector to automatically provide the input joints (Wang et al., 2014) or solves the correspondence problem by matching a spatio-temporal pose model to candidate trajectories extracted from a video (Zhou and la Torre, 2014).

Part II

Preliminaries

Chapter 3

Discriminative Learning

In this chapter, we briefly cover the basics of discriminative learning and introduce two successful methods on which this thesis is based.

In the machine learning literature, classifiers generally fall into two categories: generative classifiers and discriminative classifiers. Generative classifiers learn the joint distribution $p(y, x)$ of the input x and the label y. Combined with the prior distribution $p(x)$ of the label, the posterior distribution $p(y|x)$ is then derived using Bayes rules. Discriminative classifiers learn the posterior distribution $p(y|x)$ directly by modeling the distribution with a parametric model and optimizes the parameters using a training set. In the seminal work of (Ng and Jordan, 2002), the two types of classifiers are compared and the results suggest that discriminative models have lower asymptotic error given large training data.

In recent years, discriminative models have taken large strides in computer vision as more large-scale datasets are becoming available annotated with class labels, object bounding boxes and even detailed segmentation masks. Several discriminative models quickly become very successful with applications to object recognition. Among them are Support Vector Machine (Cortes and Vapnik, 1995; Vapnik and Kotz, 1982) and Deep Neural Networks (Krizhevsky et al., 2012; LeCun et al., 1998) which underpin the latest advances in the field.

3.1 Support Vector Machine

One particularly successful discriminative model is Support Vector Machine (SVM), with many applications to document classification, object classification etc. SVM was originally started when statistical learning theory was developed further by Vapnik (Vapnik and Kotz, 1982) and later extended closer to its current form (Cortes and Vapnik, 1995). For simplicity, we only discuss the linear SVM. The non-linear SVM extends the linear case and builds on the idea of kernel methods, which is out of the scope of this thesis.

The linear SVM learns a separating hyperplane $w^T x + b = 0$ from labeled examples $D = \{\langle x_1, y_1\rangle, ..., \langle x_n, y_n\rangle\}$, where $y_i \in \{-1, 1\}$. In order to achieve robustness to noise and gain better generalization to unseen data, SVM maximizes the margin of the separating hyperplane to the examples. Formally, linear SVM can be formulated as the following optimization problem,

$$\min_{w,b,\xi \geq 0} \frac{1}{2} w^T w + C \sum_i \xi_i \tag{3.1}$$

$$\text{s.t. } y_i(w^T x_i + b) \geq 1 - \xi_i, \ i = 1, ..., n, \tag{3.2}$$

where ξ_i is called the slack variable, introduced for penalizing the incorrectly classified examples, C is the weight on the penalty cost.

By taking the gradient w.r.t. w and b on the corresponding Lagrangian, we can derive the equivalent dual problem as,

$$\max_{\alpha \geq 0} \sum_i \alpha_i - \frac{1}{2} \alpha_i \alpha_j y_i y_j x_i^T x_i \tag{3.3}$$

$$\text{s.t. } \sum_i \alpha_i y_i = 0, \alpha_i \leq C, \ i = 1, ..., n, \tag{3.4}$$

where α_i is the Lagrange multiplier for the inequality constrain 3.2 in the primal form. The resulting dual form of SVM is a quadratic optimization problem easier than the primal problem. The optimal solution can be derived from the KKT condition (Boyd and Vandenberghe, 2004). Both specialized solvers, e.g. SMO (Platt et al., 1998), and generic primal-dual optimization algorithms, e.g. ADMM (Boyd et al., 2011), have been designed

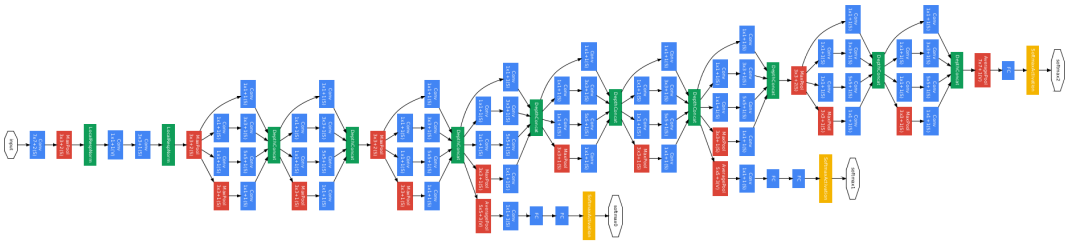


Figure 3.1: GoogleNet desgined by (Szegedy et al., 2015): an example of modern deep convolutional neural network with many layers of convolutions, where the blue boxes are the convolution layers.

to tackle the problem, and we refer curious readers to an excellent tutorial by Burges (1998) for more details.

3.2 Deep Neural Networks

The development of neural networks dates back to 1957 when Frank Rosenblatt invented a linear classifier called the perceptron (Rosenblatt, 1961). One of the early success of neural networks was when convolutional neural networks (LeCun et al., 1989) was invented and successfully applied to handwritten digit recognition in the 90s. After the initial hype the related research went mostly under the radar of the computer vision community partially due to the success of SVMs. Until recently, with the advances of computing hardware especially Graphics Processing Units (GPU), convolutional neural netwroks staged a huge comeback after winning the 2012 ImageNet Challenge (Krizhevsky et al., 2012). Interestingly, despite the differences of modern deep learning models with many more layers 3.1 than their predecessors, the learning algorithm of the neural networks has remain the same, namely back-propagation Rumelhart et al. (1988).

Most neural networks are organized into groups of units called layers. Most neural network architectures arrange these layers in a chain structure, with each layer being a

function of the layer that precedes it. In this structure, the each layer $i \in \{1, ..., n\}$ is given by

$$h^{(i)} = g^{(i)}(W^{(i)T} h^{(i-1)} + b^{(i)}) \tag{3.5}$$

where $h^{(i)}$ is the activation of each layer and $h^{(0)} = x$ is the input data. Note in the case of convolutional neural network, convolutions can be rearranged as matrix multiplications. $g^{(i)}$ is a nonlinear function applied after the linear transform of the activations from the previous layer. The choice of nonlinear activation functions can be Rectified Linear Units (ReLU), logistic, max pooling etc.

The universal approximation theorem (Hornik et al., 1989) states that under mild assumptions on the activation mapping, neural networks with a single hidden layers can approximate continuous functions on compact sets of $\mathcal{R}^n$. Modern multi-layered deep neural networks have much bigger capacity but the nonlinearity still poses a big challenge to learning such complex model.

In general, the cost function of a network is a non-convex function w.r.t. to input, thus learning the parameters of a neural network is usually resorted to gradient decent. The minimization of the cost function is generally carried out via error back-propagation (Rumelhart et al., 1988), or in other words, propagating the gradients using the chain rule. More specifically, the following derivation

$$\frac{\partial h^{(n)}}{\partial h^{(i)}} = \frac{\partial h^{(n)}}{\partial h^{(n-1)}} \frac{\partial h^{(n-1)}}{\partial h^{(i)}} \tag{3.6}$$

can be recursively expanded to compute the gradient the cost function w.r.t. the input layer. The gradient of layer n w.r.t. the activations of layer i can be expressed as the products of gradients between the intermediate layers. The challenges in learning large network is that the cost function landscape is generally highly non-linear and exists many local minima and saddle points.

3.3 Stochastic Gradient Descent

The robustness of discriminative classifiers generally improves as more training examples are exposed. However, as the datasets grow larger in scale, parameter learning with gradient descent via batch optimization usually cannot be directly applied due to machine memory limitations and computational efficiency reasons. In such cases, stochastic gradient descent is deployed iteratively to approximate the gradient that would be computed from the whole dataset.

In general, the learning algorithm minimizes the sum of empirical losses over a dataset, more specifically an objective function of the form,

$$J(\theta) = \sum_{i=1}^{n} J_i(\theta), \tag{3.7}$$

where θ is the parameter to be estimated. A standard batch gradient descent method would update θ as the following,

$$\theta = \theta - \eta \sum_{i=1}^{n} \nabla J_i(\theta) \tag{3.8}$$

With stochastic gradient descent, the batch gradient is approximated with the gradient at a single example,

$$\theta = \theta - \eta \nabla J_i(\theta). \tag{3.9}$$

In practice, the gradient is computed on a mini-batch of examples. In additional many methods study the effect of mini-batch updates and provide solution to regularize the gradient for better convergence. In the literature of deep learning, for example, momentum (Krizhevsky et al., 2012), RMSProp (Tieleman and Hinton., 2012) and AdaGrad (Duchi et al., 2011) etc, are invented to accelerate the learning of neural networks. Bootstrapping or hard-negative mining (Felzenszwalb et al., 2010b) etc. are introduced for training SVM faster on the large-scale dataset.

Chapter 4

Convex Optimization

In this chapter, we discuss a class of optimization methods that are most related to this thesis called Proximal Algorithms. This class of optimization algorithms are designed for challenges raise in constrained convex optimization problems with non-smooth objective function. First, we discuss the general concepts and underlying theory of Proximal Algorithms. Then we focus our discussion on a powerful subclass: Alternating Direction Method of Multipliers.

4.1 Proximal Algorithms

Newton's method provides a standard tool for solving unconstrained smooth minimization problems of modest size. In this section, we briefly discuss a class of algorithms called *proximal algorithms* (Parikh and Boyd, 2013). Proximal algorithms are analogous to Newton's method but for solving non-smooth, constrained, large-scale, or distributed versions of these problems.

4.1.1 Proximal Operator

Definition 1. *The proximal operator* $prox_f : \mathbf{R}^n \to \mathbf{R}^n$ *of* f *is defined by*

$$prox_f(v) = \arg\min_x \left(f(x) + (1/2)\|x - v\|_2^2 \right). \tag{4.1}$$

The function minimized on the right-hand side is strongly convex w.r.t. v *and not everywhere infinite, so it has a unique minimizer for every* $v \in \mathbf{R}^n$.

The proximal operator of the scaled function λf, where $\lambda > 0$, is expressed as

$$\text{prox}_{\lambda f}(v) = \arg\min_x \left(f(x) + (1/2\lambda)\|x - v\|_2^2 \right). \tag{4.2}$$

Proposition 1. *The point* x^* *minimizes* f *if and only if*

$$x^* = prox_f(x^*),$$

i.e., if x^* *is a fixed point of* $prox_f$.

This fundamental property gives a link between proximal operators and fixed point theory; *e.g.*, many proximal algorithms for optimization can be interpreted as methods for finding fixed points of appropriate operators. For the proof of the proposition we refer to (Parikh and Boyd, 2013).

4.1.2 Moreau Decomposition

The convex conjugate (Boyd and Vandenberghe, 2004) of a function f is defined as,

$$f^*(y) = \sup_x \left(y^T x - f(x) \right). \tag{4.3}$$

The following equality always holds:

$$v = \text{prox}_f(v) + \text{prox}_{f^*}(v) \tag{4.4}$$

This property, known as *Moreau decomposition* gives a simple way to obtain the proximal operator of a function f in terms of the proximal operator of f^*. For example, if $f = \|\cdot\|$ is a general norm, then $f^* = I_B$, where

$$B = \{x \mid \|x\|_* \leq 1\}$$

is the unit ball of the dual norm $\|\cdot\|_*$, defined by

$$\|z\|_* = \sup\{z^T x \mid \|x\| \leq 1\}.$$

By Morean decomposition, this implies that

$$v = \text{prox}_f(v) + \Pi_B(v). \tag{4.5}$$

We can see that, evaluating $\text{prox}_f(v)$ is easy if we know how to project one to B.

4.1.3 Proximal Operator of the Spectral Norm

Part of this thesis regarding 3D reconstruction from single image builds on the following proposition (Zhou et al., 2015c) which states the proximal operator of the spectral norm.

Proposition 2. *The solution to the following problem*

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{Y} - \mathbf{X}\|_F^2 + \lambda\|\mathbf{X}\|_2 \tag{4.6}$$

is given by $\mathbf{X}^* = \mathcal{D}_\lambda(\mathbf{Y})$*, where*

$$\mathcal{D}_\lambda(\mathbf{Y}) = \mathbf{U}_\mathbf{Y}\, diag\,[\sigma_\mathbf{Y} - \lambda\mathcal{P}_{l_1}(\sigma_\mathbf{Y}/\lambda)]\,\mathbf{V}_\mathbf{Y}^T, \tag{4.7}$$

$\mathbf{U}_\mathbf{Y}$*,* $\mathbf{V}_\mathbf{Y}$ *and* $\sigma_\mathbf{Y}$ *denote the left singluar vectors, right singular vectors and singular values of* $\mathbf{Y}$*, respectively.* $\mathcal{P}_{l_1}$ *is the projection of a vector to the unit* l_1*-norm ball.*

Proof The problem in (4.6) is a proximal problem. The associated proximal operator is the solution to the following minimization problem

$$\text{prox}_{\lambda F}(\mathbf{Y}) = \arg\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{Y} - \mathbf{X}\|_F^2 + \lambda F(\mathbf{X}). \tag{4.8}$$

In this case, $F(\mathbf{X}) = \|\mathbf{X}\|_2 = \|\sigma_\mathbf{X}\|_\infty$, where $\|\cdot\|_\infty$ denotes the l_∞-norm. Based on the property of spectral functions (Parikh and Boyd, 2013), we have

$$\text{prox}_{\lambda F}(\sigma_\mathbf{Y}) = \mathbf{U}_\mathbf{Y}\, diag\,[\text{prox}_{\lambda f}(\sigma_\mathbf{Y})]\,\mathbf{V}_\mathbf{Y}^T, \tag{4.9}$$

where f is the l_∞-norm. The poximal operator of the l_∞-norm can be computed by Meoreau decomposition §4.1.2:

$$\text{prox}_{\lambda f}(\sigma_{\mathbf{Y}}) = \sigma_{\mathbf{Y}} - \lambda \mathcal{P}_{l_1}(\sigma_{\mathbf{Y}}/\lambda), \tag{4.10}$$

given that the l_1-norm is the dual norm of the l_∞-norm. □

4.1.4 Proximal Gradient Descent

Consider a general problem of the form

$$\min_x \; f(x) + g(x), \tag{4.11}$$

where $f : \mathbf{R}^n \to \mathbf{R}$ and $g : \mathbf{R}^n \to \mathbf{R} \cup \{+\infty\}$ are closed proper convex and f is differentiable. The optimality condition is satisfied when x^* minimizes 4.11 such that the following equivalent statements hold for any fixed scalar $\lambda > 0$

$$\begin{aligned} 0 &\in \lambda \nabla f(x^*) + \lambda \partial g(x^*), \\ 0 &\in \lambda \nabla f(x^*) - x^* + x^* + \lambda \partial g(x^*), \\ (I + \lambda \partial g)(x^*) &\in (I - \lambda \nabla f)(x^*), \\ x^* &= (I + \lambda \partial g)^{-1}(I - \lambda \nabla f)(x^*). \end{aligned} \tag{4.12}$$

Equation 4.12 leads to an fixed point iterative update scheme:

$$x_k = (I + \lambda \partial g)^{-1}(I - \lambda \nabla f)(x_{k-1}), \; (\lambda > 0) \tag{4.13}$$

which is equivalent to

$$x_k = \text{prox}_{\lambda g}(x_{k-1} - \lambda \nabla f(x_{k-1})) \tag{4.14}$$

4.2 Alternating Direction Method of Multipliers

Alternating Direction Method of Multipliers (ADMM) (Boyd et al., 2011) is an important class of proximal algorithms. ADMM can be viewed as a primal-dual method, which

solves for the saddle point of the augmented Lagrangian associated with the constrained optimization problem. ADMM has a rather long history in optimzation but recently recognized as a powerful tool to solve many non-smooth, constrained and large scale optimization problem. ADMM is also known as Split Bregman method (Goldstein and Osher, 2009).

ADMM solves problems in the form

$$\begin{aligned} \min_{x,z} \ & f(x) + g(z) \\ \text{s.t. } & Ax + Bz = c. \end{aligned} \tag{4.15}$$

with variables $x \in \mathbf{R}^n$ and $z \in \mathbf{R}^m$, where $A \in \mathbf{R}^{p\times n}$, $B \in \mathbf{R}^{p\times m}$, and $c \in \mathbf{R}^p$. Both f and g are assumed to be convex.

4.2.1 Augmented Lagrangian

Consider the following problem

$$\begin{aligned} \min_{x} \ & f(x) \\ \text{s.t. } & Ax = b, \end{aligned} \tag{4.16}$$

with variable $x \in \mathbf{R}^n$, where $A \in \mathbf{R}^{m\times n}$ and $f : \mathbf{R}^n \to \mathbf{R}$ is convex. The Lagrangian of problem (4.16) is

$$L(x, y) = f(x) + y^T(Ax - b).$$

Now let us look at an equivalent problem

$$\begin{aligned} \min_{x} \ & f(x) + \frac{\rho}{2}\|Ax - b\|_2^2 \\ \text{s.t. } & Ax = b. \end{aligned} \tag{4.17}$$

The equivalence is clear since for any feasible x, the added $(\rho/2)\|Ax-b\|_2^2$ term is always zero. The associated Lagrangian

$$L_\rho(x, y) = f(x) + y^T(Ax - b) + \frac{\rho}{2}\|Ax - b\|_2^2. \tag{4.18}$$

is called the *augmented Lagrangian* of the problem (4.16).

4.2.2 Alternating Direction Minimization

The augmented Lagrangian associated with the problem (4.15) is

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - z) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2.$$

ADMM minimizes the objective function by alternatively optimizing the augmented Lagrangian with respect to variable x and z with method of multipliers. Formally, ADMM consists of the following iterations

$$x^{k+1} := \arg\min_x L_\rho(x, z^k, y^k) \tag{4.19}$$

$$z^{k+1} := \arg\min_z L_\rho(x^{k+1}, z, y^k) \tag{4.20}$$

$$y^{k+1} := y^k + \rho(Ax^{k+1} + Bz^{k+1} - c), \tag{4.21}$$

where $\rho > 0$. The algorithm consists of an x-minimization step (4.19)), a z-minimization step (4.20), and a dual variable update (4.21). The step size of dual update is equal to the augmented Lagrangian parameter ρ.

ADMM is a class of proximal algorithm because the x-update step (4.19) and the z-update step (4.20) is closely related to the proximal update of f and g. For simplicity, let us consider the case when $A = I,\ B = I$ and $c = 0$. Then the problem reduced to

$$\begin{aligned} &\min_{x,z} \ f(x) + g(z) \\ &\text{s.t. } x - z = 0. \end{aligned} \tag{4.22}$$

The associated augmented Lagrangian is

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(x - z) + \frac{\rho}{2}\|x - z\|_2^2.$$

Then the corresponding ADMM iteration can be expressed as

$$x^{k+1} := \text{prox}_{\lambda f}(z^k - u^k) \tag{4.23}$$

$$z^{k+1} := \text{prox}_{\lambda g}(x^{k+1} + u^k) \tag{4.24}$$

$$y^{k+1} := y^k + (x^{k+1} + z^{k+1}). \tag{4.25}$$

Convergence Under rather mild assumptions: 1) Function f and g are closed, proper and convex; 2) The unaugmented Lagrangian L_0 has a saddle point; The ADMM iterations satisfy the following:

- *Residual convergence.* $r^k \to 0$ as $k \to \infty$, *i.e.,* the iterates approach feasibility.
- *Objective convergence.* $f(x^k) + g(x^k) \to p^*$ as $k \to \infty$, *i.e.,* the objective function of the iterates approaches the optimal value.
- *Dual variable convergence.* $y^k \to y^*$ as $k \to \infty$, where y^* is a dual optimal point.

The convergence proof is detailed in (Boyd et al., 2011). In practice, ADMM converges to sufficient accuracy within a few tens of iterations, although convergence to high accuracy can be very slow.

Part III

Models and Methods

Chapter 5

Active Deformable Part Models

5.1 Introduction

Part-based models such as deformable part models (DPM) Felzenszwalb et al. (2010b) have become the state of the art in today's object detection methods. They offer powerful representations which can be learned from annotated datasets and capture both the appearance and the configuration of the parts. DPM-based detectors achieve unrivaled accuracy on standard datasets but their computational demand is high since it is proportional to the number of parts in the model and the number of locations at which to evaluate the part filters. Approaches for speeding-up the DPM inference such as cascades, branch-and-bound, and multi-resolution schemes, use the responses obtained from initial part-location evaluations to reduce the future computation. This paper introduces two novel ideas, which are missing in the state-of-the-art methods for speeding up DPM inference.

First, at each location in the image pyramid, a part-based detector has to make a decision: whether to evaluate more parts and in what order or to stop and predict a label. This decision can be regarded as a *planning problem*, whose state space consists of the set of previously used parts and the confidence of whether an object is present or not. While existing approaches rely on a predetermined sequence of parts, our approach optimizes the order in which to apply the part filters so that a minimal number of part evaluations

provides maximal classification accuracy at each location. Our second idea is to use a decision loss in the optimization, which quantifies the trade-off between false positive and false negative mistakes, instead of the threshold-based stopping criterion utilized by most other approaches. These ideas have enabled us to propose a novel object detector, Active Deformable Part Models, named so because of the active part selection. The detection procedure consists of two phases: an off-line phase, which learns a part scheduling policy from the training data and an online phase (inference), which uses the policy to optimize the detection task on test images. During inference, each image location starts with equal probabilities for object and background. The probabilities are updated sequentially based on the responses of the part filters suggested by the policy. At any time, depending on the probabilities, the policy might terminate predicting either a background label (which is what most cascaded methods take advantage of) or a positive label, in which case all unused part filters are evaluated in order to obtain the complete DPM score. Fig. 5.1 exemplifies the inference process.

We evaluated our approach on the PASCAL VOC 2007 and 2010 datasets Everingham et al. (2010) and achieved state of the art accuracy but with a 7 times reduction in the number of part-location evaluations and an average speed-up of 3 times compared to the cascade DPM Felzenszwalb et al. (2010a). This paper makes the following **contributions** to the state of the art in part-based object detection:

1. We obtain an active part selection policy which optimizes the order of the filter evaluations and balances number of evaluations used with the classification accuracy based on the scores obtained during inference.

2. The proposed detector achieves a significant speed-up versus the cascade DPM without sacrificing accuracy.

3. The approach can be generalized to any detection problem, which involves a linear additive score and uses several parts (stages) even if they are just SIFT points.

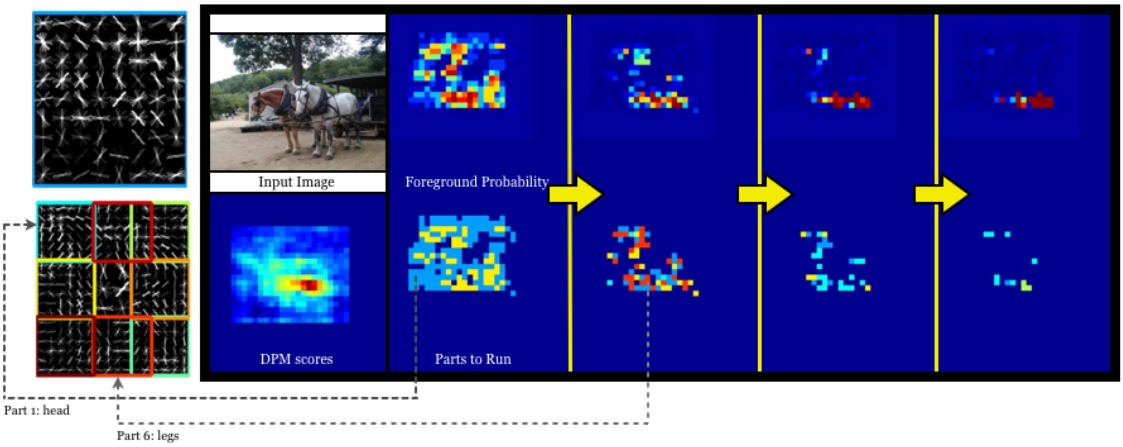


Figure 5.1: **Active DPM Overview**: A deformable part model trained on the PASCAL VOC 2007 horse class is shown with colored root and parts in the first column. The second column contains an input image and the original DPM scores as a baseline. The rest of the columns illustrate the inference process of the Active DPM, which proceeds in rounds. The foreground probability (of a horse being present) is maintained at each image location (top row) and is updated sequentially based on the responses of the part filters (high values are red; low values are blue). A policy (learned off-line) is used to select the best sequence of parts to apply at different locations. The bottom row shows the part filters applied at consecutive rounds with colors corresponding to the parts on the left. The policy decides to stop the inference at each location based on the confidence of foreground. As a result, the complete sequence of part filters is evaluated at very few locations, leading to a significant speed-up versus the traditional DPM inference. Our experiments show that the accuracy remains unaffected.

5.2 Related Work

We will refer to work on object detection that optimizes the inference stage rather than the representations since our representation is identical with DPM (Felzenszwalb et al., 2010b). Our method is inspired by an acceleration of the DPM object detector, the cascade DPM (Felzenszwalb et al., 2010a). However, while the sequence of parts evaluated

in the cascade DPM is pre-defined and a set of thresholds has to be determined empirically, our approach selects the part order and the stopping time at each location based on an optimization criterion. We find the next closest approach to be (Sznitman et al., 2013), which maintains a foreground probability at each stage of a multi-stage ensemble classifier and determines a stopping time based on the corresponding entropy. The difference of our approach is that it jointly optimizes the stage order and the stopping criterion. Kokkinos (Kokkinos, 2011) uses Branch-and-Bound (BB) to prioritize the search over image locations driven by an upper bound on the classification score. It is related to our approach in that object-less locations are easily detected and the search is guided in location space but with the difference that our policy proposes the next part to be tested in cases that no label can yet be given to a particular location. Earlier approaches (Lampert, 2010; Lampert et al., 2008; Lehmann et al., 2011b) relied on BB to constrain the search space of object detectors based on a sliding window or a Hough transform but without deformable parts. Another related group of approaches focuses on learning a sequence of object template tests in position, scale, and orientation space that minimizes the total computation time through a coarse-to-fine evaluation (Fleuret and Geman, 2001; Pedersoli et al., 2011).

The classic work by Viola and Jones (Viola and Jones, 2001) intorduced a cascade of classifiers whose order was determined by importance weights learned by AdaBoost. The approach was studied extensively in (Bourdev and Brandt, 2005; Brubaker et al., 2008; Gualdi et al., 2012; Lehmann et al., 2011a; Zhang et al., 2011). Recently, Dollar et al. (Dollár et al., 2012) introduced cross-talk cascades which allow detector responses to trigger or suppress the evaluation of weak classifiers in their neighborhood by exploiting the correlation of the classifier responses in the neighboring positions and scales. Weiss et al. (Weiss et al., 2012) used structured prediction cascades to optimize a function with two objectives: pose refinement and minimum filter evaluation cost. Sapp et al. (Sapp et al., 2010) learn a cascade of pictorial structures with increasing pose resolution by progressively filtering the pose state space. Its emphasis is on pre-filtering structures through

max-margin scoring rather than part locations so that human poses with weak individual part appearances can still be recovered. Rahtu et al. (Rahtu et al., 2011) use general "objectness" filters to produce location proposals which are fed into a cascade, designed to maximize the quality of the locations that advance to the next stage. Our approach is also related to and could be combined with active learning using Gaussian processes for classification (Kapoor et al., 2010).

Similarly to the closest approaches above (Felzenszwalb et al., 2010a; Kokkinos, 2011; Sznitman et al., 2013), our method aims to balance the number of part filter evaluations with the classification accuracy in part-based object detection. The novelty and the main advantage of our approach is that in addition it optimizes the part filter ordering. Since our "cascades" still run only on parts, we do not expect the approach to show higher accuracy than structured prediction cascades (Sapp et al., 2010) which consider more sophisticated representations that the pictorial structures in the DPM.

5.3 Technical approach

The state-of-the-art performance in object detection is obtained by star-structured models such as DPM Felzenszwalb et al. (2010b). A star-structured model of an object with n parts is formally defined by a $(n+2)$-tuple $(F_0, P_1, \ldots, P_n, b)$, where F_0 is a root filter, b is a real-valued bias term, and P_k are the part models. Each part model $P_k = (F_k, v_k, d_k)$ consists of a filter F_k, a position v_k of the part relative to the root, and the deformation coefficients d_k of a quadratic function specifying a deformation cost for placing the part away from v_k.

The object detector is applied in a sliding window fashion and outputs a prediction, $score(x)$, at each location x in an image pyramid, where $x = (r, c, l)$ specifies a position (r, c) in the l-th level (scale) of the pyramid. The space of all possible locations (position-scale tuples) in the image pyramid is denoted by $\mathcal{X}$. The response of the detector at a

given root location $x = (r, c, l) \in \mathcal{X}$ is:

$$\begin{aligned} \textit{score}(x) &= F_0' \cdot \phi(H, x) \\ &\quad + \sum_{k=1}^{n} \max_{x_k} \Big(F_k' \cdot \phi(H, x_k) - d_k \cdot \phi_d(\delta_k) \Big) + b, \end{aligned}$$

where $\phi(H, x)$ is the histogram of oriented gradients (HOG) feature vector at location x and $\delta_k := (r_k, c_k) - (2(r, c) + v_k)$ is the displacement of the k-th part from its anchor position v_k relative to the root location x. Each term in the above sum implicitly depends on x since the part locations x_k are chosen relative to root location at x. The score can be written as:

$$\textit{score}(x) = \textstyle\sum_{k=0}^{n} m_k(x) + b, \tag{5.1}$$

where $m_0(x) := F_0' \cdot \phi(H, x)$ and for $k > 0$, $m_k(x) := \max_{x_k} \big(F_k' \cdot \phi(H, x_k) - d_k \cdot \phi_d(\delta_k) \big)$. From this perspective, there is no difference between the root and the parts and we can think of the model as one consiting of $n + 1$ parts.

5.3.1 Score Likelihoods for the Parts

The object detection task requires labeling every $x \in \mathcal{X}$ with a label $y(x) \in \{\ominus, \oplus\}$. The traditional approach is to compute the complete score in (5.1) at every position-scale tuple $x \in \mathcal{X}$. In this paper, we argue that it is not necessary to obtain all $n + 1$ part responses in order to label a location x correctly. Treating the part scores as noisy observations of the true label $y(x)$, we choose an effective order in which to receive observations and an optimal time to stop. The stopping criterion is based on a trade-off between the cost of obtaining more observations and the cost of labeling the location x incorrectly.

Formally, the part scores $m_0, \ldots, m_n$ at a fixed location x are random variables, which depend on the input image, i.e. the true label $y(x)$. To emphasize this we denote them with upper-case letters M_k and their realizations with lower-case letters m_k. In order to predict an effective part order and stopping time, we need statistics which describe the part responses. Let $h^{\oplus}(m_0, m_1, \ldots, m_n)$ and $h^{\ominus}(m_0, m_1, \ldots, m_n)$ denote the joint probability

aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	**mean**
0.36	0.37	0.14	0.18	0.24	0.29	0.40	0.16	0.13	0.17	0.44	0.11	0.23	0.21	0.14	0.21	0.26	0.22	0.24	0.20	0.23

Table 5.1: Average correlation coefficients among pairs of part responses for all 20 classes in the PASCAL 2007 dataset

density functions (pdf) of the part scores conditioned on the true label being positive $y = \oplus$ and negative $y = \ominus$, respectively. We make the following assumption.

Assumption. *The responses of the parts of a star-structured model with a given root location $x \in \mathcal{X}$ are independent conditioned on the the true label $y(x)$, i.e.*

$$h^{\oplus}(m_0, m_1, \ldots, m_n) = \prod_{k=0}^{n} h_k^{\oplus}(m_k), \tag{5.2}$$
$$h^{\ominus}(m_0, m_1, \ldots, m_n) = \prod_{k=0}^{n} h_k^{\ominus}(m_k),$$

where $h_k^{\oplus}(m_k)$ is the pdf of $M_k \mid y = \oplus$ and $h_k^{\ominus}(m_k)$ is the pdf of $M_k \mid y = \ominus$.

We learn non-parametric representations for the $2(n+1)$ pdfs $\{h_k^{\oplus}, h_k^{\ominus}\}$ from an annotated set D of training images. We emphasize that the above assumption does not always hold in practice but simplifies the representation of the score likelihoods significantly[1]. Our algorithm for choosing a part order and a stopping time can be used without the independence assumption. However, we expect the performance to be similar while an unreasonable amount of training data would be required to learn a good representation of the joint pdfs. To evaluate the fidelity of the decoupled representation in (5.2) we computed correlation coefficients between all pairs of part responses (Table 5.1) for the classes in the PASCAL 2007 dataset. The mean over all classes, 0.23, indicates a weak correlation. We observed that the few highly correlated parts have identical appearances (e.g. car wheels) or a spatial overlap.

To learn representations for the score likelihoods, $\{h_k^{\oplus}, h_k^{\ominus}\}$, we collected a set of scores for each part from the the training set D. Given a positive example $I_i^{\oplus} \in D$

[1]Removing the independence assumption would require learning the 2 joint $(n+1)$ dimensional pdfs of the part scores in (5.2) and extracting the $2(n+1)$ marginals and the $2(n+1)(2^n - 1)$ conditionals of the form $h(m_k \mid m_I)$, where $I \subseteq \{0, \ldots, n\} \setminus \{k\}$.

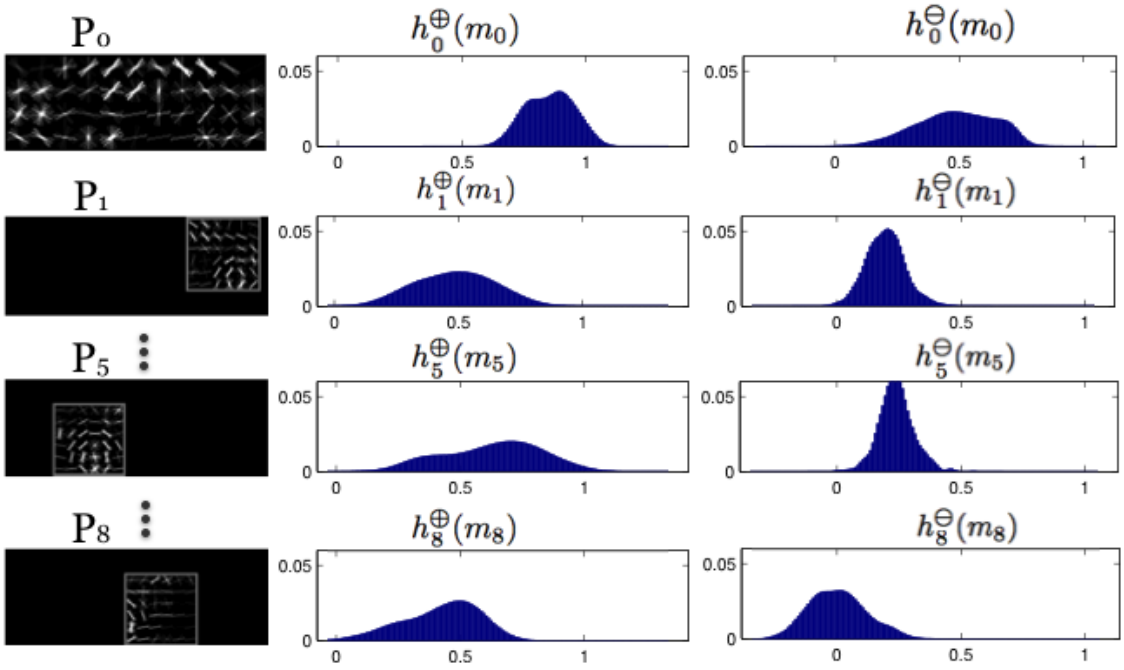


Figure 5.2: Score likelihoods for several parts from a car DPM model. The root (P_0) and three parts of the model are shown on the left. The corresponding positive and negative score likelihoods are shown on the right.

of a particular DPM component, the root was placed at the scale and position x^* of the top score within the ground-truth bounding box. The response m_0^i of the root filter was recorded. The parts were placed at their optimal locations relative to the root location x^* and their scores $m_k^i,\ k > 0$ were recorded as well. This procedure was repeated for all positive examples in D to obtain a set of scores $\{m_k^i \mid \oplus\}$ for each part k. For negative examples, x^* was selected randomly over all locations in the image pyramid and the same procedure was used to obtain the set $\{m_k^i \mid \ominus\}$. Kernel density estimation was applied to the score collections in order to obtain smooth approximations to $h_k^{\oplus}$ and $h_k^{\ominus}$. Fig. 5.2 shows several examples of the score likelihoods obtained from the part responses of a car model.

5.3.2 Active Part Selection

This section discusses how to select an ordered subset of the $n+1$ parts, which when applied at a given location $x \in \mathcal{X}$ has a small probability of mislabeling x. The detection at x proceeds in rounds $t = 0, \ldots, n+1$. The DPM inference applies the root and parts in a predefined topological ordering of the model structure. Here, we do not fix the order of the parts a priori. Instead, we select which part to run next *sequentially*, depending on the part responses obtained in the past. The part chosen at round t is denoted by $k(t)$ and can be any of the parts that have not been applied yet. We take a Bayesian approach and maintain a probability $p_t := \mathbb{P}(y = \oplus \mid m_{k(0)}, \ldots, m_{k(t-1)})$ of a positive label at location x conditioned on the part scores from the previous rounds. The state at time t consists of a binary vector $s_t \in \{0,1\}^{n+1}$ indicating which parts have already been used and the information state $p_t \in [0,1]$. Let $S_t := \{s \in \{0,1\}^{n+1} \mid \mathbf{1}^T s = t\}$ be the set[2] of possible values for s_t. At the start of a detection, $s_0 = \mathbf{0}$ and $p_0 = 1/2$, since no parts have been used and we have an uninformative prior for the true label.

Suppose that part $k(t)$ is applied at time t and its score is $m_{k(t)}$. The indicator vector s_t of used parts is updated as:

$$s_{t+1} = s_t + e_{k(t)}. \tag{5.3}$$

Due to the independence of the score likelihoods (5.2), the posterior label distribution is computed using Bayes rule:

$$p_{t+1} = \frac{h^{\oplus}_{k(t)}(m_{k(t)})}{h^{\oplus}_{k(t)}(m_{k(t)}) + h^{\ominus}_{k(t)}(m_{k(t)})} p_t. \tag{5.4}$$

In this setting, we seek a conditional plan π, which chooses which part to run next or stops and decides on a label for x. Formally, such a plan is called a *policy* and is a function $\pi(s,p) : \{0,1\}^{n+1} \times [0,1] \to \{\ominus, \oplus, 0, \ldots, n\}$, which depends on the previously used parts s and the label distribution p. An admissible policy does not allow part repetitions

[2] *Notation*: $\mathbf{1}$ denotes a vector with all elements equal to one, $\mathbf{0}$ denotes a vector with all elements equal to zero, and e_i denotes a vector with one in the i-th element and zero everywhere else.

and satisfies $\pi(\mathbf{1}, p) \in \{\ominus, \oplus\}$ for all $p \in [0, 1]$, i.e. has to choose a label after all parts have been used. The set of admissible policies is denoted by Π.

Let $\tau(\pi) := \inf\{t \geq 0 \mid \pi(s_t, p_t) \in \{\ominus, \oplus\}\} \leq n + 1$ denote the stopping time of policy $\pi \in \Pi$. Let $\hat{y}_\pi \in \{\ominus, \oplus\}$ denote the label guessed by policy π after its termination. We would like to choose a policy, which decides *quickly* and *correctly*. To formalize this, define the probability of making an error as $Pe(\pi) := \mathbb{P}(\hat{y}_\pi \neq y)$, where y is the hidden correct label of x.

Problem 2 (Active Part Selection)**.** *Given $\epsilon > 0$, choose an admissible part policy π with minimum expected stopping time and probability of error bounded by ϵ:*

$$\begin{aligned} \min_{\pi \in \Pi} \quad & \mathbb{E}[\tau(\pi)] \\ \textit{s.t.} \quad & Pe(\pi) \leq \epsilon, \end{aligned} \tag{5.5}$$

where the expectation is over the hidden label y and the part scores $M_{k(0)}, \ldots, M_{k(\tau-1)}$.

Note that if ϵ is chosen too small, (5.5) might be infeasible. In other words, even the best sequencing of the parts might not reduce the probability of error sufficiently. To avoid this issue, we relax the constraint in (5.5) by introducing a Lagrange multiplier $\lambda > 0$ as follows:

$$\min_{\pi \in \Pi} \quad \mathbb{E}[\tau(\pi)] + \lambda Pe(\pi). \tag{5.6}$$

The Lagrange multiplier λ can be interpreted as a cost paid for choosing an incorrect label. To elaborate on this, we rewrite the cost function as follows:

$$\begin{aligned} &\mathbb{E}\Big[\tau + \lambda \mathbb{E}_y\big[\mathbb{1}_{\{\hat{y} \neq y\}} \mid M_{k(0)}, \ldots, M_{k(\tau-1)}\big]\Big] \\ &\quad = \mathbb{E}\Big[\tau + \lambda \mathbb{1}_{\{\hat{y} \neq \oplus\}} \mathbb{P}\big(y = \oplus \mid M_{k(0)}, \ldots, M_{k(\tau-1)}\big) \\ &\qquad\qquad + \lambda \mathbb{1}_{\{\hat{y} \neq \ominus\}} \mathbb{P}\big(y = \ominus \mid M_{k(0)}, \ldots, M_{k(\tau-1)}\big)\Big] \\ &\quad = \mathbb{E}\Big[\tau + \lambda p_\tau \mathbb{1}_{\{\hat{y} = \ominus\}} + \lambda(1 - p_\tau)\mathbb{1}_{\{\hat{y} = \oplus\}}\Big]. \end{aligned}$$

The term λp_τ above is the cost paid if label $\hat{y} = \ominus$ is chosen incorrectly. Similarly, $\lambda(1 - p_\tau)$ is the cost paid if label $\hat{y} = \oplus$ is chosen incorrectly. To allow flexibility, we introduce separate costs λ_{fp} and λ_{fn} for false positive and false negative mistakes. The final form of the **Active Part Selection** problem is:

$$\min_{\pi \in \Pi} \quad \mathbb{E}\Big[\tau + \lambda_{fn} p_\tau \mathbb{1}_{\{\hat{y}=\ominus\}} + \lambda_{fp}(1 - p_\tau)\mathbb{1}_{\{\hat{y}=\oplus\}}\Big]. \tag{5.7}$$

Computing the Part Selection Policy Problem (5.7) can be solved using Dynamic Programming Bertsekas (1995). For a fixed policy $\pi \in \Pi$ and a given initial state $s_0 \in \{0,1\}^{n+1}$ and $p_0 \in [0,1]$, the value function:

$$V_\pi(s_0, p_0) := \mathbb{E}\Big[\tau + \lambda_{fn} p_\tau \mathbb{1}_{\{\hat{y}=\ominus\}} + \lambda_{fp}(1 - p_\tau)\mathbb{1}_{\{\hat{y}=\oplus\}}\Big],$$

is a well-defined quantity. The *optimal* policy π^* and the corresponding *optimal* value function are obtained as:

$$V^*(s_0, p_0) = \min_{\pi \in \Pi} V_\pi(s_0, p_0),$$
$$\pi^*(s_0, p_0) = \arg\max_{\pi \in \Pi} V_\pi(s_0, p_0).$$

To compute π^* we proceed backwards in time. Suppose that the policy has not terminated by time $t = n + 1$. Since there are no parts left to apply the policy is forced to terminate. Thus, $\tau = n + 1$ and $s_{n+1} = \mathbf{1}$ and for all $p \in [0,1]$ the optimal value function becomes:

$$\begin{aligned} V^*(\mathbf{1}, p) &= \min_{\hat{y} \in \{\ominus, \oplus\}} \Big\{ \lambda_{fn} p \mathbb{1}_{\{\hat{y}=\ominus\}} + \lambda_{fp}(1 - p)\mathbb{1}_{\{\hat{y}=\oplus\}} \Big\} \\ &= \min\{\lambda_{fn} p, \lambda_{fp}(1 - p)\}. \end{aligned} \tag{5.8}$$

The intermediate stage values for $t = n, \ldots, 0$, $s_t \in S_t$, and $p_t \in [0,1]$ are:

$$\begin{aligned} V^*(s_t, p_t) = \min\Big\{ & \lambda_{fn} p_t, \lambda_{fp}(1 - p_t), \\ & 1 + \min_{k \in \mathcal{A}(s_t)} \mathbb{E}_{M_k} V^*\left(s_t + e_k, \frac{h_k^{\oplus}(M_k) p_t}{h_k^{\oplus}(M_k) + h_k^{\ominus}(M_k)}\right) \Big\}, \end{aligned} \tag{5.9}$$

where $\mathcal{A}(s) := \{i \in \{0, \ldots, n\} \mid s_i = 0\}$ is the set of available (unused) parts[3]. The optimal policy is readily obtained from the optimal value function. At stage t, if the first term in (5.9) is smallest, the policy stops and chooses $\hat{y} = \ominus$; if the second term is smallest, the policy stops and chooses $\hat{y} = \oplus$; otherwise, the policy chooses to run the part k, which minimizes the expectation.

Alg. 14 summarizes the steps necessary to compute the optimal policy π^* using the score likelihoods $\{h_k^{\oplus}, h_k^{\ominus}\}$ from Sec. 5.3.1. The one dimensional space $[0, 1]$ of label probabilities p can be discretized into d bins in order to store the function π returned by Alg. 14. The memory required is $O(d2^{n+1})$ since the space $\{0, 1\}^{n+1}$ of used-part indicator vectors grows exponentially with the number of parts. Nevertheless, in practice the number of parts in a DPM is rarely more than 20 and Alg. 14 can be executed.

5.3.3 Active DPM Inference

A policy π is obtained *offline* using Alg. 14. In the online phase, π is used to select a sequence of parts to apply at each location $x \in \mathcal{X}$ in the image pyramid. Note that the labeling of each location is treated as an independent problem and proceeds in parallel. Alg. 24 summarizes the Active DPM inference process. Fig. 5.3 illustrates the policy making decisions at different image locations.

At the start of a detection at location x, $s_0 = \mathbf{0}$ since no parts have been used and $p_0 = 1/2$ since we have an uninformative label prior (Line 5). At each round t, the policy is queried to obtain either the next part to run or a predicted label for x (Line 7). Note that querying the policy is an $O(1)$ operation since it is stored as a lookup table. If the policy terminates and labels $y(x)$ as foreground (Line 8), all unused part filters are applied in order to obtain the final discriminative score in (5.1). On the other hand, if the policy terminates and labels $y(x)$ as background, no additional part filters are evaluated and the final score is set to $-\infty$ (Line 18). In this case, our algorithm makes computational savings

[3]Each score likelihood was discretized using 201 bins to obtain a histogram. Then, the expectation in (5.9) was computed as a sum over the bins. Alternatively, Monte Carlo integration can be performed by sampling from the Gaussian mixtures directly.

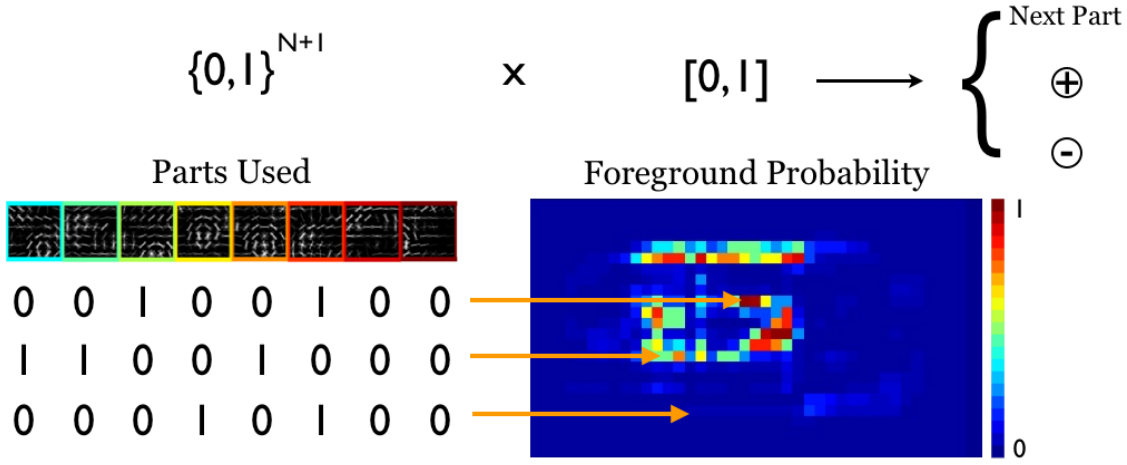


Figure 5.3: The detection process of ADPM is treated as a *planning problem*. At each location in the image pyramid, ADPM makes a decision: whether to evaluate more parts and in what order or to stop and predict a label. Such decision is based on a learned policy and determined on-the-fly conditioned on the history of part evaluations and the current foreground probability at each location.

compared to the DPM. The potential speed-up and the effect on accuracy are discussed in the Sec. 5.4. Finally, if the policy returns a part index k, the corresponding score $m_k(x)$ is computed by applying the part filter (Line 21). This operation is $O(|\Delta|)$, where Δ is the space of possible displacements for part k with respect to the root location x. Following the analysis in Felzenszwalb et al. (2010a), searching over the possible locations for part k is usually no more expensive than evaluating its linear filter F_k once because the spatial extent of the filter is of similar size as its range of displacement. This is the case because once F_k is applied at some location x_k, the resulting response $\Phi_k(x_k) = F_k' \cdot \phi(H, x_k)$ is cached to avoid recomputing it later. The use of a memorized version $\tilde{\Phi}_k(x_k)$ of $\Phi_k(x_k)$ amortizes the complexity of the search over Δ. The score m_k of part k is used to update the total score at x (Line 22). Then, the dynamics in (5.3) and (5.4) are used to update the state (s_t, p_t) (Line 23 - 24). Since the policy lookups and the state updates are all of $O(1)$ complexity, the worst-case complexity of Alg. 24 is $O(n|\mathcal{X}||\Delta|)$. The worst-case

complexity is the same as that of the DPM and the cascade DPM. The average running time of our algorithm depends on the total number of score m_k evaluations, which in turn depends on the choice of the parameters λ_{fn} and λ_{fp} and is the subject of the next section.

5.4 Experiments

5.4.1 Speed-Accuracy Trade-Off

The two parameters of the Active DPM (ADPM) method are the penalty, λ_{fp}, for incorrectly predicting background as foreground and the penalty, λ_{fn}, for incorrectly predicting foreground as background. The accuracy and the speed of the ADPM inference depend on these parameters. To get an intuition, consider making both λ_{fp} and λ_{fn} very small. The cost of an incorrect prediction will be negligible, thus encouraging the policy to sacrifice accuracy and stop immediately. In the other extreme, when both parameters are very large, the policy will delay the prediction as much as possible in order to obtain more information.

To evaluate the effect of the parameters, we compared the average precision (AP) and the number of part evaluations of Alg. 24 to those of the traditional DPM as a baseline. Let R_M be the total number of score $m_k(x)$ evaluations for $k > 0$ (excluding the root) over all locations $x \in \mathcal{X}$ performed by method M. For example, $R_{DPM} = n|\mathcal{X}|$ since the DPM evaluates all parts at all locations in $\mathcal{X}$. We define the **relative number of part evaluations** (RNPE) of our method (ADPM) versus method M as the ratio of R_M to R_{ADPM}. The AP and the RNPE versus DPM of APDM were evaluated on several classes from the PASCAL VOC 2007 training set. Fig. 5.4 shows the performance as the parameter $\lambda = \lambda_{fn} = \lambda_{fp}$ is varied. As expected, the AP increases while the RNPE decreases as the penalty of an incorrect declaration λ grows because ADPM evaluates more parts. The dip in RNPE for very low λ values is due to fact that ADPM starts reporting too many false-positives. In the case of a positive declaration all part responses need to be computed, which reduces the speed-up versus DPM.

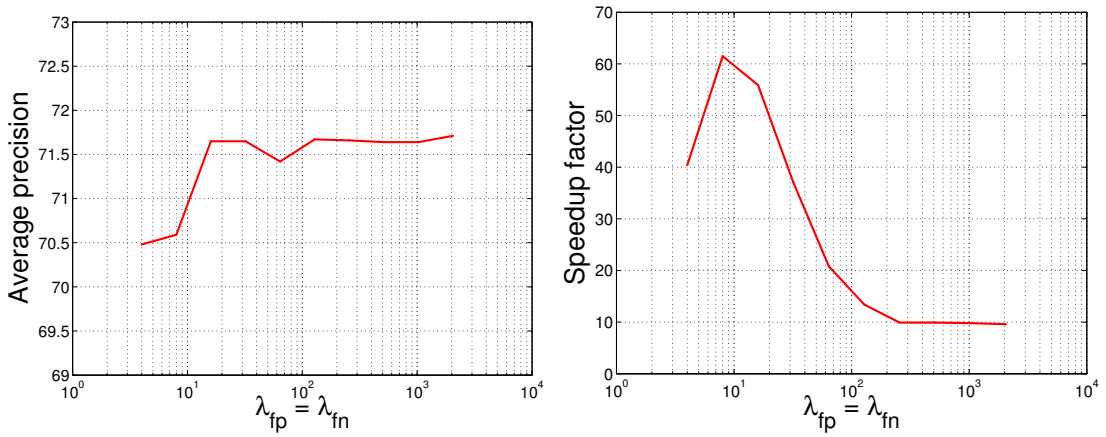


Figure 5.4: Average precision and relative number of part evaluations versus DPM as a function of the parameter $\lambda = \lambda_{fn} = \lambda_{fp}$ on a log scale. The curves are reported on the bus class from the VOC 2007 training set.

Since a positive declaration always requires $n+1$ part evaluations, we limit the number of false positive mistakes made by the policy by setting $\lambda_{fp} > \lambda_{fn}$. While this might hurt the accuracy, it will certainly result in significantly less part evaluations. To verify this intuition we performed experiments with $\lambda_{fp} > \lambda_{fn}$ on the PASCAL VOC 2007 dataset. Table 5.2 reports the AP and the RNPE versus DPM from a grid search over the parameter space. Generally, as the ratio between λ_{fp} and λ_{fn} increases, the RNPE increases while the AP decreases. Notice, however, that the increase in RNPE is significant, while the hit in accuracy is negligible.

5.4.2 Results

In this section we compare ADPM[4] versus two baselines, the DPM and the cascade DPM (Cascade) in terms of average precision (AP), relative number of part evaluations (RNPE), and relative wall-clock time speedup (Speedup). Experiments were carried out on all 20 classes in the PASCAL VOC 2007 and 2010 datasets. Publicly available PASCAL 2007

[4]ADPM code and trained policies are available at:
http://cis.upenn.edu/~menglong/adpm.html

and 2010 DPM and Cascade models were used for all three methods.

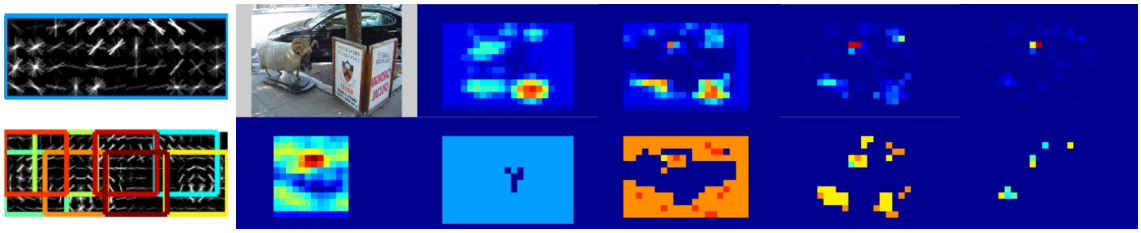
ADPM vs DPM: The inference process of ADPM is shown in detail on two input images in Fig. 5.1 and Fig. 5.5. The probability of a positive label p_t (top row) becomes more contrasted as additional parts are evaluated. The number of locations at which the algorithm has not terminated decreases rapidly as the time progresses. Visually, the locations with a maximal posterior are identical to the top scores obtained by the DPM. The order of parts chosen by the policy is indicative of their informativeness. For example, in Fig. 5.5 the wheel filters are applied first which agrees with intuition. In this example, the probability p_t remains low at the correct location for several iterations due to the occlusions. Nevertheless, the policy recognizes that it should not terminate and as more parts are evaluated, the posterior reflects the correct location of the highest DPM score.

ADPM was compared to DPM in terms of AP and RNPE to demonstrate the ability of ADPM to reduce the number of necessary part evaluations with minimal loss in accuracy irrespective of the features used. The ADPM parameters were set to $\lambda_{fp} = 20$ and $\lambda_{fn} = 5$ based on the analysis in Sec. 5.4.1. Table 5.3 shows that ADPM achieves a significant decrease (about 90 times on average) in the number of evaluated parts compared to DPM, while the loss in accuracy is negligible. The precision-recall curves of the two methods are shown for several classes in Fig. 5.6.

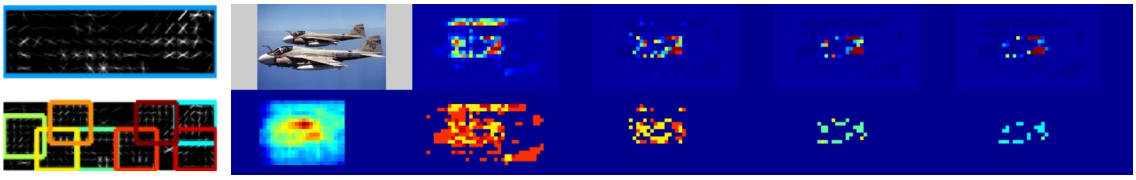
ADPM vs Cascade: The improvement in detection speed achieved by ADPM is demonstrated via a comparison to Cascade in terms of AP, RNPE, and wall-clock time (in seconds). Note that Cascade's implementation makes use of PCA-projected (top five dimensions) HOG features, which are very fast to compute. During inference, Cascade prunes the image locations in two passes. In the first pass, the locations are filtered using the PCA-projections and the low-scoring ones are discarded. In the second pass, the remaining locations are filtered using the full-dimensional features. To make a fair comparison, we adopted a similar two-stage approach for the active part selection. An additional policy was learned using PCA score likelihoods and was used to schedule PCA filters during the first pass. The locations, which were selected as foreground in the first stage, were

filtered again, using the original policy to select the order of the full-dimensional filters. The parameters λ_{fp} and λ_{fn} were set to 20 and 5 for the PCA policy and to 50 and 5 for the full-dimensional policy. A higher λ_{fp} was chosen to make the prediction more precise (albeit slower) during the second stage. Deformation pruning was not used for either method. Table 5.4 summarizes the results.

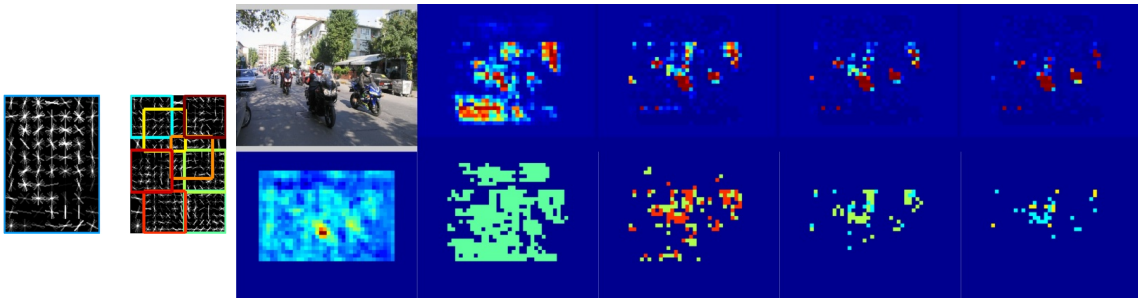
A discrepancy in the speedup of ADPM versus Cascade is observed in Table 5.4. On average, ADPM is 7 times faster than Cascade in terms of RNPE but only 3 times faster in seconds. A breakdown of the computational time during inference on a single image is shown in Table 5.5. We observe that the ratios of part evaluations and of seconds are consistent within individual stages (PCA and full). However, a single filter evaluation during the full-filter stage is significantly slower than one during the PCA stage. This does not affect the cumulative RNPE but lowers the combined seconds ratio. While ADPM is significanlty faster than Cascade during the PCA stage, the speedup (in seconds) is reduced during the slower full-dimensional stage.



(a) class: car



(b) class: aeroplane



(c) class: motorbike

Figure 5.5: Illustration of the ADPM inference process on a car example. The DPM model with colored root and parts is shown on the left. The top row on the right consists of the input image and the evolution of the positive label probability (p_t) for $t \in \{1, 2, 3, 4\}$ (high values are red; low values are blue). The bottom row consists of the full DPM $score(x)$ and a visualization of the parts applied at different locations at time t. The pixel colors correspond to the part colors on the left. In this example, despite the car being heavily occluded, ADPM converges to the correct location after four iterations.

Algorithm 1: Active Part Selection

Input: Score likelihoods $\{h_k^{\ominus}, h_k^{\oplus}\}_{k=0}^{n}$ for all parts,

false positive cost λ_{fp}, false negative cost λ_{fn}

Output: Policy $\pi : \{0,1\}^{n+1} \times [0,1] \to \{\ominus, \oplus, 0, \ldots, n\}$

$S_t := \{s \in \{0,1\}^{n+1} \mid \mathbf{1}^T s = t\}$;

$\mathcal{A}(s) := \{i \in \{0,\ldots,n\} \mid s_i = 0\}, \quad \forall s \in \{0,1\}^{n+1}$;

$V(\mathbf{1}, p) := \min\{\lambda_{fn} p, \lambda_{fp}(1-p)\}, \quad \forall p \in [0,1]$;

$$\pi(\mathbf{1}, p) := \begin{cases} \ominus, & \lambda_{fn} p \le \lambda_{fp}(1-p), \\ \oplus, & \text{otherwise;} \end{cases}$$

for $t = n, n-1, \ldots, 0$ **do**

  for $s \in S_t$ **do**

    for $k \in \mathcal{A}(s)$ **do**

      $$Q(s,p,k) := \mathbb{E}_{M_k} V\left(s + e_k, \frac{h_k^{\oplus}(M_k) p}{h_k^{\oplus}(M_k) + h_k^{\ominus}(M_k)}\right);$$

    end

    $$V(s,p) := \min\left\{\lambda_{fn} p, \lambda_{fp}(1-p), 1 + \min_{k \in \mathcal{A}(s)} Q(s,p,k)\right\};$$

    $$\pi(s,p) := \begin{cases} \ominus, & V(s,p) = \lambda_{fn} p, \\ \oplus, & V(s,p) = \lambda_{fp}(1-p), \\ \arg\min_{k \in \mathcal{A}(s)} Q(s,p,k), & \text{otherwise;} \end{cases}$$

  end

end

return π;

Algorithm 2: Active DPM Inference

Input: Image pyramid, model $(F_0, P_1, \ldots, P_n, b)$,

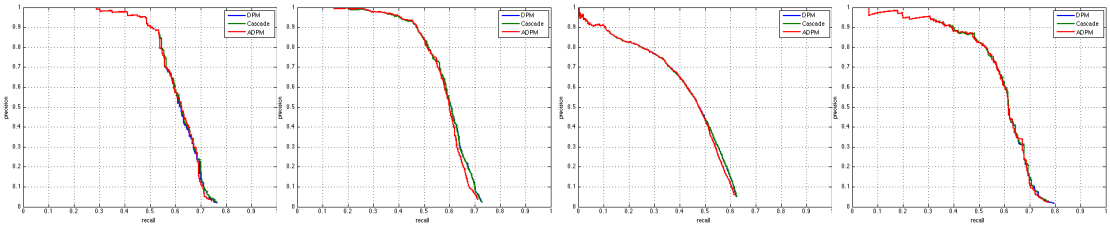
score likelihoods $\{h_k^{\ominus}, h_k^{\oplus}\}_{k=0}^{n}$ for all parts, policy π

Output: $score(x)$ at all locations $x \in \mathcal{X}$ in the image pyramid

for $x \in 1 \ldots |\mathcal{X}|$ **do**
  $s_0 := \mathbf{0}$; $p_0 = 0.5$; $score(x) := 0$;
  for $t = 0, 1, \ldots, n$ **do**
    $k := \pi(s_t, p_t)$; // Lookup next best part
    if $k = \oplus$ **then**
      // Labeled as foreground
      for $i \in \{0, 1, \ldots, n\}$ **do**
        if $s_t(i) = 0$ **then**
          Compute score $m_k(x)$ for part k ; // $O(|\Delta|)$
          $score(x) := score(x) + m_k(x)$;
        end
      end
      $score(x) := score(x) + b$; // Add bias to final score
      break;
    else if $k = \ominus$ **then**
      $score(x) := -\infty$; // Labeled as background
      break;
    else
      Compute score $m_k(x)$ for part k ; // $O(|\Delta|)$
      $score(x) := score(x) + m_k(x)$;
      $p_{t+1} := \frac{h_k^{\oplus}(m_k(x))p_t}{h_k^{\oplus}(m_k(x)) + h_k^{\ominus}(m_k(x))}$; // Update probability
      $s_{t+1} = s_t + e_k$;
    end
  end
end

Average Precision						RNPE vs DPM					
$\lambda_{fp}/\lambda_{fn}$	4	8	16	32	64	$\lambda_{fp}/\lambda_{fn}$	4	8	16	32	64
4	70.3					4	40.4				
8	70.0	71.0				8	80.7	61.5			
16	69.6	71.1	71.5			16	118.6	74.5	55.9		
32	70.5	70.7	71.6	71.6		32	178.3	82.1	59.8	37.0	
64	67.3	69.6	71.5	71.6	71.4	64	186.9	96.4	56.2	34.5	20.8

Table 5.2: Average precision and relative number of part evaluations versus DPM obtained on the bus class from PASCAL VOC 2007 training set. A grid search over the parameter space $(\lambda_{fp}, \lambda_{fn}) \in \{4, 8, \ldots, 64\} \times \{4, 8, \ldots, 64\}$ with $\lambda_{fp} \geq \lambda_{fn}$ is shown.



(a) class: bicycle (b) class: car (c) class: person (d) class: horse

Figure 5.6: Precision recall curves for bicycle, car, person, and horse classes from PASCAL 2007. Our method's accuracy ties with the baselines.

VOC2007	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	**mean**
DPM RNPE	102.8	106.7	63.7	79.7	58.1	155.2	44.5	40.0	58.9	71.8	69.9	49.2	51.0	59.6	45.3	49.0	62.6	68.6	79.0	100.6	**70.8**
DPM AP	33.2	60.3	10.2	16.1	27.3	54.3	58.2	23.0	20.0	24.1	26.7	12.7	58.1	48.2	43.2	12.0	21.1	36.1	46.0	43.5	**33.7**
ADPM AP	33.5	59.8	9.8	15.3	27.6	52.5	57.6	22.1	20.1	24.6	24.9	12.3	57.6	48.4	42.8	12.0	20.4	35.7	46.3	43.2	**33.3**
VOC2010	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	**mean**
DPM RNPE	110.0	100.8	47.9	98.8	111.8	214.4	75.6	202.5	150.8	147.2	62.4	126.2	133.7	187.1	114.4	59.3	24.3	131.2	143.8	106.0	**117.4**
DPM AP	45.6	49.0	11.0	11.6	27.2	50.5	43.1	23.6	17.2	23.2	10.7	20.5	42.5	44.5	41.3	8.7	29.0	18.7	40.0	34.5	**29.6**
ADPM AP	45.3	49.1	10.2	12.2	26.9	50.6	41.9	22.7	16.5	22.8	10.6	19.7	40.8	44.5	36.8	8.3	29.1	18.6	39.7	34.5	**29.1**

Table 5.3: Average precision (AP) and relative number of part evaluations (RNPE) of DPM versus ADPM on all 20 classes in PASCAL VOC 2007 and 2010.

VOC2007	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	**mean**
Cascade RNPE	5.93	5.35	9.17	6.09	8.14	3.06	5.61	4.51	6.30	4.03	4.83	7.77	3.61	6.67	17.8	9.84	3.82	2.43	2.89	6.97	**6.24**
ADPM Speedup	3.14	1.60	8.21	4.57	3.36	1.67	2.11	1.54	3.12	1.63	1.28	2.72	1.07	1.50	3.59	6.15	2.92	1.10	1.11	3.26	**2.78**
Cascade AP	33.2	60.8	10.2	16.1	27.3	54.1	58.1	23.0	20.0	24.2	26.8	12.7	58.1	48.2	43.2	12.0	20.1	35.8	46.0	43.4	**33.7**
ADPM AP	31.7	59.0	9.70	14.9	27.5	51.4	56.7	22.1	20.4	24.0	24.7	12.4	57.7	48.5	41.7	11.6	20.4	35.9	45.8	42.8	**33.0**
VOC2010	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	**mean**
Cascade RNPE	7.28	2.66	14.80	7.83	12.22	5.47	6.29	6.33	9.72	4.16	3.74	10.77	3.21	9.68	21.43	12.21	3.23	4.58	3.98	8.17	**7.89**
ADPM Speedup	2.15	1.28	7.58	5.93	4.68	2.79	2.28	2.44	3.72	2.42	1.52	2.76	1.57	2.93	4.72	8.24	1.42	1.81	1.47	3.41	**3.26**
Cascade AP	45.5	48.9	11.0	11.6	27.2	50.5	43.1	23.6	17.2	23.1	10.7	20.5	42.4	44.5	41.3	8.7	29.0	18.7	40.1	34.4	**29.6**
ADPM AP	44.5	49.2	9.5	11.6	25.9	50.6	41.7	22.5	16.9	22.0	9.8	19.8	41.1	45.1	40.2	7.4	28.5	18.3	38.0	34.5	**28.8**

Table 5.4: Average precision (AP), relative number of part evaluations (RNPE), and relative wall-clock time speedup (Speedup) of ADPM versus Cascade on all 20 classes in PASCAL VOC 2007 and 2010.

	PCA no cache	PCA cache	PE	Full no cache	Full cache	PE	Total no cache	Total cache	Total PE
CASCADE	4.34s	0.67s	208K	0.13s	0.08s	1.1K	4.50s	0.79s	209K
ADPM	0.62s	0.06s	36K	0.06s	0.04s	0.6K	0.79s	0.19s	37K

Table 5.5: An example demonstrating the computational time breakdown during inference of ADPM and Cascade on a single image. The number of part evaluations (PE) and the inference time (in seconds) is recorded for the PCA and the full-dimensional stages. The results are reported once without and once with cache use. The number of part evaluations is independent of caching. The total times are not equal to the sum of the two stages because of the additional but minimal time spent in I/O operations.

Chapter 6

3D Object Detection and Pose Estimation of Object Instances

6.1 Introduction

In this paper, we address the problem of a robot grasping 3D objects of known 3D shape from their projections in single images of cluttered scenes. In the context of object grasping and manipulation, object recognition has always been defined as simultaneous detection and segmentation in the 2D image and 3D localization. 3D object recognition has experienced a revived interest in both the robotics and computer vision communities with RGB-D sensors having simplified the foreground-background segmentation problem. Nevertheless, difficulties remain as such sensors cannot generally be used in outdoor environments yet.

The goal of this paper is to detect and localize objects in single view RGB images of environments containing arbitrary ambient illumination and substantial clutter for the purpose of autonomous grasping. Objects can be of arbitrary color and interior texture and, thus, we assume knowledge of only their 3D model without any appearance/texture information. Using 3D models makes an object detector immune to intra-class texture variations.

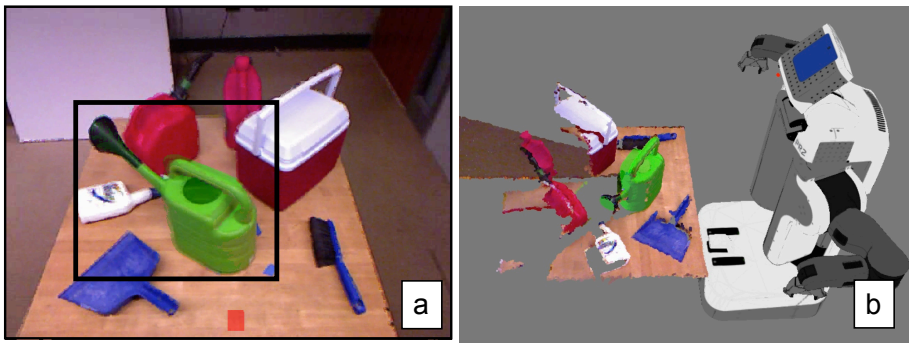


Figure 6.1: Demonstration of the proposed approach on a PR2 robot platform. a) Single view input image, with the object of interest highlighted with a black rectangle. b) Object model (in green) is projected with the estimated pose in 3D, ready for grasping. The Kinect point cloud is shown for the purpose of visualization.

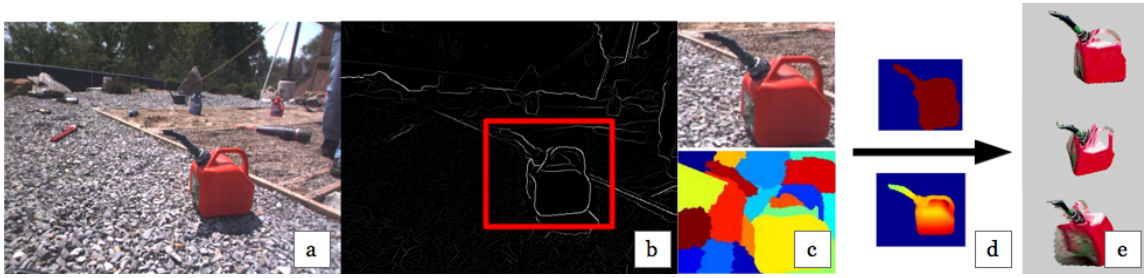


Figure 6.2: Overview of the proposed approach. From left-to-right: a) The input image. b) S-DPM inferences on the gPb contour image yielding an object detection hypothesis. c) The hypothesis bounding box (red) is segmented into superpixels. d) The set of superpixels with the closest chordiogram distance to the model silhouette is selected. Pose is iteratively refined such that the model projection aligns well with the foreground mask silhouette. e) To visualize the pose accuracy, the side of the 3D model facing the camera is textured with the corresponding 2D pixel color; three textured synthetic views of the final pose estimate are shown.

We further abstract the 3D model by only using its 2D silhouette and thus detection is driven by the shape of the 3D object's projected occluding boundary. Object silhouettes

with corresponding viewpoints that are tightly clustered on the viewsphere are used as positive exemplars to train the state-of-the-art Deformable Parts Model (DPM) discriminative classifier (Felzenszwalb et al., 2010c). We term this shape-aware version S-DPM. This detector simultaneously detects the object and coarsely estimates the object′s pose. The focus of the current paper is on instance-based rather than category-based object detection and localization; however, our approach can be extended to multiple instance category recognition since S-DPM is agnostic to whether the positive exemplars are multiple poses from a single instance (as considered in the current paper) or multiple poses from multiple instances.

We propose to use an S-DPM classifier as a first high recall step yielding several bounding box hypotheses. Given these hypotheses, we solve for segmentation and localization simultaneously. After over-segmenting the hypothesis region into superpixels, we select the superpixels that best match a model boundary using a shape-based descriptor, the chordiogram (Toshev et al., 2012). A chordiogram-based matching distance is used to compute the foreground segment and rerank the hypotheses. Finally, using the full 3D model we estimate all 6-DOF of the object by efficiently iterating on the pose and computing matches using dynamic programming.

Our approach advances the state-of-the-art as follows:

- In terms of assumptions, our approach is among the few in the literature that can detect 3D objects in single images of cluttered scenes independent of their appearance.
- It combines the high recall of an existing discriminative classifier with the high precision of a holistic shape descriptor achieving a simultaneous segmentation and detection reranking.
- Due to the segmentation, it selects the correct image contours to use for 3D pose refinement, a task that was previously only possible with stereo or depth sensors.

An overview of the components of our approach is shown in Fig. 6.2. In the video

supplement, we demonstrate our approach with a (PR2) robot grasping 3D objects on a cluttered table based on a single view RGB image. Figure 6.8 shows an example of the process. We report 3D pose accuracy by comparing the estimated pose rendered by the proposed approach with a ground truth point cloud recovered with a RGB-D sensor. Such grasping capability with accurate pose is crucial for robot operation, where popular RGB-D sensors cannot be used (e.g., outdoors) and stereo sensors are challenged by the uniformity of the object's appearance within their boundary. We also document an extensive evaluation on outdoor imagery with diverse backgrounds. The dataset contains a set of 3D object models, annotated single-view imagery of heavily cluttered outdoor scenes[1], and indoor imagery of cluttered tabletops in RGB-D images.

6.2 Related Work

Geometry-based object recognition arguably outdates appearance-based approaches. A major advantage of these approaches is their invariance to material properties, viewpoint and illumination. We first survey approaches that use a 3D model, either synthetic or obtained from 3D reconstruction. Next, we describe approaches using multiple view exemplars annotated with their pose. We close with a brief description of 2D shape-based approaches and approaches applied to RGB-D test data.

Early approaches based on using explicit 3D models are summarized in Grimson's book (Grimson, 1990) and focus on efficient techniques for voting in pose space. Horaud (Horaud, 1987) investigated object recognition under perspective projection using a constructive algorithm for objects that contain straight contours and planar faces. Hausler (Häusler and Ritter, 1999) derived an analytical method for alignment under perspective projection using the Hough transform and global geometric constraints. Aspect graphs in their strict mathematical definition (each node sees the same set of singularities) were not

[1]The annotated dataset and 3D models are available at the project page: http://www.seas.upenn.edu/~menglong/outdoor-3d-objects.html

considered practical enough for recognition tasks but the notion of sampling of the view-space for the purpose of recognition was introduced again in (Cyr and Kimia, 2001) which were applied in single images with no background. A Bayesian method for 3D reconstruction from a single image was proposed based on the contours of objects with sharp surface intersections (Han and Zhu, 2003). Sethi et al. (Sethi et al., 2004) compute global invariant signatures for each object from its silhouette under weak perspective projection. This approach was later extended (Lazebnik et al., 2002) to perspective projection by sampling a large set of epipoles for each image to account for a range of potential viewpoints. Liebelt et al. work with a view space of rendered models in (Liebelt et al., 2008a) and a generative geometry representation is developed in (Liebelt and Schmid, 2010). Villamizar et al. (Villamizar et al., 2011) use a shared feature database that creates pose hypotheses verified by a Random Fern pose specific classifier. In (Glasner et al., 2011a), a 3D point cloud model is extracted from multiple view exemplars for clustering pose specific appearance features. Others extend deformable part models to combine viewpoint estimates and 3D parts consistent across viewpoints, e.g., (Pepik et al., 2012a). In (Hao et al., 2013), a novel combination of local and global geometric cues was used to filter 2D image to 3D model correspondences.

Others have pursued approaches that not only segment the object and estimate the 3D pose but also adjusts the 3D shape of the object model. For instance, Gaussian Process Latent Variable Models were used for the dimensionality reduction of the manifold of shapes and a two-step iteration optimizes over shape and pose, respectively (Prisacariu et al., 2013). The drawback of these approaches is that in the case of scene clutter they do not consider the selection of image contours. Further, in some cases tracking is used for finding the correct shape. This limits applicability to the analysis of image sequences, rather than a single image, as is the focus in the current paper.

Our approach resembles early proposals that avoid appearance cues and uses only the silhouette boundary, e.g., (Cyr and Kimia, 2001). None of the above or the exemplar-based approaches surveyed below address the amount of clutter considered here and in

most cases the object of interest occupies a significant portion of the field of view.

Early view exemplar-based approaches typically assume an orthographic projection model that simplifies the analysis. Ullman (Ullman and Basri, 1991) represented a 3D object by a linear combination of a small number of images enabling an alignment of the unknown object with a model by computing the coefficients of the linear combination, and, thus, reducing the problem to 2D. In (Basri, 1993), this approach was generalized to objects bounded by smooth surfaces, under orthographic projection, based on the estimation of curvature from three or five images. Much of the multiview object detector work based on discrete 2D views (e.g., (Gu and Ren, 2010a)) has been founded on successful approaches to single view object detection, e.g., (Felzenszwalb et al., 2010c). Savarese and Fei-Fei (Savarese and Fei-Fei, 2007) presented an approach for object categorization that combines appearance-based descriptors including the canonical view for each part, and transformations between parts. This approach reasons about 3D surfaces based on image appearance features. In (Payet and Todorovic, 2011), detection is achieved simultaneously with contour and pose selection using convex relaxation. Hsiao et al. (Hsiao et al., 2010) also use exemplars for feature correspondences and show that ambiguity should be resolved during hypothesis testing and not at the matching phase. A drawback of these approaches is their reliance on discriminative texture-based features that are hardly present for the types of textureless objects considered in the current paper.

As far as RGB-D training and test examples are concerned, the most general and representative approach is (Lai et al., 2011). Here, an object-pose tree structure was proposed that simultaneously detects and selects the correct object category and instance, and refines the pose. In (Rusu et al., 2010), a viewpoint feature histogram is proposed for detection and pose estimation. Several similar representations are now available in the Point Cloud Library (PCL) (Rusu and Cousins, 2011). We will not delve here into approaches that extract the target objects during scene parsing in RGB-D images but refer the reader to (Koppula et al., 2011) and the citations therein.

The 2D-shape descriptor, chordiogram (Toshev et al., 2012), we use belongs to approaches based on the optimal assembly of image regions. Given an over-segmented image (i.e., superpixels), these approaches determine a subset of spatially contiguous regions whose collective shape (Toshev et al., 2012) or appearance (Vijayanarasimhan and Grauman, 2011) features optimize a particular similarity measure with respect to a given object model. An appealing property of region-based methods is that they specify the image domain where the object-related features are computed and thus avoid contaminating objected-related measurements from background clutter.

6.3 Technical approach

An overview of the components of our approach is shown in Fig. 6.2. 3D models are acquired using a low-cost depth sensor (Sec. 6.3.1). To detect an object robustly based *only* on shape information, the gPb contour detector Arbelaez et al. (2011) is applied to the RGB input imagery (Sec. 6.3.2). Detected contours are fed into a parts-based object detector trained on model silhouettes (Sec. 6.3.3). Detection hypotheses are over-segmented and shape verification simultaneously computes the foreground segments and reranks the hypotheses (Sec. 6.3.5). Section 6.3.4 describes the shape descriptor used for shape verification. The obtained object mask enables the application of an iterative 3D pose refinement algorithm to accurately recover the 6-DOF object pose based on the initial coarse pose estimate rendered by the object detector (Sec. 6.3.6).

6.3.1 3D model acquisition and rendering

3D CAD models have been shown to be very useful for object detection and pose estimation both in 2D images and 3D point clouds. We utilize a low-cost RGB-D depth sensor and a dense surface reconstruction algorithm, KinectFusion Izadi et al. (2011), to efficiently reconstruct 3D object models from the depth measurements of real objects. The 3D object model is acquired on a turntable with the camera pointing in a fixed position.

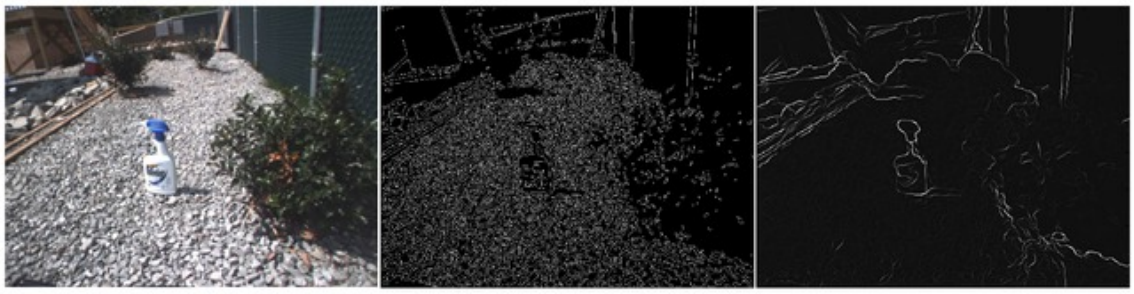


Figure 6.3: Comparison of the two edge detection results on same image. (left-to-right) Input image, Canny edge and gPb, respectively.

After the model is reconstructed with the scene, we manually remove the background and fill holes in the model.

To render object silhouettes from arbitrary poses, we synthesize a virtual camera at discretized viewpoints around the object center at a fixed distance. Each viewpoint is parameterized by the azimuth, a, elevation, e, and distance, d, of the camera relative to the object. Viewpoints are uniformly sampled on the viewsphere at a fixed distance and at every ten degrees for both the azimuth and elevation.

6.3.2 Image feature

Our approach to shape-based recognition benefits from recent advances in image contour detection. In unconstrained natural environments, the Canny edge detector Canny (1986) generally responds uniformly to both object occlusion boundaries and texture. One can falsely piece together the silhouette of a target object from a dense set of edge pixels. The state-of-the-art contour detection algorithm gPb Arbelaez et al. (2011) computes the likelihood of each pixel being an object contour and thus suppresses many edges due to texture/clutter. Figure 6.3 shows an example of Canny edge detection and gPb on the same input image. Compared to Canny edges, gPb suppresses ubiquitous edge responses from background clutter.

Given detected contours in the image, we seek to localize the subset of contour pixels

that best represent the object silhouette. We will show that for cluttered scenes, discriminative power is essential to achieve high recall with desired precision.

6.3.3 Object detection

The Deformable Parts Model (DPM) Felzenszwalb et al. (2010c) is one of the most successful object detector. DPM is a star-structured conditional random field (CRF), with a root part, F_0, capturing the holistic appearance of the object and several parts $(P_0, \ldots, P_n)$ connected to the root where $P_i = (F_i, v_i, s_i, a_i, b_i)$. Each model part has a default relative position (the anchor point), v_i, with respect to the root position. Parts are also allowed to translate around the anchor point with a quadratic offset distance penalty, parameterized by the coefficients a_i and b_i. The anchor points are learned from the training data and the scales of the root and parts, s_i, are fixed. The detection score is defined as:

$$\sum_{i=0}^{n} F_i \cdot \phi(H, p_i) - \sum_{i=1}^{n} a_i \cdot (\tilde{x}_i, \tilde{y}_i) + b_i \cdot (\tilde{x_i}^2, \tilde{y_i}^2), \tag{6.1}$$

where $\phi(H, p_i)$ is the histogram of gradients (HOG) Dalal and Triggs (2005a) feature extracted at image location p_i, and $(\tilde{x}_i, \tilde{y}_i)$ is the offset to the part anchor point with respect to the root position p_0. At test time, the root and part model weights are each separately convolved with the HOG feature of the input image. Due to the star structure of the model, maximizing the above score function at each image location can be computed efficiently via dynamic programming. To deal with intra-class variation, DPM is generalized by composing several components, each trained on a subset of training instances of similar aspect ratio. We refer to Felzenszwalb et al. (2010c) for more details.

To simultaneously detect an object and coarsely estimate its pose from the edge map using only model silhouette shape information, we train a shape-aware modified version of DPM, which we term S-DPM. Each component of the learned S-DPM corresponds to a coarse pose of the object. More specifically, the silhouettes of the synthetic views of the object are clustered into 16 discrete poses by grouping nearby viewpoints. An S-DPM component is trained based on the silhouettes of a coarse pose cluster used as positive

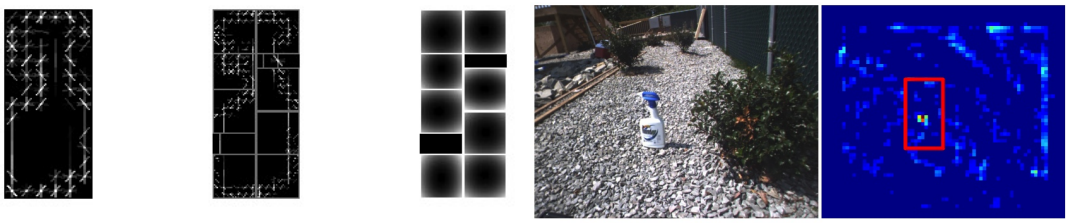


Figure 6.4: Spray bottle detection using S-DPM. (first row, left-to-right) Root appearance model, part appearance models centered at their respective anchor points and the quadratic deformation cost; brighter regions indicate larger penalty cost. (second row) Input image and detection response map of the spray-bottle; red, yellow and blue indicate large, intermediate and low detection responses, respectively.

training data and silhouettes of other poses and objects and random background edges used as negatives. Figure 6.4 shows an example of a trained spray bottle model. During inference, each of the model components are evaluated on the input contour imagery and the hypotheses with a detection score above a threshold are retained. Detections of different components are combined via non-maximum suppression. This step retains high scoring detections and filters out neighboring lower scoring ones whose corresponding 2D bounding box overlaps with that of the local maximum by greater than 50% (PASCAL criteria Everingham et al. (2010)). The coarse pose of the object is determined by the maximum scoring component at each image location.

6.3.4 Shape descriptor

We represent the holistic shape of each S-DPM detected object with the chordiogram descriptor Toshev et al. (2012). Given the object silhouette, this representation captures the distribution of geometric relationships (relative location and normals) between pairs of boundary edges, termed chords. Formally, the chordiogram is a K-dimensional histogram of all chord features on the boundary of a segmented object. A chord is a pair of points (p, q) on the boundary points. Chord feature $d_{pq} = (l_{pq}, \psi_{pq}, \theta_p - \psi_{pq}, \theta_q - \psi_{pq})^\top$ is defined

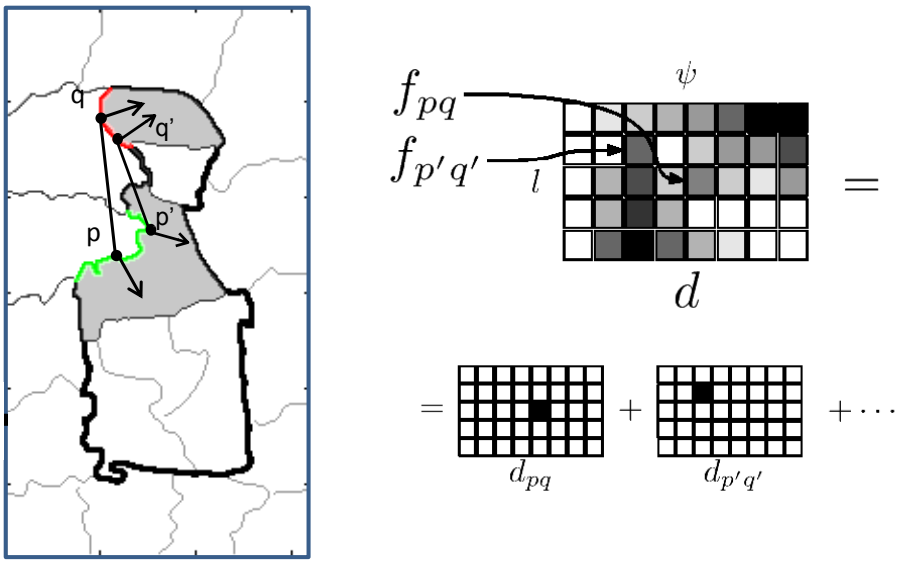


Figure 6.5: Chordiogram construction. The bold boundary in the image on the left denotes the correct superpixel boundary of the object. Gray highlighted regions denote the foreground superpixels under consideration. At the two chords, pq and $p'q'$, the features, f_{pq} and $f_{p'q'}$, fall into different bins of the histogram, i.e., the chordiogram shown on the right. At each boundary point, the foreground selection of bordering superpixels defines the normal direction.

by chord vector length l_{pq}, orientation ψ_{pq} and normals θ_p and θ_q of the object boundary at p and q. The edge normal direction points towards the segment interior to distinguish the same edge with different foreground selection of bordering superpixels. Figure 6.5 shows two examples of chord features and their corresponding chordiogram feature bins when the bordering foreground superpixels differ. The chordiogram is translation invariant since it only relates the relative position of boundary pixels rather than the absolute position in the image.

6.3.5 Shape verification for silhouette extraction

We use the chordiogram descriptor for two tasks: (i) to recover the object foreground mask (i.e., the silhouette) for accurate 3D pose estimation and (ii) to improve detection precision and recall by verifying that the shape of the foreground segmentation resembles the model mask.

The fact that S-DPM operates on HOG features provides flexibility in dealing with contour extraction measurement noise and local shape variance due to pose variation. However, S-DPM only outputs the detections of the object hypotheses rather than the exact location of the object contour. Even in the object hypothesis windows, the subset of edge pixels that correspond to the object silhouette is not apparent. In addition, contour-based object detection in cluttered scenes is susceptible to false detections caused by piecing together irrelevant contours.

To recover exact object contour pixel locations and reduce false positives, an additional shape matching step is required on top of the object hypotheses. Here, we propose using the collective shape of a subset of superpixels within each hypothesis region to verify the presence of an object.

For each detection hypothesis region, superpixels are computed directly from gPb Arbelaez et al. (2011). Searching over the entire space of superpixel subsets for the optimal match between the collective shape of the superpixels and the object model is combinatorial and impractical. Instead, we use a greedy algorithm to efficiently perform the search. In practice, with limited superpixels to select from, our greedy approach recovers the correct region with high probability. Figure 6.6 shows example results of shape verification. The greedy algorithm begins with a set of connected superpixels as a seed region and greedily searches over adjacent superpixels, picking the superpixel that yields the smallest χ^2 distance to the chordiogram of model silhouette. Intuitively, if we have a set of superpixels forming a large portion of the object with a few missing pieces, adding these pieces yields the best score. The initial seeds are formed by choosing all triplets of adjacent superpixels, and limiting examination to the top five seeds that yield the smallest

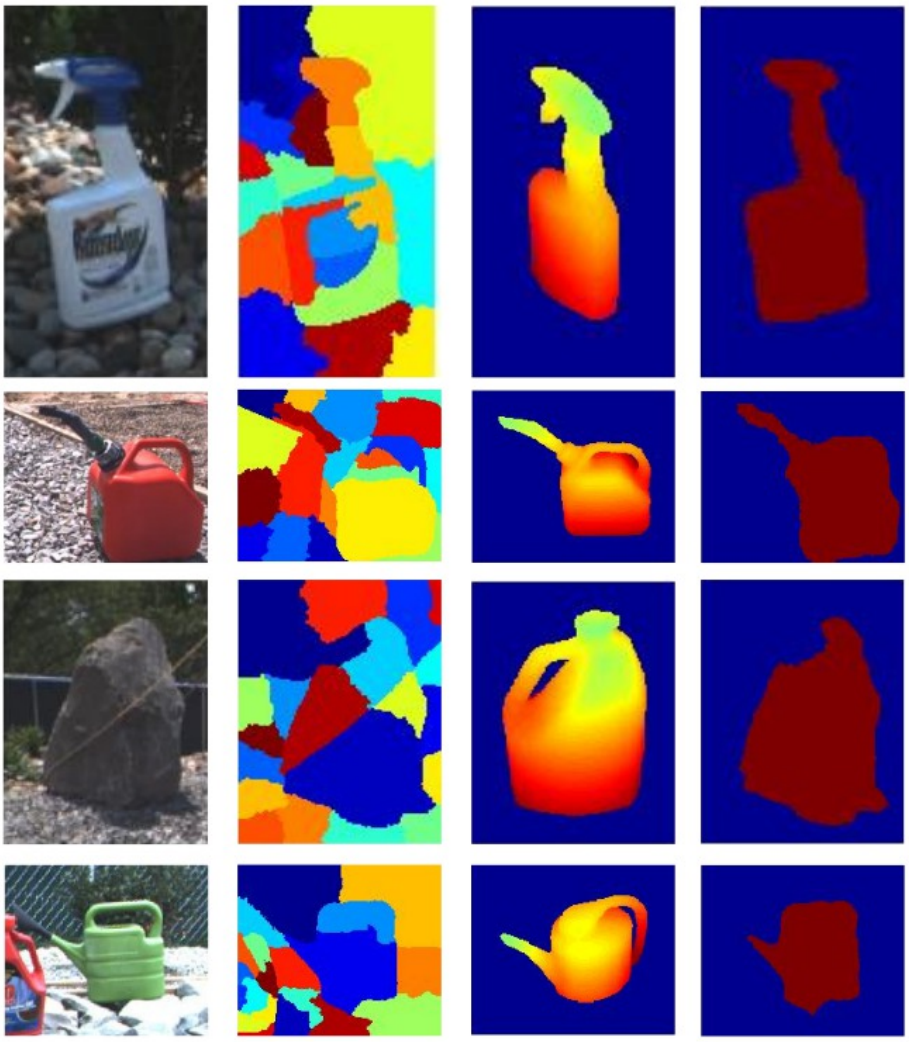


Figure 6.6: Shape descriptor-based verification examples. (left-to-right) Detection hypothesis window of the object, superpixel over-segmentation of the hypothesis region, visualization of the coarse object pose from the object detector and selected foreground mask.

χ^2 distance. The connectivity graph of superpixels is a planar graph with limited node degrees. The complexity of finding triplets in such a planar graph is $O(N \log N)$ in the number of nodes.

Once the correct foreground superpixels are selected, the detection bounding box is re-cropped to reflect the recovered foreground mask. Empirically, this cropping step yields a better localization of the detection result over the S-DPM, as measured in terms of precision and recall, see Sec. 6.4 Edges of the foreground mask are extracted and used in the subsequent processing stage for accurate 6-DoF *continuous* pose estimation.

6.3.6 Pose refinement

Robotic grasping requires an accurate estimate of an object's 3D pose. To improve upon the coarse pose estimate provided by the S-DPM, we perform a final iterative pose refinement step to recover the full *continuous* 6-DoF pose. This step is restricted to the region of the verified superpixel mask.

Our iterative refinement process consists of two steps: (i) determining the correspondence between the projected occluding boundary of the 3D model and the contour points along object segmentation mask, and (ii) estimating an optimal object pose based on the correspondences.

The contour correspondences are estimated using dynamic programming (DP) to ensure local matching smoothness. Given the initial (coarse) pose output from the object detection stage, the 3D object model is rendered to the image and its corresponding projected occluding boundary is extracted. Each point on the contour is represented by a descriptor capturing close-range shape information. The 31-dimensional contour descriptor includes the gradient orientation of a contour point (the central point) and the gradient orientations of the nearest 15 points on each side of the central point along the contour. The gradient orientation of the central point is subtracted from all elements of the descriptor, which gives in-plane rotation invariance. The matching cost between each pair is set to be the l_2 distance of the feature descriptor extracted at each point. DP is then used to

establish the correspondences between contour points.

To estimate the refined pose we use the motion field equation Horn (1986):

$$u(x,y) = \frac{1}{Z}(xt_z - t_x) + \omega_x(xy) - \omega_y(x^2+1) + \omega_z(y)$$
$$v(x,y) = \frac{1}{Z}(yt_z - t_y) - \omega_x(y^2+1) - \omega_y(xy) + \omega_z(x),$$

where $u(x,y), v(x,y)$ denote the horizontal and vertical components of the displacement vectors, respectively, between the model and matched image contour points, computed by DP, $Z(x,y)$ denotes the depth of the 3D model point for the current pose estimate and the Euler angles $(\omega_x, \omega_y, \omega_z)$ and 3D translation vector (t_x, t_y, t_z) denote the (locally) optimal motion of the object yielding the refined pose. The motion update of the current pose is recovered using least squares. This procedure is applied iteratively until convergence. In practice, we usually observe fast convergence with only three to five iterations. The running time of the pose refinement is about one second on an Intel 2.4GHz i7 CPU.

6.4 Experiments

Outdoor detection evaluation We introduce a challenging outdoor dataset for 3D object detection containing heavy background clutter. This dataset was collected from a moving robot and consists of eight sequences containing a total of 3403 test images; the dimensions of each image are 512×386. Figure 6.7 shows a set of representative imagery from the introduced dataset. The scenes contain a variety of terrains (e.g., grass, rock, sand, and wood) observed under various illumination conditions. The dataset represents the task of a robot navigating a complex environment and searching for objects of interest. The objects of interest are mostly comprised of textureless daily equipment, such as a watering pot, gas tank, watering can, spray bottle, dust pan, and liquid container. For each frame, 2D bounding boxes that tightly outline each object are provided. Further, the dataset includes the corresponding 3D model files used in our empirical evaluation.

On the outdoor dataset, we performed a shape-based object detection evaluation. We compared four methods, DOT Hinterstoisser et al. (2010), S-DPM with only the root

Figure 6.7: Representative images from the introduced outdoor dataset. The dataset was captured using a ground robot and includes diverse terrains, e.g., rocks, sand and grass, with illumination changes. Portions of the terrain are non-flat. Objects are scattered around the scene and typically do not occupy a major portion of the scene.

model, full S-DPM with root and parts, and the full S-DPM plus shape verification (proposed approach), on a detection task on the introduced dataset. Both DOT and S-DPM used the same training instances from Sec. 6.3.1 with a slight difference. For S-DPM, we trained one model component for each of 16 discrete poses. For DOT, we used the same quantization of the viewsphere but trained with 10 different depths ranging from close to far in the scene. During testing, S-DPM is run on different scales by building an image pyramid. The input to both methods were the same gPb thresholded images. In all our experiments, the threshold is set to 40 (gPb responses range between 0 and 255), where edges with responses below the threshold were suppressed. The default parameters of gPb were used. We did not observe a noticeable difference in the detection and pose estimate accuracy with varying the gPb parameter settings.

Table 6.3 shows a comparison of the average precision for detection on the outdoor dataset. The proposed approach consisting of the full S-DPM plus shape verification achieves the best mean average precision. It demonstrates that using shape verification improves detection due to the refinement of the bounding box to reflect the recovered silhouette. Full S-DPM outperforms both the root only S-DPM and DOT. This shows the benefit of the underlying flexibility in S-DPM.

Table top evaluation We evaluated our pose refinement approach under two settings. First, we recorded an indoor RGB-D dataset, with multiple objects on a table, from a head mounted Kinect on a PR2 robot. The RGB-D data is used as ground truth. We evaluated using three objects, watering can, gas tank, watering pot, placed at two different distances from the robot on the table and two different poses for each distance. For each scene, the target object was detected among all objects on the table and segmented using shape verification, and then the 6-DoF pose was estimated, as described in Sec. 6.3.6. The model point cloud was projected into the scene and Iterative Closest Point (ICP) Besl and McKay (1992) was performed between the model point cloud and the Kinect point cloud. We report ICP errors for both rotation and translation in Tables 6.1 and 6.2, resp. Errors in the rotations and translations are small for different angles and different depth.

		Estimated Rotation			Error		
		Roll	Pitch	Yaw	Roll	Pitch	Yaw
watering can	dist1	1.65	48.44	-145.37	0.99	3.57	-1.63
		5.50	50.73	-22.37	-3.20	-3.92	-0.07
	dist2	-4.33	41.93	48.78	-3.20	-3.92	-0.07
		2.44	49.60	-54.82	-0.12	1.95	-1.92
watering pot	dist1	-0.43	59.20	-73.00	-1.25	-0.28	1.36
		0.69	51.90	156.86	-1.82	-0.63	-3.48
	dist2	-10.43	66.93	38.28	-1.078	-6.67	-2.43
		-0.633	52.24	-131.94	-0.21	1.14	-0.88
gas tank	dist1	-0.15	50.58	-136.17	1.43	2.73	-4.58
		2.84	50.15	-51.15	-2.63	3.20	2.79
	dist2	-2.44	48.24	129.43	-3.57	0.02	-2.14
		-7.40	45.22	109.90	-1.55	-1.79	-1.03

Table 6.1: Estimated absolute rotation of the object and error in degrees.

Translation errors in the X and Y directions are smaller than in Z direction. Since Z is the depth direction, it is most affected by the 3D model acquisition and robot calibration. Both measurements show our method is robust and suitable for grasping task.

In addition, using the object pose estimated from our approach, we demonstrate with a PR2 robot successful detections and grasps of various objects from a cluttered table. In Fig. 6.8, we show qualitative results of the PR2 successfully grasping various objects on a cluttered table.

		Estimated Translation			Error		
		X	Y	Z	X	Y	Z
watering can	dist1	-46.5	-82.3	-1023.6	-1.14	-0.7	-2.8
		-57.1	-86.4	-1023.2	-1.2	2.8	-7.2
	dist2	-85.1	183.2	-1182.9	3.6	3.6	4.8
		-114.9	186.0	-1200.3	4.5	2.2	-5.1
watering pot	dist1	16.4	-154.0	-1020.9	2.8	1.0	0.06
		-117.6	-112.4	-1028.3	0.4	0.2	2.2
	dist2	-6.8	32.7	-1051.2	2.0	-2.9	-3.5
		-106.5	-6.6	-1053.1	-0.5	-0.2	-1.9
gas tank	dist1	-23.8	21.2	-1061.2	-1.8	-0.9	-3.2
		19.5	-116.0	-958.8	-0.4	1.7	-3.2
	dist2	-77.0	6.7	-1064.6	0.4	-0.9	-2.0
		-111.3	178.9	-1200.8	0.6	-0.4	-1.4

Table 6.2: Estimated absolute translation of the object and error in centimeters.

	watering pot	gas tank	watering can	spray bottle	dust pan	liquid container	average AP
S-DPM full+shape	0.686	0.645	0.523	0.515	0.429	0.506	**0.5507**
S-DPM full	0.688	0.610	0.547	0.507	0.387	0.509	0.5413
S-DPM root only	0.469	0.535	0.433	0.436	0.295	0.436	0.4340
DOT	0.407	0.412	0.340	0.089	0.111	0.188	0.2578

Table 6.3: Average precision on the introduced outdoor dataset.

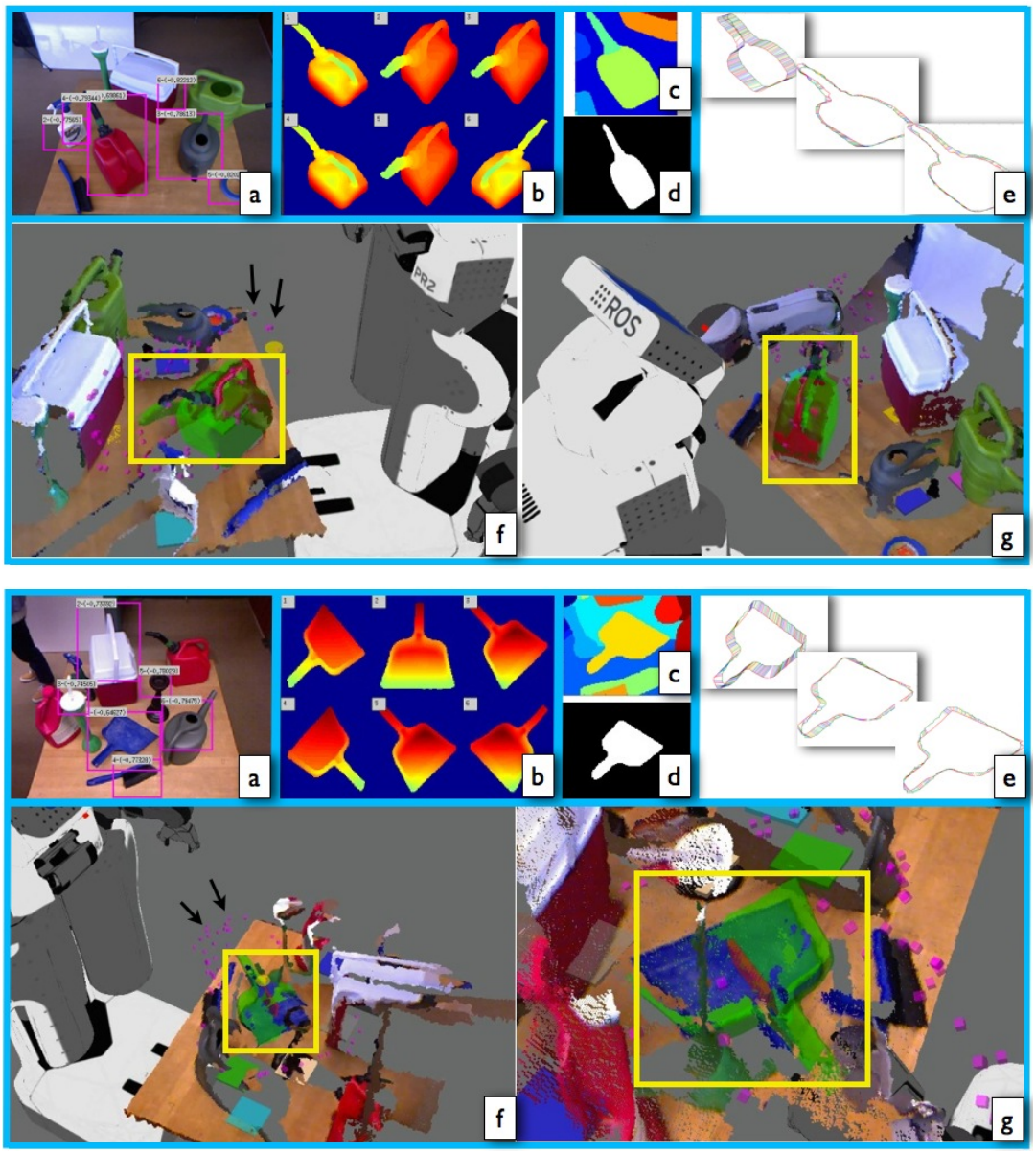


Figure 6.8: PR2 grasping process for two example input images. Top panel for gas tank and bottom for dust pan. a) S-DPM detection bounding boxes ordered by the detection score in decreasing order. b) Corresponding pose output from S-DPM for each detection. c) Segmentation of top scored detection window. d) Foreground mask selected by shape verification. e) Three iterations in pose refinement, alignments (shown in color) between curves are computed using DP. f) Visualization of PR2 model with the Kinect point cloud. Notice that the estimated model given in light green is well aligned with the point cloud. Grasping points are indicated by arrow. g) Another view of the same scene.

Chapter 7

3D Pose Estimation and Shape Reconstruction of Object Categories

7.1 Introduction

Recovering 3D geometry from 2D imaginary of an object is one of the most fundamental and challenging problems in computer vision. Geometric features were the main representation of objects in the 20th century and have long been used to establish correspondence between vertices and edges of a 3D model and their image projections Grimson (1990). Although such representation was successful with geometric invariance it could not cope with the complexity of appearance of 3D object categories in the real world which could only be learned from exemplars.

As soon as massive 2D image exemplars became available on the Internet and through tedious annotation, the computer vision community has harnessed fruitful results as the state of the art in detecting object categories has improved dramatically Felzenszwalb et al. (2010b); Girshick et al. (2014). More recently, researchers have focused on combining such approaches with 3D geometry to build more powerful object detectors that are also able to provide weak 3D information such as viewpoint Hu and Zhu (2014); Liebelt et al. (2008b); Pepik et al. (2012b,c); Zia et al. (2013). In this paper, we go beyond viewpoint

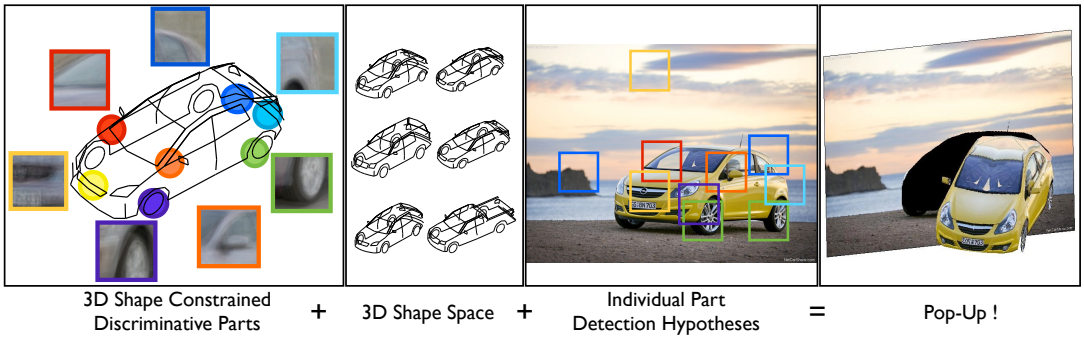


Figure 7.1: Illustrative summary of our approach: 3D Landmarks on a 3D model are associated with discriminatively learned part descriptors (left). Intra-class shape variation is captured with linear combinations of a sparse shape basis (2nd left). Learned part descriptors produce multiple maximum responses for each part in a testing image (3rd from left). The selection of the part hypotheses, 3D pose and 3D shape are simultaneously estimated and the result is illustrated through a popup (right).

estimation to establishing the actual 3D shape of an object for the sake of fine grained classification or 3D interaction such as grasping and manipulation. Very few efforts have been devoted to such combined estimation of pose and shape from a single image Hejrati and Ramanan (2012); Lin et al. (2014).

Recent advances in recognition have opened doors to better understanding of 3D in the wild, but there are three main challenges in the marriage of 2D appearance and 3D geometry: (1) how to learn a representation that captures appearance variation of geometric features across instances and poses, (2) how to establish the 3D shape of an object without exhaustively comparing to all possible instances or when that instance has not been seen before, and (3) how to optimize for appearance and correspondence compatibility as well as 3D shape and pose at the same time, without splitting the problem into subproblems.

In this paper, we propose a novel approach that marries the power of discriminative parts with an explicit 3D geometric representation with the goal to infer 3D shape and continuous pose of an object (or *pop-up*) from a single image. Part descriptors are discriminatively learned in training images. Such parts are centered around projections of 3D

landmarks which are given in abundance on the training 3D models. To establish a compact representation we minimize the number of needed landmarks by solving a facility-location problem. To deal with geometric deformation, we summarize the training set of 3D models into a shape dictionary from which we can generalize by linear combination. Given a test image we detect top location hypotheses of each part. The challenge is how to fit best these parts by maximizing the geometric consistency. This entails the selection among the hypotheses of each part and the shape/pose computation. Unlike other approaches which rely on local optimization and initialize pose by DPM-based discretized pose estimation Lin et al. (2014); Zia et al. (2013), we compute the selection as well as the shape and pose parameters in one step using a convex program solved with the alternating direction method of multipliers (ADMM).

Figure 7.1 illustrates the outline of our approach. In summary, the major technical contributions are:

- A convex optimization framework for joint landmark localization, fine grained 3D shape and continuous pose estimation from a single image.
- Our convex objective does not require viewpoint or detection initialization.
- An automatic landmark selection method considering both discriminative power in appearance and spatial coverage in geometry.

7.2 Related Work

The most related work includes the family of methods that estimate an object shape by aligning a deformable shape model to image features. This idea originated from the active shape model (ASM) (Cootes et al., 1995a), which was originally proposed for segmentation and tracking based on low-level image features. Cristinacce and Cootes (Cristinacce and Cootes, 2006) proposed the constrained local models (CLM), which combined ASM with local appearance models for 2D feature localization in face images. Gu and Kanade

(Gu and Kanade, 2006) presented a method to align 3D deformable models to 2D images for 3D face alignment. The similar methods were also proposed for 3D car modeling (Hejrati and Ramanan, 2012; Hu and Zhu, 2014; Lin et al., 2014; Zia et al., 2013) and human pose estimation (Ramakrishna et al., 2012; Zhou and De la Torre, 2014). Our method differs in that we use a data-driven approach for discriminative landmark selection and we solve landmark localization and shape reconstruction in a single convex framework, which enables a global solution.

The representation of our model is inspired by recent advances in part-based modeling (Felzenszwalb et al., 2010b; Hariharan et al., 2012; Kokkinos, 2011; Singh et al., 2012), which models the appearance of object classes with a collection of mid-sized discriminative parts. Our optimization approach is related to the previous work on using convex relaxation techniques for object matching (Jiang et al., 2011; Li et al., 2011; Maciel and Costeira, 2003). These methods focused on finding the point-to-point correspondence between an object template and an image in 2D, while our method considers 3D to 2D matching as well as shape variability.

Our paper is also related to recent work on 3D pose estimation which encodes the geometric relations among local parts and achieved continuous pose estimation. Several work leveraged 3D models to warp features or parts into their canonical view (Savarese and Li, 2007; Xiang and Savarese, 2012; Yan et al., 2007). Other work rendered local appearances and depth from 3D models and subsequently encoded in a 3D voting scheme (Glasner et al., 2011b; Liebelt et al., 2008b; Sun et al., 2010). DPM was further lifted to 3D deformable models (Fidler et al., 2012; Pepik et al., 2012b) to predict continuous viewpoint. Instance models were also used to recover 3D pose of an object (Aubry et al., 2014; Lim et al., 2013). But this line of work focused on pose estimation and either used generic class models or instance-based models. Our approach differs in that we not only provide a detailed shape representation but also consider intra-class variability.

7.3 Shape Constrained Discriminative Parts

Our proposed method models both 2D appearance variation and 3D shape deformation of an object class. The 2D appearance is modeled as a collection of discriminatively trained parts. Each part is associated with a 3D landmark point on a deformable 3D shape.

Unlike the previous works that manually define landmarks on the shape model, we propose an *automatic* selection scheme: we first learn the appearance models for all points on the 3D model, evaluate their detection performance, and select a subset of them as our part models based on their detection performance in 2D and the spatial coverage in 3D.

7.3.1 Learning Discriminative Parts

One of the main challenges in object pose estimation rises from the fact that due to perspective transform and self occlusions, even the same 3D position of an object has very different 2D appearances in the image observed from different viewpoints. We tackle this problem by learning a mixture of discriminative part models for each point in the 3D model to capture the variety in appearance. Each part detector consists of a simple but fast HOG detector Dalal and Triggs (2005b) and a more sophiscated but slow deep classifiers trained with deep Convolutional Neural Net (CNN) Krizhevsky et al. (2012). The HOG detectors provide location proposals to deep classifiers. Such design is chosen to balance speed versus accuracy.

Given a training set D, each training image $I_i \in D$ is associated with the 3D points of the object shape $S \in \mathbb{R}^{3\times p}$, their 2D projections $L_i \in \mathbb{R}^{2\times p}$ annotated in the image.

HOG Part Detectors We bootstrap the learning of a discriminative mixture model for each part via clustering whitened HOG (WHO) features Hariharan et al. (2012); Singh et al. (2012). Denote $\phi(L_{ij})$ as the HOG feature of the positive image patch centered at L_{ij} and $\overline{\phi}_{\mathbf{bg}}$ as the mean of background HOG features. We compute the WHO feature as $\Sigma^{-1/2}(\phi(L_{ij}) - \overline{\phi}_{\mathbf{bg}})$, where Σ is the shared covariance matrix computed from all positive

and negative features. Then we cluster the WHO features of each part j into m clusters using K-means.

A linear classifier W_{cj} is trained for each cluster c of a part j. We apply linear discriminant analysis due to efficiency in training and limited loss in detection accuracy Girshick and Malik (2013); Hariharan et al. (2012),

$$W_{cj} = \Sigma^{-1} \left(\overline{\phi}(L_{ij}; z_{ij} = c) - \overline{\phi}_{\mathbf{bg}} \right), \tag{7.1}$$

where $z_{ij} \in \{1, \ldots, m\}$ is the cluster assignment for each feature, and $\overline{\phi}(L_{ij}; z_{ij} = c)$ is the mean feature over all L_{ij} of cluster c. Let $\mathbf{x} = (x, y)$ be the position (x, y) in the image. The response of part j at a given location $\mathbf{x}$ is the max response over all its c components: $score_j(\mathbf{x}) = \max_c \{W_{cj} \cdot \phi(\mathbf{x})\}$.

We introduce a latent variable for each training patch, $r_{ij} \in \mathbb{R}^2$ to represent the relative center location to the annotated landmark location L_{ij}. We improve the classifiers learned from (7.1) by repositioning the patch center in the neighborhood $\Delta(L_{ij})$ of L_{ij} and retrain the classifiers. Note that the latent update procedure is similar to that of DPM Felzenszwalb et al. (2010b) with the difference that we do not apply generalized distance transform to filter responses but only consider maximum responses within a local region. The reason is that our model, as will be discussed in Section 7.3.3, is constrained by the 3D shape space instead of learned 2D deformations. We want, thus, to obtain accurate part localization to estimate the object pose and shape. A 2×2 covariance matrix D_j is estimated for each landmark j from latent variables r_{ij}, to model the uncertainty of the detected landmark position $\mathbf{x}^*_{ij}$ relative to the ground truth.

Deep Part Classifiers HOG part detections serve as part proposals and are subsequently re-ranked by forwarding through a CNN and applying SVM on the extracted Pool5 layer features. During training, Pool5 features were extracted for both positive and negative patches and an SVM is trained for each part mixture. During our experiments, we observed that 1) fine-tuning from pre-trained AlexNet Krizhevsky et al. (2012) with part patches of the same object category improves part detection accuracy, 2) Pool5 has better

Method	HOG-SVM	CNN
mAP	0.41	**0.53**

Table 7.1: Comparison of CNN and HOG-SVM in part localization. Mean average precision (mAP) of localizing the 12 parts of PASCAL3D car category are shown.

performance than fully connected layers (fc6, fc7) for mid-level patches, 3) training separate classifiers for each part mixture component outperform a combined classifier. We used publicly available deep learning toolbox Caffe Jia et al. (2014) in our experiments.

The performance of deep part classifiers is evaluated by comparing against SVM trained HOG filters (HOG-SVM) with hard negative mining. Localization accuracy is measured by the average precision of detecting the part within the close vicinity of the groundtruth location. Table 7.1 shows performance comparison of CNN and HOG-SVM on the 12 parts of PASCAL3D dataset car category.

7.3.2 Selecting Discriminative Landmarks

Seeking a compact representation of the object, we try to select only a small subset of discriminative landmarks S_D among all 3D landmarks S. We want the selected landmarks S_D to be both associated with discriminative part models and have a good spatial coverage of the object shape model in 3D. The selection problem is formulated as a **facility location problem**,

$$\begin{aligned} \min_{y_u, x_{uv}} \quad & \sum_u z_u y_u + \lambda \sum_{uv} d_{uv} x_{uv}, \\ \text{s.t.} \quad & \sum_v x_{uv} = 1, \\ & x_{uv} \leq y_v, \qquad \forall u, v, \\ & x_{uv}, y_u \in \{0, 1\}, \qquad \forall u, v, \end{aligned} \tag{7.2}$$

where the interpretations of each symbol are presented in Table 7.2.

Symbol	Interpretation
z_u	cost of selecting landmark u
y_u	binary landmark selection variable
d_{uv}	cost of landmark v "serving" u
x_{uv}	binary variable for landmark v "serving" u
λ	trade off between unary costs and binary costs

Table 7.2: Notations interpretation in (7.2)

The cost z_u for a landmark u should be lower if the associated part model is more discriminative. We model the discriminativeness by evaluating the Average Precision (AP) of detecting each landmark in the training set. For any landmark u, we perform detection with the learned part model in the training set S to generate a list of location hypotheses H_u. A hypothesis $h \in H_u$ is considered as true positive if the ground truth location L_{iu} is within a small radius δ. Let the computed AP for a part u be AP_u, we set $z_u = 1 - AP_u$. The costs of "serving" (or suppressing) other landmarks are set to be the euclidean distance between landmarks in 3D, i.e., $d_{uv} = ||S_u - S_v||_2$. The value of λ is set to 1 in our experiments. The minimization problem 7.2 is a Mixed Integer Programming (MIP) problem, which is known to be NP-hard. But a good approximation solution can be obtained by relaxing the integer constrains to be $x_{uv} \in [0, 1],\ y_u \in [0, 1]$, solving the relaxed Linear Programming problem, and thresholding the solution. Figure 7.2 visualizes an example result of MIP optimization for landmark selection.

7.3.3 3D Shape Model

We start our description by explaining how we would estimate the shape of an object if 2D part - 3D landmark correspondences were known. We represent a 3D object model as a linear combination of a few basis shapes to constrain the shape variability. This assumption has been widely used in various shape-related problems such as object segmentation

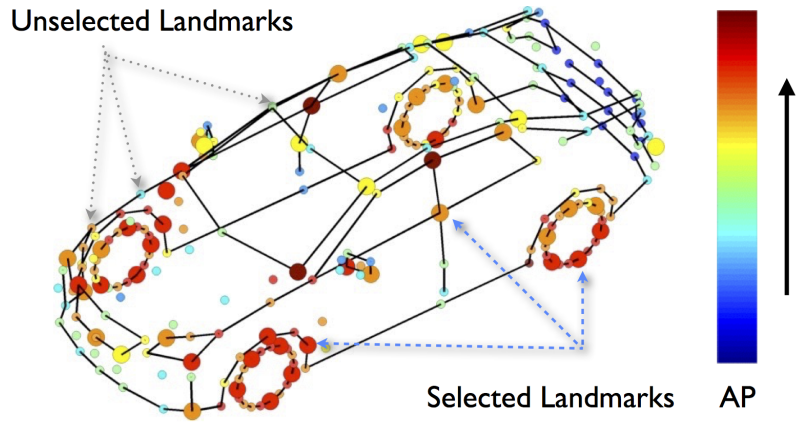


Figure 7.2: Visualization of the landmark selection optimization result. All 256 landmark points of a car are shown in circle markers. The color of the markers represents the Average Precision(AP) of the landmark part detection on the training set, red means higher AP and blue means lower AP. The size of the landmark represents the selection result, the larger ones are selected via the MIP optimization and the smaller ones are not selected. The red landmarks are preferred since they have higher detection accuracy, but only a subset of red landmarks are selected because they are close in 3D.

Cootes et al. (1995a), nonrigid structure from motion Bregler et al. (2000) and single image-based shape recovery Gu and Kanade (2006); Zia et al. (2013). We use a weak-perspective model, which is a good approximation when the depth of the object is smaller than the distance from the camera. With these two assumptions, the 2D shape $P \in \mathbb{R}^{2\times p}$ can be described by

$$P = R\sum_{i=1}^{k} c_i B_i + \mathbf{t}\mathbf{1}^T, \tag{7.3}$$

where $B_i \in \mathbb{R}^{3\times p}$ denotes the i-th basis shape, $R \in \mathbb{R}^{2\times 3}$ represents the first two rows of camera rotation, and $\mathbf{t} \in \mathbb{R}^2$ is the translation vector. In model inference, the reprojection error is minimized to find the optimal parameters.

However, the model in (7.3) is bilinear in R and c_is yielding a nonconvex problem. In order to have a linear representation, we use the method proposed in Zhou et al. (2015a), which assumes that the unknown shape is a linear combination of scalable and rotatable basis shapes:

$$P = \sum_{i=1}^{k} T_i B_i + \mathbf{t}\mathbf{1}^T, \tag{7.4}$$

where $T_i \in \mathbb{R}^{2\times 3}$ denotes the first two rows of a similarity transformation matrix. In order to enforce T_i to be orthogonal, the spectral norms of T_is are minimized during model inference. The spectral norm is the largest singular value of a matrix, and minimizing it enforces the two singular values of T_i to be equal, which yields an orthogonal matrix Zhou et al. (2015a). After T_is are estimated, the third rows of T_is can be recovered from the orthogonality and then the estimated 2D shape can be lifted to 3D.

7.4 Model Inference

Finally, we obtain global geometry-constrained local-part models, in which the unknowns are the 2D part locations as well as the 3D pose and shape. In model inference, we maximize the detector responses over the part locations while minimizing the geometric reprojection error.

7.4.1 Objective Function

We try to locate a part by finding its correspondence in a set of hypotheses given by the trained detector. The cost without geometric constraints is

$$f_{score}(\mathbf{x}_1, \cdots, \mathbf{x}_p) = -\sum_{j=1}^{p} \mathbf{r}_j^T \mathbf{x}_j, \tag{7.5}$$

where $\mathbf{x}_j \in \{0,1\}^l$ is the selection vector and $\mathbf{r}_j \in \mathbb{R}^l$ is the vector of the detection scores for all hypotheses for the j-th part.

Geometric consistency is imposed by minimizing the following reprojection error:

$$\begin{aligned} &f_{geom}(\mathbf{x}_1, \cdots, \mathbf{x}_p, T_1, \cdots, T_k, \mathbf{t}) = \\ &\frac{1}{2} \sum_{j=1}^{p} \left\| D_j^{-\frac{1}{2}} \left(L_j^T \mathbf{x}_j - \left[\sum_{i=1}^{k} T_i B_i \right]_j - \mathbf{t} \right) \right\|^2, \end{aligned} \tag{7.6}$$

where we concatenate the 2D locations of hypotheses for part j in $L_j \in \mathbb{R}^{l \times 2}$ and denote the covariance estimated in training as D_j.

As introduced in Section 7.3.3, we add the following regularizer to enforce the orthogonality of T_i:

$$f_{reg}(T_1, \cdots, T_k) = \sum_{i=1}^{k} \|T_i\|_2, \tag{7.7}$$

where we use $\|T_i\|_2$ to represent the spectral norm of T_i, i.e., the largest singular value.

To simplify the computation, we relax the binary constraint on $\mathbf{x}_i$ and allow it to be a soft-assignment vector $\mathbf{x}_i \in \mathcal{A}$, where $\mathcal{A} = \{\mathbf{x} \in [0,1]^l | \ \sum_{i=1}^{l} x_i = 1.\}$.

Finally, the objective function reads

$$\begin{aligned} &\min_{\overline{X}, \overline{T}, \mathbf{t}} \ f_{geom}(\overline{X}, \overline{T}, \mathbf{t}) + \lambda_1 f_{score}(\overline{X}) + \lambda_2 f_{reg}(\overline{T}), \\ &\text{s.t.} \quad \mathbf{x}_j \in \mathcal{A}, \ \forall j = 1:p, \end{aligned} \tag{7.8}$$

where $\overline{X}$ and $\overline{T}$ represent the unions of $\mathbf{x}_1, \cdots, \mathbf{x}_p$ and $T_1, \cdots, T_k$, respectively. After solving (7.8), we recover the 3D shape S and pose $\theta = (R, \mathbf{t})$ from T_is, as introduced in Section 7.3.3.

$$\mathbf{t} \leftarrow \arg\min_{\mathbf{t}} \mathcal{L}, \tag{7.11}$$

$$\overline{X} \leftarrow \arg\min_{\overline{X}} \mathcal{L}, \tag{7.12}$$

$$\overline{T} \leftarrow \arg\min_{\overline{T}} \mathcal{L}, \tag{7.13}$$

$$\overline{Z} \leftarrow \arg\min_{\overline{Z}} \mathcal{L}, \tag{7.14}$$

$$Y \leftarrow \rho(\overline{T} - \overline{Z}). \tag{7.15}$$

7.4.2 Optimization

The problem in (7.8) is convex since f_{score} is a linear term, f_{geom} is the sum of squares of linear terms, and f_{reg} is the sum of norms of unknown variables. We use the alternating direction method of multipliers (ADMM) Boyd (2010) to solve the convex problem in (7.8). Since f_{reg} is nondifferentiable, which is not straightforward to optimize, we introduce an auxiliary variable Z and reformulate the problem as follows:

$$\begin{aligned} &\min_{\overline{X},\overline{T},\mathbf{t},\overline{Z}} \; f_{geom}(\overline{X},\overline{T},\mathbf{t}) + \lambda_1 f_{score}(\overline{X}) + \lambda_2 f_{reg}(\overline{Z}), \\ &\text{s.t.} \quad \overline{T} = \overline{Z}, \quad \mathbf{x}_j \in \mathcal{A}, \; \forall j = 1:p. \end{aligned} \tag{7.9}$$

The corresponding augmented Lagrangian is:

$$\begin{aligned} \mathcal{L} = {} & f_{geom}(\overline{X},\overline{T},\mathbf{t}) + \lambda_1 f_{score}(\overline{X}) + \lambda_2 f_{reg}(\overline{Z}) \\ & + \langle Y, \overline{T} - \overline{Z} \rangle + \frac{\rho}{2} \|\overline{T} - \overline{Z}\|_F^2. \end{aligned} \tag{7.10}$$

The ADMM algorithm iteratively updates variables by the following steps to find the stationary point of (7.10):

It can be shown that (7.11), (7.12) and (7.13) are all quadratical programming problems, which have closed-form solutions or can be solved efficiently using existing convex solvers. (7.14) is a spectral-norm regularized proximal problem, which also admits a closed-form solution Zhou et al. (2015a).

7.4.3 Visibility Estimation

In model inference, only visible landmarks should be considered. To estimate the unknown visibility, we adopt the following strategy. We first assume that all landmarks are visible and solve our model in (7.8) to obtain a rough estimate of the viewpoint. Since the landmark visibility of a car only depends on the aspect graph, the roughly estimated viewpoint can give us a good estimate of the landmark visibility. We observed that our model could reliably estimate the coarse view by assuming the full visibility, which might be attributed to the global optimization. After obtaining the visibility, we solve our model again by only considering the visible landmarks. The full shape can be reconstructed by the linear combination of full meshes of basis shapes after the coefficients are estimated.

7.4.4 Successive Refinement

The relaxation of binary selection vectors $\mathbf{x}_j$s in (7.8) may yield inaccurate localization, since it allows the landmark to be located inside the convex hull of the hypotheses. To improve the precision, we apply the following scheme: we solve our model in (7.8) repeatedly, and in each iteration we define a trust region based on the previous result for each landmark and merely keep the hypotheses inside the trust region as the input to fit the model again. We use three iterations. We can start from a large trust region to achieve global fitting and gradually decrease the trust region size in each iteration to reject outliers and improve localization. This successive refinement scheme has been widely-used for feature matching Jiang et al. (2011); Li et al. (2011).

7.5 Experiments

In this section, we evaluate our method (PopUp) in terms of both shape and pose estimation accuracy. The experiments are carried out on the Fine Grained 3D Car dataset (FG3DCar) Lin et al. (2014) and PASCAL3D Xiang et al. (2014). Both datasets have landmark locations in the image and pose annotation for 3D objects.

7.5.1 FG3D Car Dataset

FG3DCar dataset consists of 300 images with 30 different car models of 6 car types under different viewing angles. Each car instance is associated a shape model of 256 3D landmark points and their projected 2D locations annotated in the image as well as 3D pose annotation. We perform the following evaluations: First, we compare the accuracy of pose and shape estimation to the iterative model fitting method of Lin et al. (2014) (FG3D) in terms of 2D landmark projection error. Second, we compare the coarse viewpoint estimation error to viewpoint-DPM (VDPM) Gu and Ren (2010b); Xiang et al. (2014). In addition, since our viewpoint estimation is continuous, we also show the angular errors comparing to the groundtruth annotation. Through out the experiments, we follow the same training-testing split as Lin et al. (2014).

We learn a mixture of discriminative part models of three components for each of 256 landmark points as described in Section 7.3. The Average Precision (AP) of the landmark detection is evaluated on the training set. We count a detection as true positive only if the detected location is close to the annotated location. We optimize the landmark selection with unary cost as 1 - AP of each landmark and pairwise cost as the average pairwise 3D distance over all the 3D models. 52 out of 256 landmark points are selected with MIP optimization while FG3DCar provides 62 manually selected landmark. To build the shape models, we learned a dictionary consisting of 10 basis shapes from the 3D models provided in the FG3DCar dataset.

Note that, unlike FG3D, our method does not need an external object detector to initialize either the location and scale in the image or coarse landmark locations. We perform pose and shape estimation on the original image with background clutter.

3D Shape Estimation 3D Shape estimation accuracy is evaluated in terms of meanAPD which is the average landmark projection error in pixels over the landmarks and the test instances. In the following experiments, we investigate the effect of using different 3D shapes on the model fitting error. We compare three setups with different basis shapes:

Method		meanAPD (SL)	meanAPD
PopUp	Mean shape	16.5	20.6
PopUp	Class mean	15.4	18.9
PopUp	Shape space	**14.6**	**17.7**
FG3D	Class mean	-	18.1
FG3D	Shape space	-	20.3

Table 7.3: Model fitting error of PopUp versus FG3D in terms of mean APD in pixels evaluated on 52 selected landmarks (SL) and 64 landmarks provided in the dataset.

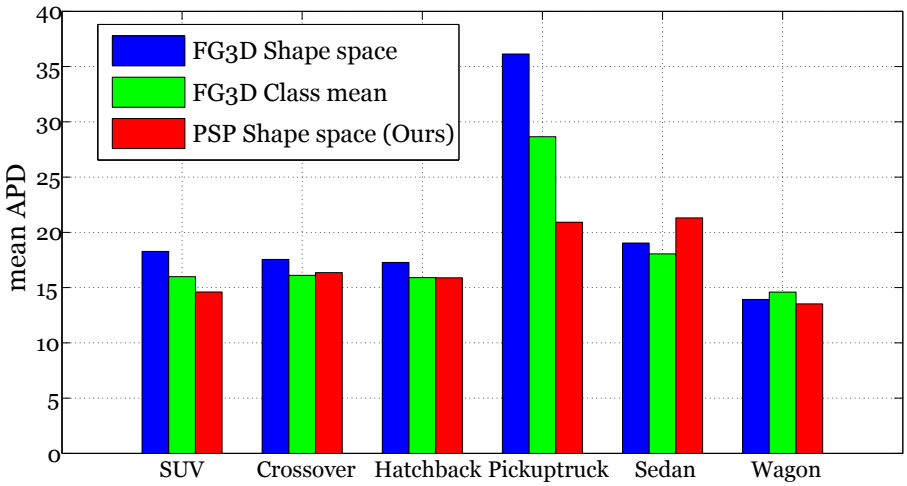


Figure 7.3: Car type specific meanAPD of PopUp versus FG3D with mean prior and class prior. Comparing to the FG3D method, our method achieves lower meanAPD on most car types. For the type of pickup truck, our method significantly outperforms FG3D.

	Accuracy	
Method	40° per view	20° per view
VDPM	82.7%	71.3%
PopUp	89.3%	84.7%

Table 7.4: Coarse viewpoint estimation accuracy versus VDPM evaluated on the FG3DCar dataset. Accuracies are compared with two discretization schemes, 20 degrees per coarse viewpoint and 40 degrees per coarse viewpoint.

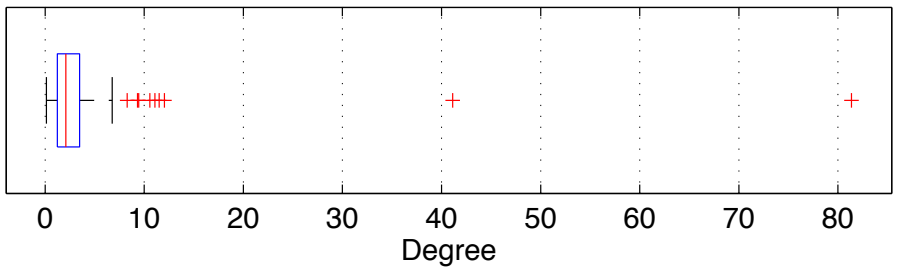


Figure 7.4: Continuous viewpoint (azimuth) error comparing to the groundtruth on all 150 test images in the FG3DCar dataset. The mean error is 3.4 in degrees.

only the mean shape, class-mean shapes and the learned shape space (10 basis shapes). The middle column of Table 7.3 shows the fitting error on selected discriminative landmarks. The fitting error decreases when we use the shape space instead of the mean shape or the class mean, which validates the use of shape space to express intra-class shape variation.

Since the selected discriminative landmarks are not identical to the landmarks provided in the FG3DCar dataset, we also compare the meanAPD on the landmarks provided in the dataset. Our method outperforms FG3D using the shape space without knowing the class type. Note that, their detectors are trained on the manually selected 64 landmarks provided in the dataset while our detectors are trained on the 52 automatic selected discriminative landmarks.

Although our objective is to optimize the projection error on the discriminative landmarks, the fitting error on the dataset provided landmarks is also minimized. This shows the effectiveness of the landmark selection process. The error is reported on the same scale as FG3D. Figure 7.3 shows the per class 3D model fitting error. Our method outperforms FG3D on most class types with particular success on the pickup trucks.

Viewpoint Estimation We compare PopUp to VDPM in discrete viewpoint estimation accuracy. For VDPM we train two sets of baseline VDPM with coarse viewpoints (azimuth) of every 20 degrees and every 40 degrees for each view. Each component of VDPM corresponds to a viewpoint label. During inference, the viewpoint of the test car instance is predicted as the training viewpoint of the max scoring component. For PopUp, the estimated continuous viewpoint is discretized in the same way as VDPM. Table 7.4 shows the comparison of the two methods. In both two cases, PopUp outperforms VDPM. We further analyze the estimation error of PopUp by looking at continuous viewpoint estimation error and show that the majority error is introduced by discretization. We compare our estimation to the ground-truth viewpoint (azimuth) and report the absolute angular value in Figure 7.4. The mean error over the whole test set is only 3.4 in degree.

Number of basis	10	6	3
meanAPD	**17.7**	18.4	19.1

Table 7.5: Model fitting error of PopUp with different different number of shape basis in terms of mean APD in pixels evaluated on 64 landmarks provided in the dataset.

Selection Method	Optimized	Greedy	Uniform
meanAPD	**17.7**	20.1	22.9

Table 7.6: Model fitting error of PopUp with different landmark selection methods in terms of mean APD in pixels evaluated on 64 landmarks provided in the dataset.

In addition to the quantitative evaluations, we show qualitative results on the test images from FG3DCar in Figure 7.6, where we project the 3D model wireframe with the estimated pose and shape on to the image. We also show the textured model rendered at novel views.

7.5.2 Sensitivity Analysis

In order to gain a better insight to the modeling and the optimization, we investigate the model sensitivity under different conditions. First, we model sensitivity with different number of shape basis. Table 7.5 shows the meanAPD using different number of basis, 10, 6 and 3. In the case of 6 basis, all car types have at least one corresponding basis. In the case of 3 basis, we only kept sedan, SUV and pickup truck. The results show robustness to the number of shape basis. Second, we compare the 2D landmark localization error related to different landmark sampling method. Table 7.6 shows the meanAPD by using optimized selection in Section 7.3.2, greedy selection of the most discriminative landmarks and uniform sampling of the landmarks. The optimized selection outforms both greedy selection and uniform sampling.

Method	Views	bicycle	bus	car	mbike
PopUp (ours)	4	42.6	49.3	29.8	39.9
	8	33.2	36.7	27.4	24.4
	16	16.9	40.7	21.4	16.6
	24	13.0	31.5	16.0	11.3
VDPM Xiang et al. (2014)	4	41.7	26.1	20.2	30.4
	8	36.5	35.5	23.5	25.1
	16	18.4	46.9	18.1	16.1
	24	14.3	39.2	13.7	10.1
Ghodrati et al. Ghodrati et al. (2014)	4	34.4	50.7	28.9	29.4
	8	27.6	50.3	26.6	24.7
	16	18.0	42.9	19.6	15.9
	24	12.6	40.2	15.9	13.2
DPM-3D Pepik et al. (2012c)	4	43.9	50.7	36.9	31.8
	8	40.3	50.3	36.6	32.0
	16	22.9	42.9	29.6	16.7
	24	16.7	42.1	24.6	10.5

Table 7.7: Average (discrete) Viewpoint Accuracy on four categories of PASCAL3D dataset.

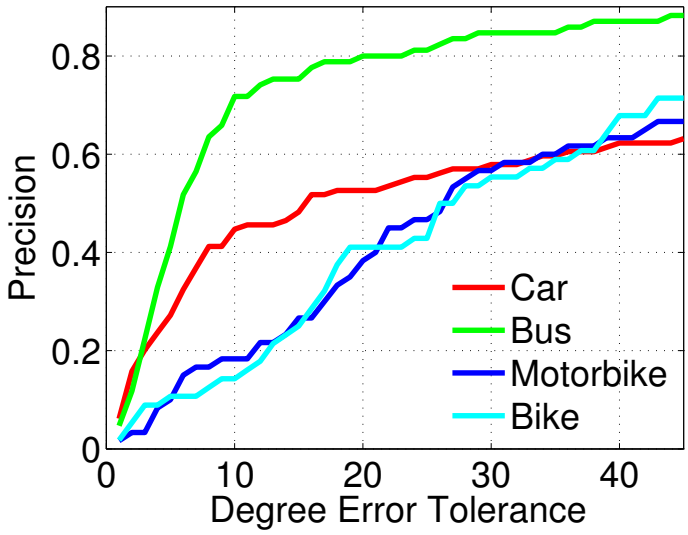


Figure 7.5: Precision recall curves for continuous viewpoint estimation on four categories of PASCAL3D, occluded instances are excluded. The horizontal axis is the tolerance of viewpoint error to count a prediction as correct, in the range of $[0, \pi/4]$. The vertical axis shows the precision.

7.5.3 PASCAL3D Dataset

The PASCAL3D dataset augments a subset of PASCAL dataset Everingham et al. (2010) with 3D models and pose annotations. PASCAL3D consists of images captured under various natural conditions. Occlusions and various object sizes cast great challenges to 3D estimation. We validate our method on four categories of PASCAL3D (test set): bicycle, bus, car and motorbike. Both discrete and continuous viewpoint accuracies are evaluated. Part Detectors are trained on the PASCAL3D training set. We use the provided landmarks and 3D models.

For discrete viewpoint accuracy, we compare Average Viewpoint Accuracy to recently reported state-of-art results on the benchmark Ghodrati et al. (2014); Pepik et al. (2012c); Xiang et al. (2014). We use VDPM as base detector and estimate viewpoint within each

detection hypothesis and quantize our continuous viewpoint output into discrete bins. Table 7.7 shows the accuracy of viewpoint prediction with different quantization of the azimuth angle, namely 4, 8, 16 and 24 views. Our results are comparable on different categories. While our model-based method performs well on larger objects, statistical learning based approaches as Pepik et al. (2012c) have advantages on small and heavily occluded instances in terms of viewpoint prediction.

We evaluated the continuous viewpoint accuracy on non-occluded instances within groundtruth bounding boxes. Figure 7.5 shows the precision-recall curves for four categories as the viewpoint error tolerance changes within $[0, \pi/4]$. We can observe that for bus and car the precision increases quickly as the angular tolerance increases from 0 to 10 degrees, meaning that the majority angular errors are less than 10 degrees. Bus and car outperform bicycle and motorbike with our method because their landmarks have larger appearance variation. Figure 7.7 shows qualitative results with estimated visible landmarks reprojected.

We break down the running time of our system on a 3.3Ghz Intel i7 CPU and an Nvidia TitanZ GPU as the following. Estimating a single object instance in a PASCAL3D image (500x300 pixels) requires: 0.08 seconds building HOG pyramid; 1.41 seconds in filters convolution; 3.76 seconds in CNN classification and 1.52 seconds in ADMM.

Figure 7.6: Example 3D estimation results from FG3DCar are shown. In the first two rows, the 3D wire frame of the car model is projected on the image with estimated pose and shape. Red solid lines represent visible wire frames and blue dotted lines represent invisible wire frames. In the last row, the textured 3D reconstructions of the cars in the second row are rendered at novel viewpoints. (We use symmetry to texture the invisible faces).

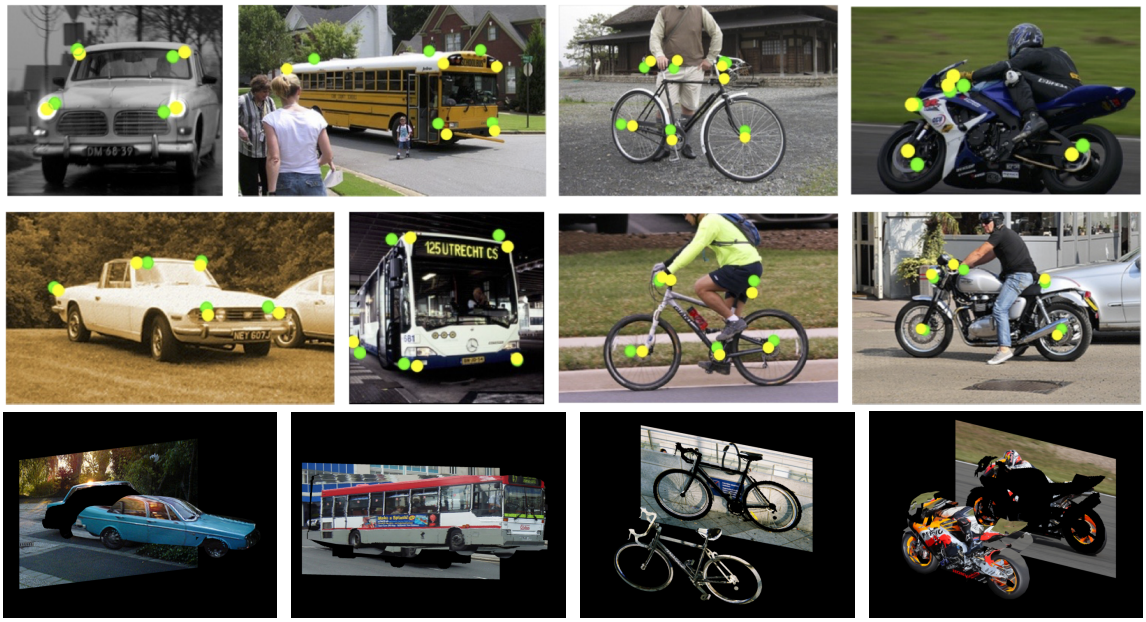


Figure 7.7: Examples of landmark localization results from different categories of PASCAL3D are shown in the first two rows. Visible 3D landmarks are projected back to the image. The yellow dots are groundtruth locations and the green dots are the estimation. The last row shows example pop-up results of different object classes.

Chapter 8

Articulated 3D Pose Estimation from Image Sequences

8.1 Introduction

This paper is concerned with the challenge of recovering the 3D full-body human pose from a monocular RGB image sequence. Potential applications of the presented research include human-computer interaction (cf. Shotton et al. (2011)), surveillance, video browsing and indexing, and virtual reality.

From a geometric perspective, 3D articulated pose recovery is inherently ambiguous from monocular imagery Lee and Chen (1985). Further difficulties are raised due to the large variation in human appearance (e.g., clothing, body shape, and illumination), arbitrary camera viewpoint, and obstructed visibility due to external entities and self-occlusions. Notable successes in pose estimation consider the challenge of 2D pose recovery using discriminatively trained 2D part models coupled with 2D deformation priors, e.g., Andriluka et al. (2014); Xiaohan Nie et al. (2015); Yang and Ramanan (2011), and more recently using deep learning, e.g., Toshev and Szegedy (2014). Here, the 3D pose geometry is not leveraged. Combining robust image-driven 2D part detectors, expressive 3D geometric pose priors and temporal models to aggregate information over

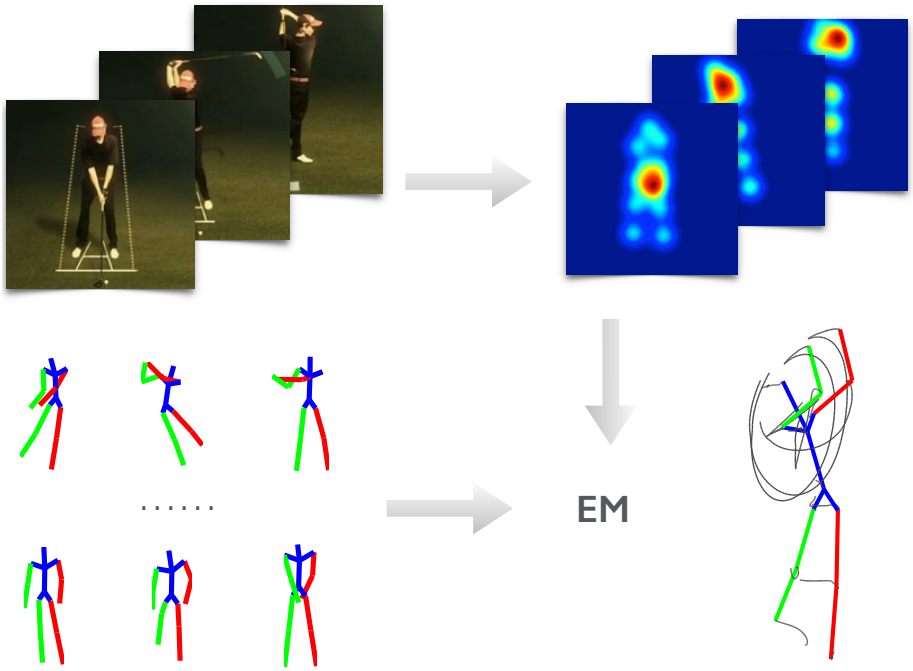


Figure 8.1: Overview of the proposed approach. (top-left) Input image sequence, (top-right) CNN-based heat map outputs representing the soft localization of 2D joints, (bottom-left) 3D pose dictionary, and (bottom-right) the recovered 3D pose sequence reconstruction.

time is a promising area of research that has been given limited attention, e.g., Andriluka et al. (2010); Zhou and la Torre (2014). The challenge posed is how to seamlessly integrate 2D, 3D and temporal information to fully account for the model and measurement uncertainties.

This paper presents a 3D pose recovery framework that consists of a novel synthesis between discriminative image-based and 3D reconstruction approaches. In particular, the approach reasons jointly about image-based 2D part location estimates and model-based 3D pose reconstruction, so that they can benefit from each other. Further, to improve the approach's robustness against detector error, occlusion, and reconstruction ambiguity, temporal smoothness is imposed on the 3D pose and viewpoint parameters. Figure 8.1 provides an overview of the proposed approach. Given the input video (Fig. 8.1, top-left),

2D joint heat maps are generated with a deep convolutional neural network (CNN) (Fig. 8.1, top-right). These heat maps are combined with a sparse model of 3D human pose (Fig. 8.1, bottom-left) within an Expectation-Maximization (EM) framework to recover the 3D pose sequence (Fig. 8.1, bottom-right).

8.1.1 Related work

Considerable research has addressed the challenge of 3D human motion capture from video (Brubaker et al., 2010; Moeslund et al., 2006; Sminchisescu, 2007). Early research on 3D monocular pose estimation in videos largely centred on incremental frame-to-frame pose tracking, e.g., (Bregler and Malik, 1998; Sigal et al., 2012; Sminchisescu and Triggs, 2003). These approaches rely on a given pose and dynamic model to constrain the pose search space. Notable drawbacks of this approach include: the requirement that the initialization be provided and their inability to recover from tracking failures. To address these limitations, more recent approaches have cast the tracking problem as one of data association across frames, i.e., "tracking-by-detection", e.g., (Andriluka et al., 2010). Here, candidate poses are first detected in each frame and subsequently a linking process attempts to establish temporally consistent poses.

Another strand of research has focused on methods that predict 3D poses by searching a database of exemplars (Jiang, 2010; Mori and Malik, 2006; Shakhnarovich et al., 2003) or via a discriminatively learned mapping from the image directly or image features to human joint locations (Agarwal and Triggs, 2006; Ionescu et al., 2014; Salzmann and Urtasun, 2010; Tekin et al., 2015; Yu et al., 2013). Recently, deep convolutional networks (CNNs) have emerged as a common element behind many state-of-the-art approaches, including human pose estimation, e.g., (Li and Chan, 2014; Li et al., 2015; Tompson et al., 2014; Toshev and Szegedy, 2014). Here, two general approaches can be distinguished. The first approach casts the pose estimation task as a joint location regression problem from the input image (Li and Chan, 2014; Li et al., 2015; Toshev and Szegedy, 2014). The second approach uses a CNN architecture for body part detection (Chen and Yuille, 2014;

Jain et al., 2014; Pfister et al., 2015; Tompson et al., 2014) and then typically enforces the 2D spatial relationship between body parts as a subsequent processing step. Similar to the latter approaches, the proposed approach uses a CNN-based architecture to regress confidence heat maps of 2D joint position predictions.

Most closely related to the present paper are generic factorization approaches for recovering 3D non-rigid shapes from image sequences captured with a single camera (Akhter et al., 2011; Bregler et al., 2000; Cho et al., 2015; Dai et al., 2012; Zhu et al., 2014c), i.e., non-rigid structure from motion (NRSFM), and human pose recovery models based on known skeletons (Lee and Chen, 1985; Park and Sheikh, 2011; Taylor, 2000; Valmadre and Lucey, 2010) or sparse representations (Akhter and Black, 2015; Fan et al., 2014; Ramakrishna et al., 2012; Zhou et al., 2015b,c). Much of this work has been realized by assuming manually labeled 2D joint locations; however, there is some recent work that has used a 2D pose detector to automatically provide the input joints (Wang et al., 2014) or solves the correspondence problem by matching a spatio-temporal pose model to candidate trajectories extracted from a video (Zhou and la Torre, 2014).

8.1.2 Contributions

In the light of previous research, the current paper makes the following contributions. Given a monocular video, two novel approaches for recovering the 3D human pose sequence are presented. The first approach assumes that the 2D poses are provided and proceeds by combining a sparse representation of 3D pose with temporal smoothness in the 3D domain to estimate the 3D poses. The second approach generalizes the first approach by relaxing the common but restrictive assumption that the 2D poses are provided or explicitly estimated (cf. Wang et al. (2014)) and instead treats the 2D pose as a latent variable. A CNN-based body joint detector is used to learn the uncertainty map for the image location of each joint. To estimate the 3D pose, an efficient EM algorithm is proposed, where the latent joint positions are marginalized to fully account for the uncertainty

in the 2D joint locations. Finally, empirical evaluation demonstrates that the proposed approaches are more accurate compared to extant approaches. In particular, in the case where 2D joint locations are provided, the proposed approach exceeds the accuracy of the state-of-the-art NRSFM baseline Dai et al. (2012) on the Human3.6M dataset Ionescu et al. (2014). In the case where the 2D landmarks are unknown, empirical results on the Human3.6M dataset demonstrate overall improvement over published results. Further, the proposed approach is shown to outperform a publicly available 2D pose estimation baseline on the challenging PennAction dataset Zhang et al. (2013). The code will be made publicly available upon publication.

8.2 Models

In this section, the models that describe the relationships between 3D poses, 2D poses and images are introduced.

8.2.1 Sparse representation of 3D poses

The 3D human pose is represented by the 3D locations of a set of p joints, which is denoted by $\mathbf{S}_t \in \mathbb{R}^{3\times p}$ for frame t. To reduce the ambiguity for 3D reconstruction, it is assumed that a 3D pose can be represented as a linear combination of predefined basis poses:

$$\mathbf{S}_t = \sum_{i=1}^{k} c_{it}\mathbf{B}_i, \tag{8.1}$$

where $\mathbf{B}_i \in \mathbb{R}^{3\times p}$ denotes a basis pose and c_{it} the corresponding weight. The basis poses are learned from training poses provided by a motion capture (MoCap) dataset. Instead of using the conventional active shape model Cootes et al. (1995b), where the basis set is small, a sparse representation is adopted which has proven in recent work to be capable of modelling the large variability of human pose, e.g., Akhter and Black (2015); Ramakrishna et al. (2012); Zhou et al. (2015b). That is, an overcomplete dictionary, $\{\mathbf{B}_1, \cdots, \mathbf{B}_k\}$, is learned with a relatively large number of basis poses, k, where the coefficients, c_{it}, are

assumed to be sparse. In the remainder of this paper, $\mathbf{c}_t$ denotes the coefficient vector $[c_{1t}, \cdots, c_{kt}]^T$ for frame t and $\mathbf{C}$ denotes the matrix composed of all $\mathbf{c}_t$.

8.2.2 Dependence between 2D and 3D poses

The dependence between a 3D pose and its imaged 2D pose is modelled with a weak perspective camera model:

$$\mathbf{W}_t = \mathbf{R}_t \mathbf{S}_t + \mathbf{T}_t \mathbf{1}^T, \tag{8.2}$$

where $\mathbf{W}_t \in \mathbb{R}^{2\times p}$ denotes the 2D pose in frame t, and $\mathbf{R}_t \in \mathbb{R}^{2\times 3}$ and $\mathbf{T}_t \in \mathbb{R}^2$ denote the camera rotation and translation, respectively. Note, the scale parameter in the weak perspective model is removed because the 3D structure, $\mathbf{S}_t$, can itself be scaled. In the following, $\mathbf{W}$, $\mathbf{R}$ and $\mathbf{T}$ denote the collections of $\mathbf{W}_t$, $\mathbf{R}_t$ and $\mathbf{T}_t$ for all t, respectively.

Considering the observation noise and model error, the conditional distribution of the 2D poses given the 3D pose parameters is modelled as

$$\Pr(\mathbf{W}|\theta) \propto e^{-\mathcal{L}(\theta;\mathbf{W})}, \tag{8.3}$$

where $\theta = \{\mathbf{C}, \mathbf{R}, \mathbf{T}\}$ is the union of all the 3D pose parameters and the loss function, $\mathcal{L}(\theta; \mathbf{W})$, is defined as

$$\mathcal{L}(\theta; \mathbf{W}) = \frac{\nu}{2} \sum_{t=1}^{n} \left\| \mathbf{W}_t - \mathbf{R}_t \sum_{i=1}^{k} c_{it} \mathbf{B}_i - \mathbf{T}_t \mathbf{1}^T \right\|_F^2, \tag{8.4}$$

with $\|\cdot\|_F$ denoting the Frobenius norm. The model in (8.3) states that, given the 3D poses and camera parameters, the 2D location of each joint belongs to a Gaussian distribution with a mean equal to the projection of its 3D counterpart and a precision (i.e., the inverse variance) equal to ν.

8.2.3 Dependence between pose and image

In the case where the 2D poses are given, it is assumed that the distribution of 3D pose parameters is conditionally independent of the image data. Therefore, the likelihood function of θ can be factorized as

$$\text{Pr}(\mathbf{I}, \mathbf{W}|\theta) = \text{Pr}(\mathbf{I}|\mathbf{W})\text{Pr}(\mathbf{W}|\theta), \tag{8.5}$$

where $\mathbf{I} = \{\mathbf{I}_1, \cdots, \mathbf{I}_n\}$ denotes the input images and $\text{Pr}(\mathbf{W}|\theta)$ is given in (8.3). $\text{Pr}(\mathbf{I}|\mathbf{W})$ is difficult to directly model, but it is proportional to $\text{Pr}(\mathbf{W}|\mathbf{I})$ by assuming uniform priors on $\mathbf{W}$ and $\mathbf{I}$, and $\text{Pr}(\mathbf{W}|\mathbf{I})$ can be learned from data.

Given the image data, the 2D distribution of each joint is assumed to be only dependent on the current image. Thus,

$$\text{Pr}(\mathbf{I}|\mathbf{W}) \propto \text{Pr}(\mathbf{W}|\mathbf{I}) = \Pi_t \Pi_j h_j(\mathbf{w}_{jt}; \mathbf{I}_t), \tag{8.6}$$

where $\mathbf{w}_{jt}$ denotes the image location of joint j in frame t, and $h_j(\cdot; \mathbf{Y})$ represents a mapping from an image $\mathbf{Y}$ to a probability distribution of joint location (termed heat map). For each joint j, the mapping h_j is approximated by a CNN learned from training data. The details of CNN learning are described in Section 8.4.

8.2.4 Prior on model parameters

The following penalty function on the model parameters is introduced:

$$\mathcal{R}(\theta) = \alpha\|\mathbf{C}\|_1 + \frac{\beta}{2}\|\nabla_t \mathbf{C}\|_F^2 + \frac{\gamma}{2}\|\nabla_t \mathbf{R}\|_F^2, \tag{8.7}$$

where $\|\cdot\|_1$ denotes the ℓ_1-norm (i.e., the sum of absolute values), and ∇_t the discrete temporal derivative operator. The first term penalizes the cardinality of the pose coefficients to induce a sparse pose representation. The second and third terms impose first-order smoothness on both the pose coefficients and rotations.

8.3 3D pose inference

In this section, the proposed approach to 3D pose inference is described. Here, two cases are distinguished: (i) the image locations of the joints are provided (Section 8.3.1) and (ii) the joint locations are unknown (Section 8.3.2).

8.3.1 Given 2D poses

When the 2D poses, $\mathbf{W}$, are given, the model parameters, θ, are recovered via penalized maximum likelihood estimation (MLE):

$$\begin{aligned}\theta^* &= \arg\max_{\theta} \ \ln \Pr(\mathbf{W}|\theta) - \mathcal{R}(\theta) \\ &= \arg\min_{\theta} \ \mathcal{L}(\theta; \mathbf{W}) + \mathcal{R}(\theta).\end{aligned} \tag{8.8}$$

The problem in (8.8) is solved via block coordinate descent, i.e., alternately updating $\mathbf{C}$, $\mathbf{R}$ or $\mathbf{T}$ while fixing the others. The update of $\mathbf{C}$ needs to solve:

$$\mathbf{C} \leftarrow \arg\min_{\mathbf{C}} \ \mathcal{L}(\mathbf{C}; \mathbf{W}) + \alpha\|\mathbf{C}\|_1 + \frac{\beta}{2}\|\nabla_t \mathbf{C}\|_F^2, \tag{8.9}$$

where the objective is the composite of two differentiable functions plus an ℓ_1 penalty. The problem in (8.9) is solved by accelerated proximal gradient (APG) Nesterov (2007). Since the problem in (8.9) is convex, global optimality is guaranteed. The update of $\mathbf{R}$ needs to solve:

$$\mathbf{R} \leftarrow \arg\min_{\mathbf{R}} \ \mathcal{L}(\mathbf{R}; \mathbf{W}) + \frac{\gamma}{2}\|\nabla_t \mathbf{R}\|_F^2, \tag{8.10}$$

where the objective is differentiable and the variables are rotations restricted to $SO(3)$. Here, manifold optimization is adopted to update the rotations using the trust-region solver in the Manopt toolbox Boumal et al. (2014). The update of $\mathbf{T}$ has the following closed-form solution:

$$\mathbf{T}_t \leftarrow \text{row mean} \left\{ \mathbf{W}_t - \mathbf{R}_t \sum_{i=1}^{k} c_{it} \mathbf{B}_i \right\}. \tag{8.11}$$

Algorithm 3: Block coordinate descent to solve (8.8).

Input: $\mathbf{W}$; // 2D joint locations

Output: $\mathbf{C}, \mathbf{R}, \mathbf{T}$; // pose parameters

initialize the parameters ; // Section 8.3.3

while *not converged* **do**

  update $\mathbf{C}$ by (8.9) with APG;

  update $\mathbf{R}$ by (8.10) with Manopt;

  update $\mathbf{T}$ by (8.11);

end

The entire algorithm for 3D pose inference given the 2D poses is summarized in Algorithm 3. The iterations are terminated once the objective value has converged. Since in each step the objective function is non-increasing, the algorithm is guaranteed to converge; however, since the problem in (8.8) is nonconvex, the algorithm requires a suitably chosen initialization (described in Section 8.3.3).

8.3.2 Unknown 2D poses

If the 2D poses are unknown, $\mathbf{W}$ is treated as a latent variable and is marginalized during the estimation process. The marginalized likelihood function is

$$\Pr(\mathbf{I}|\theta) = \int \Pr(\mathbf{I}, \mathbf{W}|\theta) d\mathbf{W}, \tag{8.12}$$

where $\Pr(\mathbf{I}, \mathbf{W}|\theta)$ is given in (8.5).

Direct marginalization of (8.12) is extremely difficult. Instead, an EM algorithm is developed to compute the penalized MLE. In the expectation step, the expectation of the penalized log-likelihood is calculated with respect to the conditional distribution of $\mathbf{W}$

given the image data and the previous estimate of all the 3D pose parameters, θ':

$$\begin{aligned} Q(\theta|\theta') &= \int \{\ln \Pr(\mathbf{I}, \mathbf{W}|\theta) - \mathcal{R}(\theta)\} \Pr(\mathbf{W}|\mathbf{I}, \theta') d\mathbf{W} \\ &= \int \{\ln \Pr(\mathbf{I}|\mathbf{W}) + \ln \Pr(\mathbf{W}|\theta) - \mathcal{R}(\theta)\} \Pr(\mathbf{W}|\mathbf{I}, \theta') d\mathbf{W} \\ &= \text{const} - \int \mathcal{L}(\theta; \mathbf{W}) \Pr(\mathbf{W}|\mathbf{I}, \theta') d\mathbf{W} - \mathcal{R}(\theta). \end{aligned} \tag{8.13}$$

It can be easily shown that

$$\int \mathcal{L}(\theta; \mathbf{W}) \Pr(\mathbf{W}|\mathbf{I}, \theta') d\mathbf{W} = \mathcal{L}(\theta; \mathbb{E}[\mathbf{W}|\mathbf{I}, \theta']) + \text{const}, \tag{8.14}$$

where $\mathbb{E}[\mathbf{W}|\mathbf{I}, \theta']$ is the expectation of $\mathbf{W}$ given $\mathbf{I}$ and θ':

$$\begin{aligned} \mathbb{E}[\mathbf{W}|\mathbf{I}, \theta'] &= \int \Pr(\mathbf{W}|\mathbf{I}, \theta') \ \mathbf{W} \ d\mathbf{W} \\ &= \int \frac{\Pr(\mathbf{I}|\mathbf{W}) \Pr(\mathbf{W}|\theta')}{Z} \ \mathbf{W} \ d\mathbf{W}, \end{aligned} \tag{8.15}$$

and Z is a constant that normalizes the probability. The derivation of (8.14) and (8.15) is given in the supplementary material. Both $\Pr(\mathbf{I}|\mathbf{W})$ and $\Pr(\mathbf{W}|\theta')$ given in (8.6) and (8.3), respectively, are products of marginal probabilities of $\mathbf{w}_{jt}$. Therefore, the expectation of each $\mathbf{w}_{jt}$ can be computed separately. In particular, the expectation of each $\mathbf{w}_{jt}$ is efficiently approximated by sampling over the pixel grid.

In the maximization step, the following is computed:

$$\begin{aligned} \theta &\leftarrow \arg\max_{\theta} Q(\theta|\theta') \\ &= \arg\min_{\theta} \ \mathcal{L}(\theta; \mathbb{E}[\mathbf{W}|\mathbf{I}, \theta']) + \mathcal{R}(\theta), \end{aligned} \tag{8.16}$$

which can be solved by Algorithm 3.

The entire EM algorithm is summarized in Algorithm 4 with the initialization scheme described next in Section 8.3.3.

8.3.3 Initialization

The convex relaxation approach proposed elsewhere Zhou et al. (2015b,c) is used to initialize the parameters. In Zhou et al. (2015b), a convex formulation was proposed to solve

Algorithm 4: The EM algorithm for pose from video.

Input: $h_j(\cdot; \mathbf{I}_t),\ \forall j, t$; // heat maps

Output: $\theta = \{\mathbf{C}, \mathbf{R}, \mathbf{T}\}$; // pose parameters

initialize the parameters ; // Section 8.3.3

while *not converged* **do**

  $\theta' = \theta$;

  // Compute the expectation of $\mathbf{W}$

  $\mathbb{E}\left[\mathbf{W}|\mathbf{I}, \theta'\right] = \int \frac{1}{Z} \Pr(\mathbf{I}|\mathbf{W}) \Pr(\mathbf{W}|\theta')\ \mathbf{W}\ d\mathbf{W}$;

  // Update θ by Algorithm 3

  $\theta = \arg\min_{\theta}\ \mathcal{L}(\theta; \mathbb{E}\left[\mathbf{W}|\mathbf{I}, \theta'\right]) + \mathcal{R}(\theta)$;

end

the single-frame pose estimation problem given 2D correspondences, which is a special case of (8.8). The approach was later extended to handle 2D correspondence outliers Zhou et al. (2015c). If the 2D poses are given, the model parameters are initialized for each frame separately with the convex method proposed in Zhou et al. (2015b). Alternatively, if the 2D poses are unknown, for each joint, the image location with the maximum heat map value is used. Next, the robust estimation algorithm from Zhou et al. (2015c) is applied to initialize the parameters.

8.4 CNN-based joint uncertainty regression

In this section, the details are provided for using CNNs to learn the mapping $\mathbf{Y} \mapsto h_j(\cdot; \mathbf{Y})$, where $\mathbf{Y}$ denotes an input image and $h_j(\cdot; \mathbf{Y})$ represents a heat map for joint j. Instead of learning p networks for p joints, a fully convolutional neural network Long et al. (2015) is trained to regress p joint distributions simultaneously by taking into account the full-body information.

During training, a rectangular patch is extracted around the subject from each image and is resized to 256×256 pixels. Random shifts are applied during cropping and RGB channel-wise random noise is added for data augmentation. Channel-wise RGB mean values are computed from the dataset and subtracted from the images for data normalization. The training labels to be regressed are multi-channel heat maps with each channel corresponding to the image location uncertainty distribution for each joint. The uncertainty is modelled by a Gaussian centered at the annotated joint location. The heat map resolution is reduced to 32×32 to decrease the CNN model size which allows a large batch size in training and prevents overfitting.

The CNN architecture used is similar to the SpatialNet model proposed elsewhere Pfister et al. (2015) but without any spatial fusion or temporal pooling. The network consists of seven convolutional layers with 5×5 filters followed by ReLU layers and a last convolutional layer with $1 \times 1 \times p$ filters to provide dense prediction for all joints. A 2×2 max pooling layer is inserted after each of the first three convolutional layers. The network is trained by minimizing the l_2 loss between the prediction and the label with the open source Caffe framework Jia et al. (2014). Stochastic gradient descent (SGD) with momentum of 0.9 and a mini-batch size of 128 is used. During testing, an image patch $\mathbf{I}_t$ is cropped around the subject in frame t and fed forward through the network to predict the heat maps, $h_j(\cdot; \mathbf{I}_t),\ \forall j = 1, \cdots, n$.

8.5 Experiments

8.5.1 Datasets and implementation details

Empirical evaluation was performed on two datasets – Human3.6M Ionescu et al. (2014) and PennAction Zhang et al. (2013). The demonstration videos are provided in the supplementary material.

The Human3.6M dataset Ionescu et al. (2014) is a recently published large-scale dataset for 3D human sensing. It includes millions of 3D human poses acquired from a MoCap

system with corresponding images from calibrated cameras. This setup provides synchronized videos and 2D-3D pose data for evaluation. It includes 11 subjects performing 15 actions, such as walking, sitting and discussion. For comparison convenience, the same data partition protocol as in previous work was used Li et al. (2015); Tekin et al. (2015): the data from five subjects (S1, S5, S6, S7, S8) was used for training and the data from two subjects (S9, S11) was used for testing. The original frame rate is 50 fps and is downsampled to 10 fps.

The PennAction dataset Zhang et al. (2013) is a recently introduced in-the-wild human action dataset containing 2326 challenging consumer videos. The dataset consists of 15 actions, such as golf swing, bowling, and tennis swing. Each of the video sequences is manually annotated frame-by-frame with 13 human body joints in 2D. In evaluation, PennAction's training and testing split was used which consists of an even split of the videos between training and testing.

The algorithm in Zhou et al. (2015c) was used to learn the pose dictionaries. The dictionary size was set to $K = 64$ for action-specific dictionaries and $K = 128$ for the nonspecific action case. For all experiments, the parameters of the proposed model were fixed ($\alpha = 0.1$, $\beta = 5$, $\gamma = 0.5$, $\nu = 4$ in a normalized 2D coordinate system).

8.5.2 Reconstruction with known 2D poses

First, the evaluation of the 3D reconstructability of the proposed method given perfect 2D poses is presented. The generic approach to 3D reconstruction from 2D correspondences across a sequence is NRSFM. The proposed method is compared to the state-of-the-art method for NRSFM Dai et al. (2012) on the Human3.6M dataset.

Performance is evaluated by the mean per joint error (mm) in 3D by comparing the reconstructed pose against the ground truth. As the standard protocol for evaluating NRSFM, the error is calculated up to a similarity transformation via the Procrustes analysis, where a single rotation is applied to the entire sequence instead of frame-by-frame rotation adjustment. To demonstrate the generality of the proposed approach, a single

	Original	Synthesized
NRSFM Dai et al. (2012)	83.04	51.94
Single-frame initialization	56.49	54.14
Optimization by Algorithm 3	**54.43**	**50.97**

Table 8.1: 3D reconstruction given 2D poses. Two input cases are considered: original 2D pose data from Human3.6M and synthesized 2D pose data with artificial camera motion. The numbers are the mean per joint errors (mm) in 3D.

pose dictionary from all the training pose data, irrespective of the action type, was used, i.e., a non-action specific model. The method from Dai et al. Dai et al. (2012) requires a predefined rank K. Here, various values of K were considered with the best result for each sequence reported.

The 2D joint locations provided in the Human 3.6M dataset were used as the input. The results are shown in the second column of Table 8.1. The proposed method clearly outperforms the NRSFM baseline. The reason is that the videos are captured by stationary cameras. Although the subject is occasionally rotating, the "baseline" between frames is generally small, and neighboring views provide insufficient geometric constraints for 3D reconstruction. In other words, NRSFM is very difficult to compute with slow camera motion. This observation is consistent with prior findings in the NRSFM literature, e.g., Akhter et al. (2011). To validate this issue, an artificial rotation was applied to the 3D poses by 15 degrees per second and the 2D joint locations were synthesized by projecting the rotated 3D poses into 2D. The corresponding results are presented in the third column of Table 8.1. In this case, the performance of NRSFM improved dramatically. Overall, the experiments demonstrate that the structure prior (even a non-action specific one) from existing pose data is critical for reconstruction. This is especially true for videos with small camera motion, which is common in real world applications. The temporal smoothness helps but the change is not significant since the single-frame initialization is very stable

	Directions	Discussion	Eating	Greeting	Phoning
LinKDE Ionescu et al. (2014)	132.71	183.55	132.37	164.39	162.12
Li et al. Li et al. (2015)	-	136.88	96.94	124.74	-
Tekin et al. Tekin et al. (2015)	102.39	158.52	87.95	126.83	118.37
Proposed	**87.36**	**109.31**	**87.05**	**103.16**	**116.18**
	Photo	Posing	Purchases	Sitting	SittingDown
LinKDE Ionescu et al. (2014)	205.94	150.61	171.31	151.57	243.03
Li et al. Li et al. (2015)	168.68	-	-	-	-
Tekin et al. Tekin et al. (2015)	185.02	114.69	107.61	136.15	205.65
Proposed	**143.32**	**106.88**	**99.78**	**124.52**	**199.23**
	Smoking	Waiting	WalkDog	Walking	WalkTogether
LinKDE Ionescu et al. (2014)	162.14	170.69	177.13	96.60	127.88
Li et al. Li et al. (2015)	-	-	132.17	69.97	-
Tekin et al. Tekin et al. (2015)	118.21	146.66	128.11	**65.86**	**77.21**
Proposed	**107.42**	**118.09**	**114.23**	79.39	97.70
	Average				
LinKDE Ionescu et al. (2014)	162.14				
Li et al. Li et al. (2015)	-				
Tekin et al. Tekin et al. (2015)	125.28				
Proposed	**113.01**				

Table 8.2: Quantitative comparison on Human 3.6M datasets. The numbers are the mean per joint errors (mm) in 3D evaluated for different actions of Subject 9 and 11.

	S11	S9	Both
Directions	5.58	10.29	7.94
Discussion	6.63	18.33	12.48
Eating	9.93	10.69	10.31
Greeting	9.48	17.38	13.43
Phoning	11.64	15.96	13.80
Photo	17.20	15.84	16.52
Posing	6.46	12.95	9.71
Purchases	13.86	14.07	13.96
Sitting	12.79	19.58	16.19
SittingDown	31.78	47.60	39.69
Smoking	12.70	13.97	13.33
Waiting	7.81	20.63	14.22
WalkTogether	12.22	12.88	12.55
Walking	7.40	10.47	8.93
WalkingDog	15.86	15.45	15.66
Average	12.09	17.07	14.58

Table 8.3: 2D joint error from CNN regression on Human 3.6M datasets. The numbers are the mean per joint errors in 2D in pixel under the image size of 256×256, evaluated for different actions of Subject 9 and 11.

	3D (mm)	2D (pixel)
Single-frame initialization	143.85	15.00
Optimization by Algorithm 4	125.55	10.85
Perspective adjustment	**113.01**	**10.85**
No smoothness	120.99	11.25
No action label	116.49	10.87

Table 8.4: The estimation errors after separate steps and under additional settings. The numbers are the average per joint errors for all testing data in both 3D and 2D.

given perfect 2D poses. Nevertheless, in the next section it is shown that the temporal smoothness is important when 2D poses are not given.

8.5.3 Evaluation with unkown poses: Human3.6M

Next, results on the Human3.6M dataset are reported when 2D poses are not given. In this experiment, the Human3.6M dataset evaluation protocol was adopted, i.e., the reconstructed pose is compared to the ground truth in the camera frame with their root locations aligned (rotation is not allowed). Results are compared to three recent baseline methods based on the mean per joint errors. The first baseline method is LinKDE which is provided with the Human3.6M dataset Ionescu et al. (2014). This baseline is based on single-frame regression. The second one is from Tekin et al. Tekin et al. (2015) which extends the first baseline method by exploring motion information in a short sequence. The third one is a recently published CNN-based method from Li et al. Li et al. (2015).

In general, it is impossible to determine the scale of the object in monocular images. The baseline methods learn the scale from training subjects. For a fair comparison, the

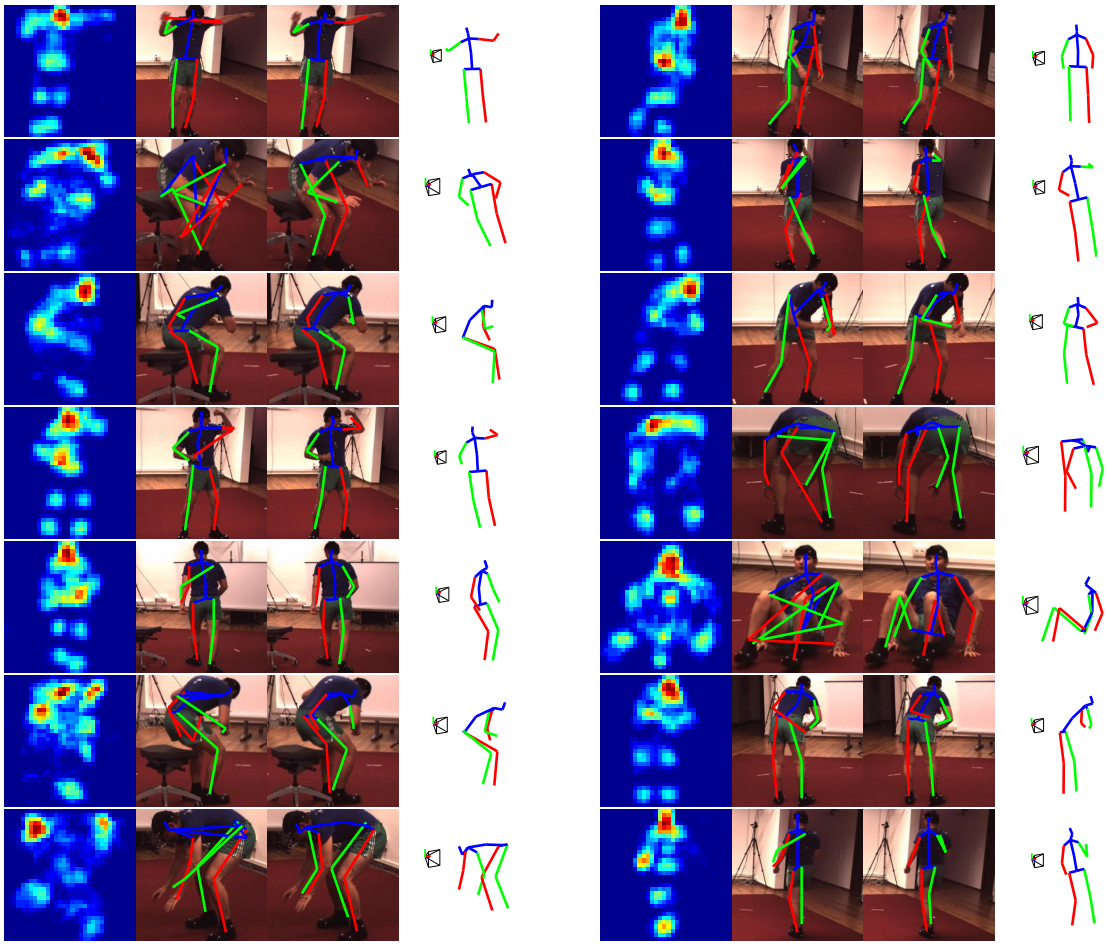


Figure 8.2: Example frame results on Human3.6M, where the errors in the 2D heat maps are corrected after considering the pose and temporal smoothness priors. Each row includes two examples from two actions. The figures from left-to-right correspond to the heat map (all joints combined), the 2D pose by greedily locating each joint separately according to the heat map, the estimated 2D pose by the proposed EM algorithm, and the estimated 3D pose visualized in a novel view. The original viewpoint is also shown.

reconstructed pose by the proposed method was scaled such that the mean limb length of the reconstructed pose was identical to the average value of all training subjects. It was found that the camera model influenced the rotation estimation. To compensate for effects of the adopted weak-perspective model, the estimated rotation was refined with a perspective camera model. In particular, the estimated 3D pose was aligned to the estimated 2D pose by a perspective-n-point (PnP) algorithm Lu et al. (2000).

The results are summarized in Table 8.2. The table shows that the proposed method achieves the best results on most of the actions except for "walk" and "walk together", which involve very predictable and repetitive motions and might favor the direct regression approach Tekin et al. (2015). In addition, the results of the proposed approach have the smallest variation across all actions with a standard deviation of 28.75 versus 37.80 from Tekin et al. The raw 2D joint errors are reported in Table 8.3

In Table 8.4, 3D reconstruction and 2D joint localization results are provided under several setup variations of the proposed approach. Note that the 2D errors are with respect to the normalized bounding box size 256×256. The table shows that the convex initialization provides suitable initial estimates, which are further improved by the EM algorithm that integrates joint detection uncertainty and temporal smoothness. The perspective adjustment is important under the Human3.6M evaluation protocol, where Procrustes alignment to the ground truth is not allowed. The proposed approach was also evaluated under two additional settings. In the first setting, the smoothness constraint was removed from the proposed model by setting $\beta = \gamma = 0$. As a result, the average error significantly increased. This demonstrates the importance of incorporating temporal smoothness. In the second setting, a single CNN and pose dictionary was learned from all training data. These models were then applied to all testing data without distinguishing the videos by their action class. As a result, the estimation error increased, which is attributed to the fact that the 3D reconstruction ambiguity is greatly enlarged if the pose prior is not restricted to an action class.

Figure 8.2 visualizes the results of some example frames. While the heat maps may be

	Walking			Jogging		
	S1	S2	S3	S1	S2	S3
Proposed	34.2	30.9	49.1	47.6	33.0	29.7
Simo-Serra et al. Simo-Serra et al. (2013)	65.1	48.6	73.5	74.2	46.6	32.2

Table 8.5: Quantitative results on the HumanEva I dataset Sigal et al. (2010). The numbers are the mean per joint errors in millimeters.

erroneous due to occlusion, left-right ambiguity, and other uncertainty from the detectors, the proposed EM algorithm can largely correct the errors by leveraging the pose prior, integrating temporal smoothness, and modelling the uncertainty.

8.5.4 Evaluation with unkown poses: HumanEva

The evaluation results on the HumanEva I dataset Sigal et al. (2010) are presented. The evaluation protocol described in Simo-Serra et al. (2013) was adopted. The walking and jogging sequences from camera C1 of all subjects were used for evaluation. The CNN joint detectors trained on the Human3.6M dataset were fine-tuned with the training sequences for each action separately. Each estimated 3D pose was aligned to the ground truth with the procrustes method. The mean 3D joint errors for the evaluation sequences were reported in Table 8.5.

8.5.5 Evaluation with unkown poses: PennAction

Finally, the applicability of the proposed approach for pose estimation with in-the-wild videos is demonstrated. Results are reported using two actions from the PennAction dataset: "golf-swing" and "tennis-forehand", both of which are very challenging due to large pose variability, self-occlusion, and image blur caused by fast motion. For the proposed approach, the CNN was trained using the annotated training images from the

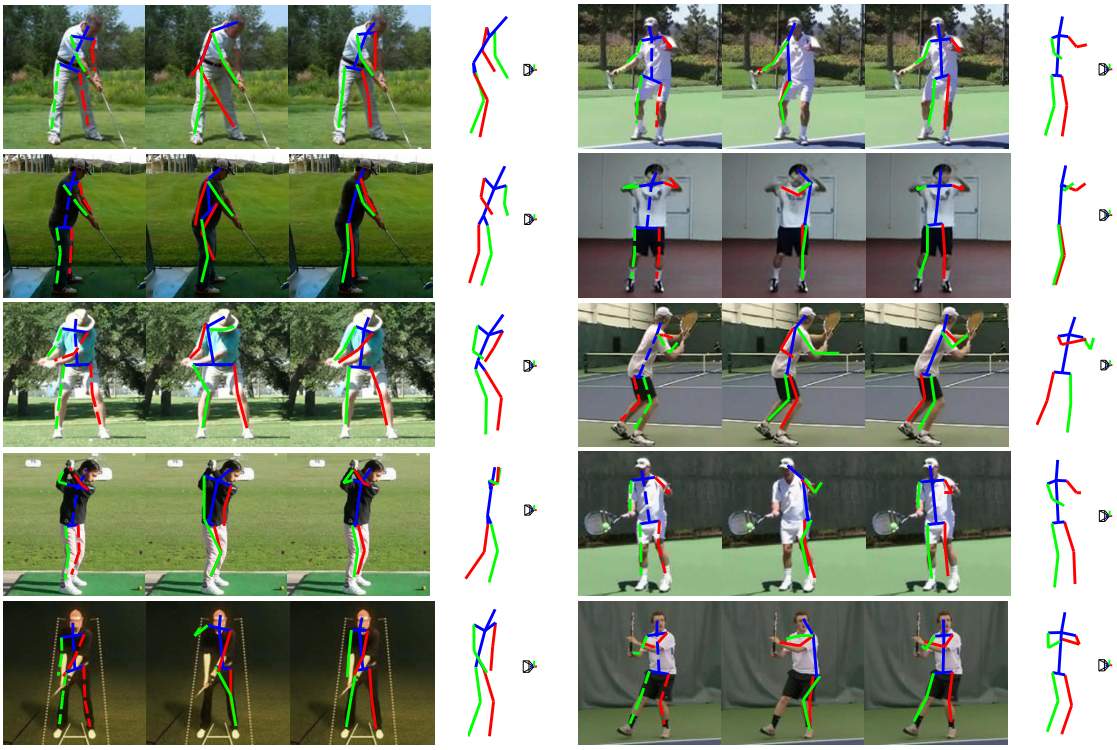


Figure 8.3: Example results on PennAction. Each row includes two examples. In each example, the figures from left-to-right correspond to the ground truth superposed on the image, the estimated pose from Yang and Ramanan (2011), the estimated pose by the proposed approach, and the estimated 3D pose visualized in a novel view. The original viewpoint is also shown.

PennAction dataset, while the pose dictionary was learned with publicly available MoCap data[1]. Due to the lack of 3D ground truth, quantitative 2D pose estimation results are reported and compared with the publicly available 2D pose detector from Yang and Ramanan Yang and Ramanan (2011). The baseline was retrained on the PennAction dataset. Note that the baseline methods considered in Section 8.5.3 are not applicable here since they require synchronized 2D image and 3D pose data for training.

To measure joint localization accuracy, both the widely used per joint distance errors and the probability of correct keypoint (PCK) metrics are used. The PCK metric measures the fraction of correctly located joints with respect to a threshold. Here, the threshold is set to 10 pixels which is roughly the half length of a head segment.

Table 8.6 summarizes the quantitative results. The initialization step alone outperformed the baseline. This demonstrates the effectiveness of CNN-based approaches, which has been shown in many recent works Pfister et al. (2015); Toshev and Szegedy (2014). The proposed EM algorithm further improves upon the initialization results by a large margin by integrating the geometric and smoothness priors. Table 8.7 shows the effect of smoothness constrain in EM optimization. Several example results are shown in Figure 8.3. It can be seen that the proposed method successfully recovers the poses for various subjects under a variety of viewpoints. In particular, compared to the baseline, the proposed method does not suffer from the well-known "double-counting" problem for tree-based models Yang and Ramanan (2011) due to the holistic 3D pose prior.

8.5.6 Running time

The experiments were performed on a desktop with an Intel i7 3.4G CPU, 8G RAM and a TitanZ GPU. The running times for CNN-based heat map generation and convex initialization were roughly 1s and 0.6s per frame, respectively; both the steps can be easily parallelized. The EM algorithm usually converged in 20 iterations with a CPU time less than 100s for a sequence of 300 frames.

[1]Data sources: http://mocap.cs.cmu.edu and http://www.motioncapturedata.com

	Baseline	Initial	Optimized
Golf	24.78 / 0.38	18.73 / 0.45	**14.03 / 0.54**
Tennis	29.15 / 0.40	25.75 / 0.42	**20.99 / 0.45**

Table 8.6: 2D pose errors on the PennAction dataset. Each pair of numbers correspond to the per joint distance error (pixels) and the PCK metric. The baseline is the retrained model from Yang and Ramanan Yang and Ramanan (2011). The last two columns correspond to the errors after initialization and EM optimization in our approach.

	Baseline	Without smoothness	With smoothness
Golf	24.78	14.49	**14.03**
Tennis	29.15	21.37	**20.99**

Table 8.7: Comparison of 2D pose errors on the PennAction dataset with and without temporal smoothness terms. Each number corresponds to the per joint distance error (pixels). The baseline is the retrained model from Yang and Ramanan Yang and Ramanan (2011). The last two columns correspond to the errors with temporal smoothness and without temporal smoothness in the EM optimization in our approach.

Part IV

Discussion and Conclusions

In summary, this thesis presents a suite of solutions for 3D object recognition, including 2D detection and 3D pose and shape estimation from single images and articulted pose estimation from monocular video.

Fast Object Detection First, an active part selection approach which substantially speeds up inference with pictorial structures without sacrificing accuracy is presented. Statistics learned from training data are used to pose an optimization problem, which balances the number of part filter convolution with the classification accuracy. Unlike existing approaches, which use a pre-specified part order and hard stopping thresholds, the resulting part scheduling policy selects the part order and the stopping criterion adaptively based on the filter responses obtained during inference. Potential future extensions include optimizing the part selection across scales and image positions and detecting multiple classes simultaneously.

3D Object Instance Pose Estimtation Second, we presented an integrated approach for detecting and localizing 3D objects using pure geometric information derived from a database of 3D models. We create an initial set of hypotheses with a state-of-the-art parts-based model trained on clusters of poses. Detection hypotheses are segmented and reranked by matching subsets of superpixels with model boundary silhouettes using the chordiogram descriptor. The resulting segmentation enables the refinement of 3D pose in a small number of steps. Due to the holistic nature of the chordiogram-based superpixel selection, our approach is resistant to clutter. We demonstrate the grasps of textureless objects in difficult cluttered environments in the video supplement.

3D Object Category Joint Pose and Shape Estimtation Third, we proposed a novel approach for estimating the pose and the shape of a 3D object from a single image. Our approach is based on a collection of automatically-selected and discriminatively-trained 2D parts with a 3D shape-space model to represent the geometric relation. In model inference, we simultaneously localized the parts, estimated the pose, and recovered the

3D shape by solving a convex program with ADMM.

Articulated Pose Estimtation Finally, a 3D pose estimation framework from video has been presented that consists of a novel synthesis between a deep learning-based 2D part regressor, a sparsity-driven 3D reconstruction approach and a 3D temporal smoothness prior. This joint consideration combines the discriminative power of state-of-the-art 2D part detectors, the expressiveness of 3D pose models and regularization by way of aggregating information over time. In practice, alternative joint detectors, pose representations and temporal models can be conveniently integrated in the proposed framework by replacing the original components. Experiments demonstrated that 3D geometric priors and temporal coherence can not only help 3D reconstruction but also improve 2D joint localization. Future extensions may include incremental algorithms for online tracking-by-detection and handling multiple subjects.

Part V

Appendix: Additional Results

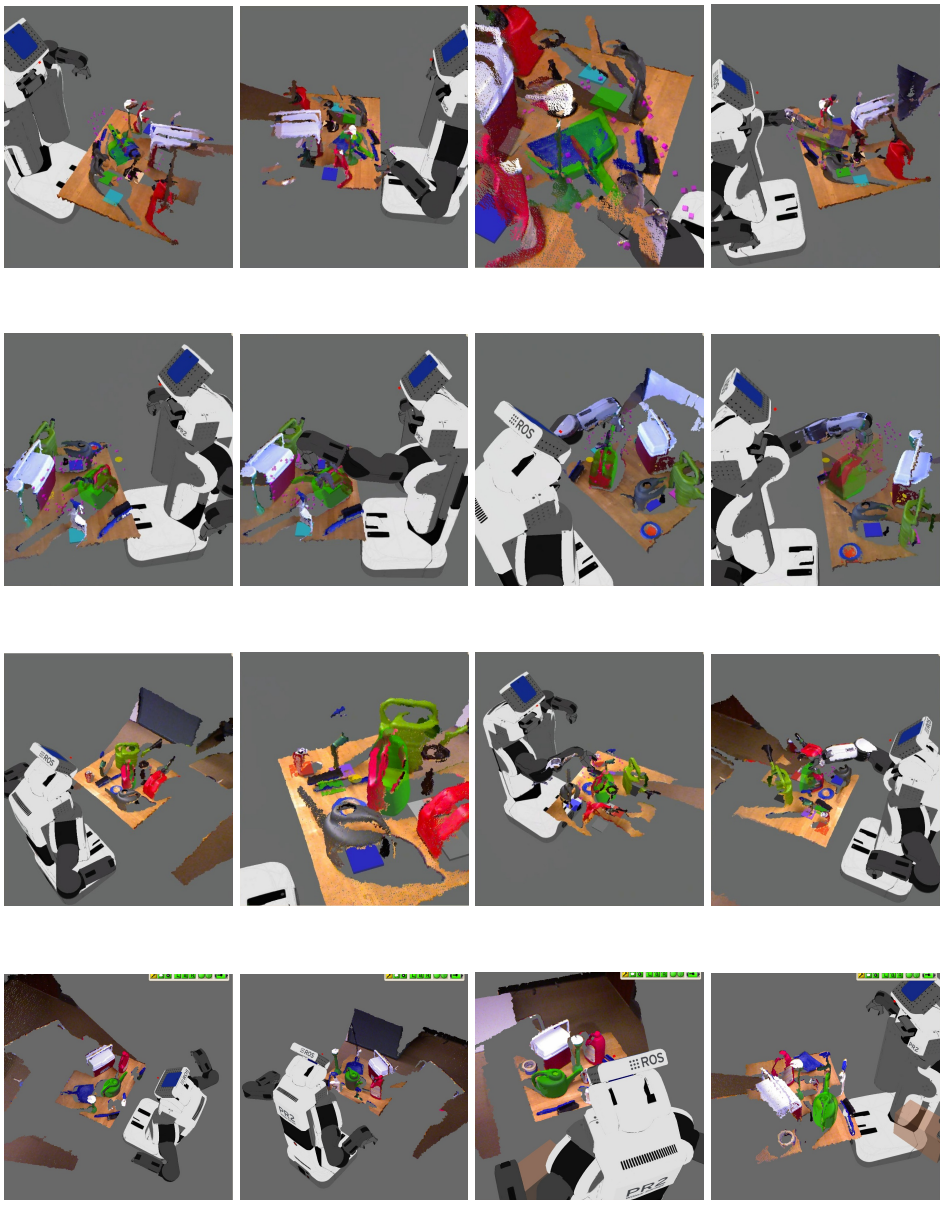


Figure 8.4: Example screen shots of PR2 grasping objects from a single image captured from the ROS visualization view. Each row shows different views of the grasping the same object. (Section 6)

Figure 8.5: In practice, due to invisible landmark caused by the self-occlusion the PopUp method requires iterative optimization to achieve the best results. The intermediate results are shown. The left column shows the results with all the landmarks including occluded points. The middle column shows the results after visibility pruning. The right column shows the final results with local pruning and shape space optimization. (Section 7)

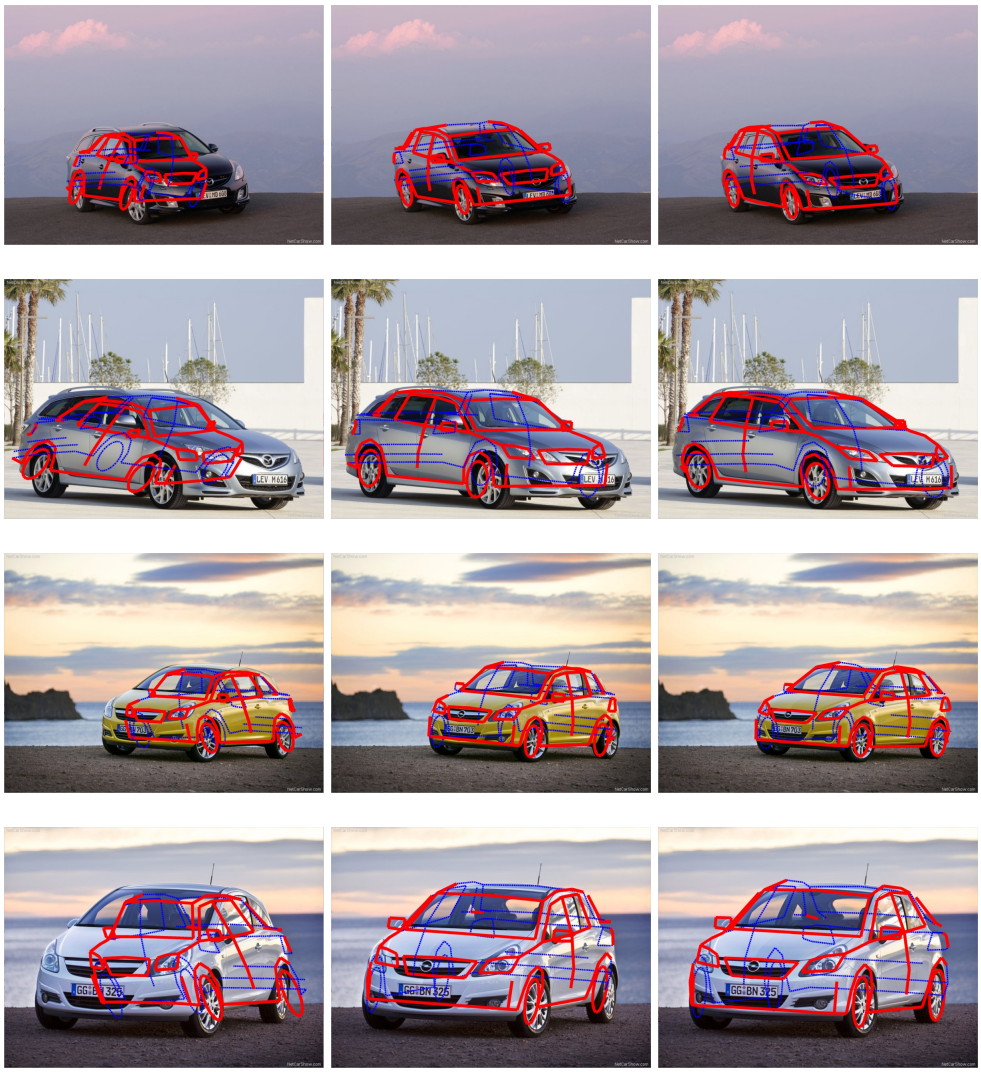


Figure 8.6: In practice, due to invisible landmark caused by the self-occlusion the PopUp method requires iterative optimization to achieve the best results. The intermediate results are shown. The left column shows the optimization results with all the landmarks including invisible points. The middle column shows the results after visibility pruning. The right column shows the final results with local pruning and shape space optimization. (Section 7)

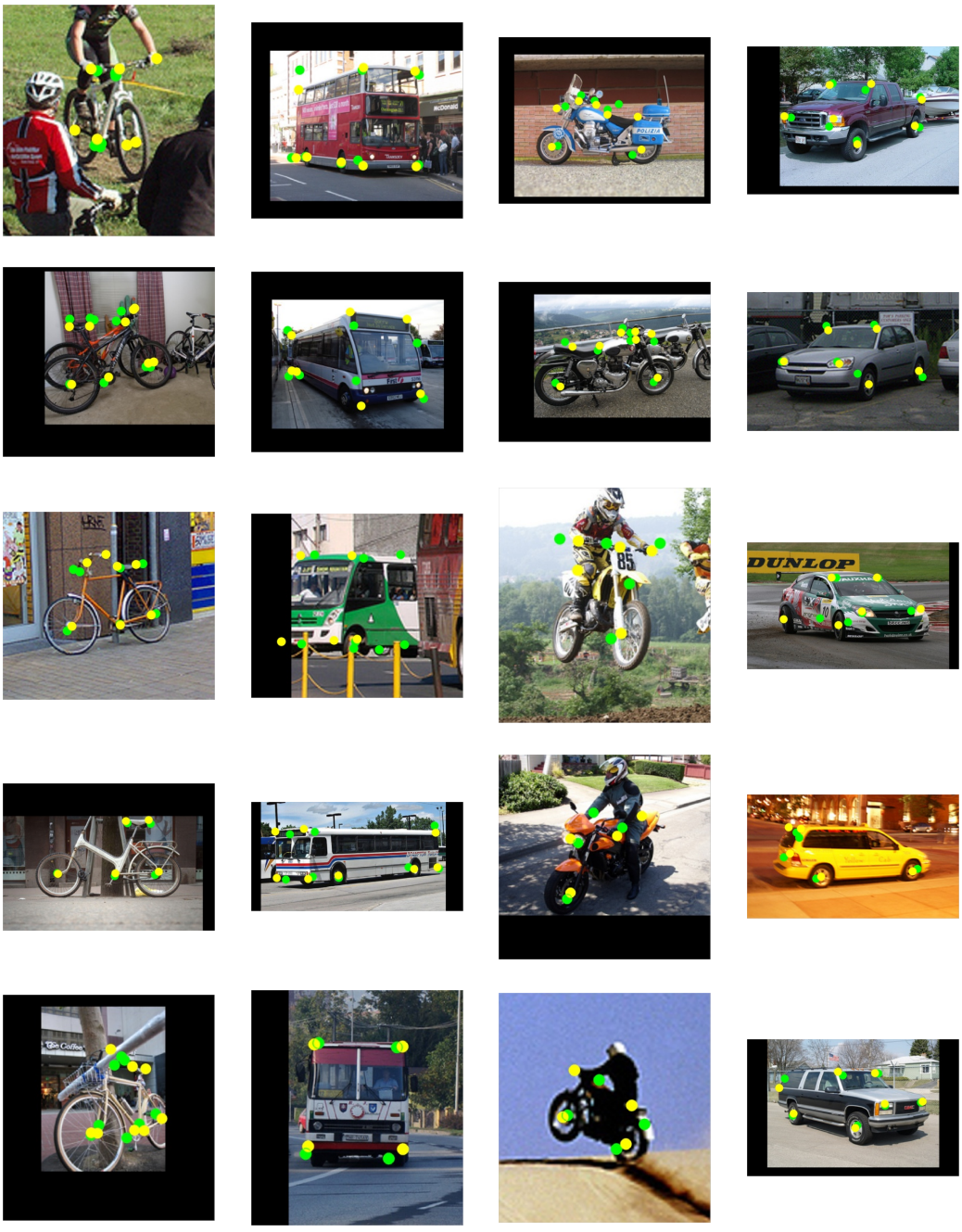


Figure 8.7: Landmark reprojection results on Pascal3D. (Section 7)

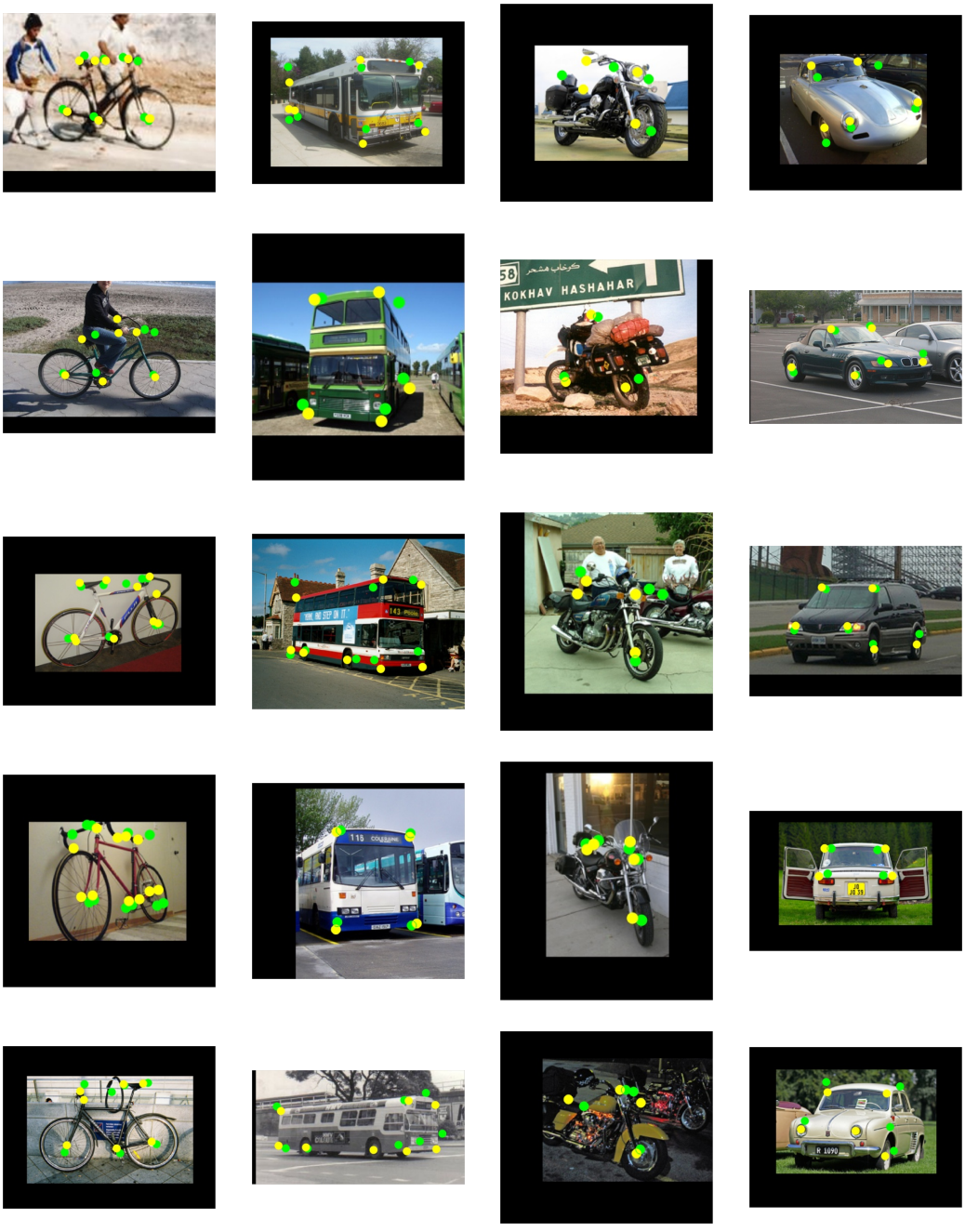


Figure 8.8: Landmark reprojection results on Pascal3D (continued). (Section 7)

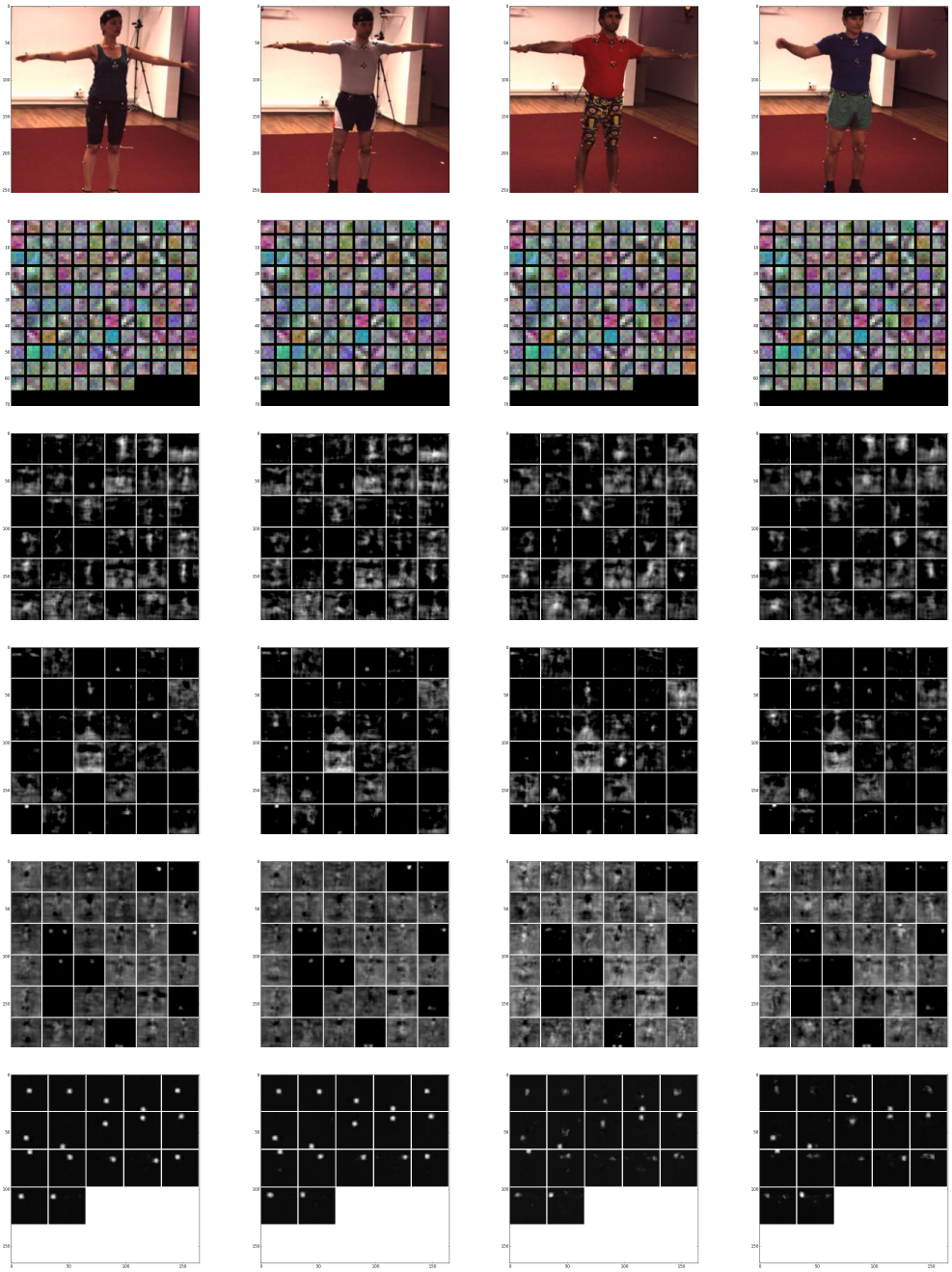


Figure 8.9: Learned filters and activations in different layers of the CNN on example training and testing images. The left two columns corresponding to training and right two columns testing. The first row shows the input images. The second row shows a subset of learned convolution kernels in the first CNN layer. Row 3-6 show the activation map of different kernels at each CNN layers. (Section 8)

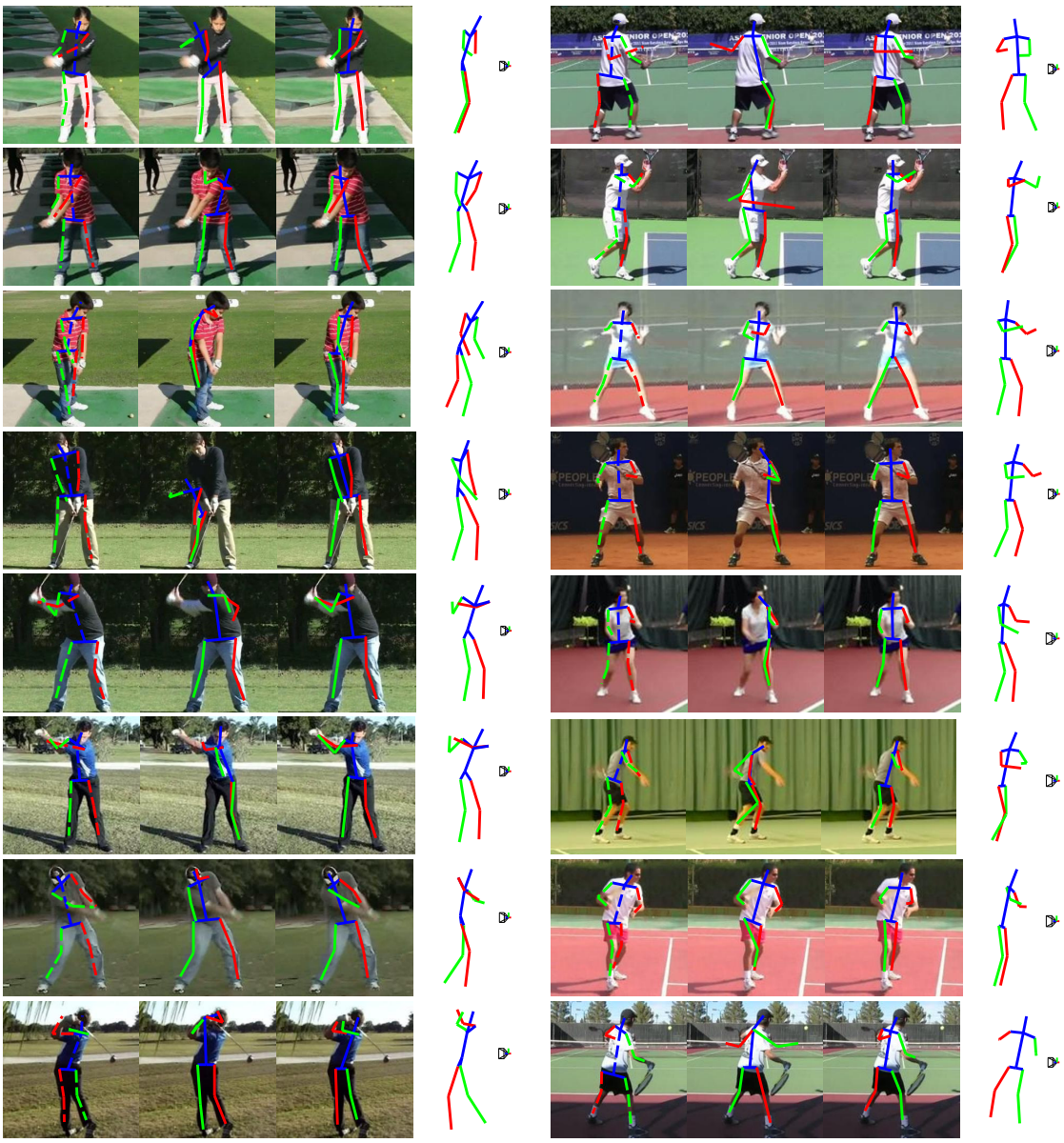


Figure 8.10: Additional example results on PennAction. Each row includes two examples. In each example, the figures from left-to-right correspond to the ground truth superposed on the image, the estimated pose from Yang and Ramanan (2011), the estimated pose by the proposed approach, and the estimated 3D pose visualized in a novel view. The original viewpoint is also shown. (Section 8)

Bibliography

Ankur Agarwal and Bill Triggs. Recovering 3D human pose from monocular images. *PAMI*, 28(1):44–58, 2006. 24, 108

Ijaz Akhter and Michael J. Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *Proc. CVPR*, 2015. 25, 109, 110

Ijaz Akhter, Yaser Sheikh, Sohaib Khan, and Takeo Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *PAMI*, 33(7):1442–1456, 2011. 25, 109, 119

Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Monocular 3D pose estimation and tracking by detection. In *Proc. CVPR*, 2010. 24, 107, 108

Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2D human pose estimation: New benchmark and state of the art analysis. In *Proc. CVPR*, 2014. 106

P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 33(5):898–916, 2011. 69, 70, 74

N. Atanasov, M. Zhu, K. Daniilidis, and G. Pappas. Localization from Semantic Observations via the Matrix Permanent. *International Journal of Robotics Research*, 2015. 17

Mathieu Aubry, Daniel Maturana, Alexei Efros, Bryan Russell, and Josef Sivic. Seeing

3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In *CVPR*, 2014. 23, 86

Ronen Basri. Recognition by prototypes. In *Conference on Computer Vision and Pattern Recognition, CVPR 1993, 15-17 June, 1993, New York, NY, USA*, pages 161–167, 1993. 21, 68

Dimitri P Bertsekas. *Dynamic Programming and Optimal Control*. Number 1. Athena Scientific, 1995. 50

P.J. Besl and N.D. McKay. A method for registration of 3-D shapes. *PAMI*, 14(2):239–256, February 1992. 79

Nicolas Boumal, Bamdev Mishra, P-A Absil, and Rodolphe Sepulchre. Manopt, a matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15: 1455–1459, 2014. 113

Lubomir Bourdev and Jonathan Brandt. Robust object detection via soft cascade. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 236–243. IEEE, 2005. 19, 43

S. Boyd. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010. 94

S.P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004. 28, 33

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. volume 3, pages 1–122. Now Publishers Inc., 2011. 28, 35, 38

C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, 2000. 25, 92, 109

Christoph Bregler and Jitendra Malik. Tracking people with twists and exponential maps. In *Proc. CVPR*, 1998. 24, 108

Marcus A. Brubaker, Leonid Sigal, and David J. Fleet. Video-based people tracking. In *Handbook of Ambient Intelligence and Smart Environments*, pages 57–87. Springer, 2010. 24, 108

S Charles Brubaker, Jianxin Wu, Jie Sun, Matthew D Mullin, and James M Rehg. On the design of cascades of boosted ensembles for face detection. *IJCV*, 77(1-3):65–86, 2008. 19, 43

Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998. 29

John Canny. A computational approach to edge detection. *PAMI*, (6):679–698, 1986. 70

Xianjie Chen and Alan Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *NIPS*, 2014. 24, 108

Jungchan Cho, Minsik Lee, and Songhwai Oh. Complex non-rigid 3D shape recovery using a procrustean normal distribution mixture model. *IJCV*, pages 1–21, 2015. 25, 109

T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models – their training and application. *CVIU*, 61(1):38–59, 1995a. 22, 85, 92

Timothy F. Cootes, Christopher J. Taylor, David H. Cooper, and Jim Graham. Active shape models–Their training and application. 61(1):38–59, 1995b. 110

Corinna Cortes and Vladimir Vapnik. Support-vector networks. volume 20, pages 273–297. Springer, 1995. 27, 28

D Cristinacce and TF Cootes. Feature detection and tracking with constrained local models. In *BMVC*, 2006. 22, 85

C.M. Cyr and B.B. Kimia. 3D Object Recognition Using Shape Similarity-Based Aspect Graph. In *Proc. ICCV*, pages 254–261, 2001. 20, 21, 67

Yuchao Dai, Hongdong Li, and Mingyi He. A simple prior-free method for non-rigid structure-from-motion factorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 17, 25, 109, 110, 118, 119

N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, pages I: 886–893, 2005a. 71

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005b. 87

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 2, 9

Piotr Dollár, Ron Appel, and Wolf Kienzle. Crosstalk cascades for frame-rate pedestrian detection. In *Proc. ECCV*. Springer, 2012. 9, 19, 43

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12: 2121–2159, 2011. 31

M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *IJCV*, 88(2):303–338, 2010. 3, 9, 41, 72, 102

Xiaochuan Fan, Kang Zheng, Youjie Zhou, and Song Wang. Pose locality constrained representation for 3d human pose reconstruction. In *Proceedings of the European Conference on Computer Vision*, 2014. 25, 109

Pedro F Felzenszwalb, Ross B Girshick, and David McAllester. Cascade object detection with deformable part models. In *Proc. CVPR*, pages 2241–2248. IEEE, 2010a. 9, 10, 18, 19, 41, 42, 44, 52

Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 32(9):1627–1645, 2010b. 3, 18, 23, 31, 40, 42, 44, 83, 86, 88

P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 32(9):1627–1645, 2010c. 9, 13, 21, 65, 68, 71

Sanja Fidler, Sven Dickinson, and Raquel Urtasun. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In *NIPS*, 2012. 23, 86

Francois Fleuret and Donald Geman. Coarse-to-fine face detection. *IJCV*, 2001. 19, 43

Amir Ghodrati, Marco Pedersoli, and Tinne Tuytelaars. Is 2d information enough for viewpoint estimation. In *BMVC*, 2014. 101, 102

Ross Girshick and Jitendra Malik. Training deformable part models with decorrelated features. In *ICCV*, 2013. 88

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 3, 83

D. Glasner, S.N. Vitaladevuni, and R. Basri. Contour-based joint clustering of multiple segmentations. In *Proc. CVPR*, pages 2385–2392, 2011a. 20, 67

Daniel Glasner, Meirav Galun, Sharon Alpert, Ronen Basri, and Gregory Shakhnarovich. Viewpoint-aware object detection and pose estimation. In *ICCV*, 2011b. 23, 86

Tom Goldstein and Stanley Osher. The split bregman method for l1-regularized problems. volume 2, pages 323–343. SIAM, 2009. 36

W.E.L. Grimson. *Object recognition by computer: The role of geometric constraints*. The MIT Press, Cambridge, MA, 1990. 20, 66, 83

C.H. Gu and X.F. Ren. Discriminative mixture-of-templates for viewpoint classification. In *ECCV*, pages V: 408–421, 2010a. 21, 68

Chunhui Gu and Xiaofeng Ren. Discriminative mixture-of-templates for viewpoint classification. In *ECCV*, 2010b. 96

Lie Gu and Takeo Kanade. 3D alignment of face in a single image. In *CVPR*, 2006. 22, 86, 92

Giovanni Gualdi, Andrea Prati, and Rita Cucchiara. Multistage particle windows for fast and accurate object detection. *PAMI*, 34(8):1589–1604, 2012. 19, 43

F. Han and S.C. Zhu. Bayesian reconstruction of 3D shapes and scenes from a single image. In *Int. Workshop on High Level Knowledge in 3D Modeling and Motion*, 2003. 20, 67

Qiang Hao, Rui Cai, Zhiwei Li, Lei Zhang, Yanwei Pang, Feng Wu, and Yong Rui. Efficient 2D-to-3D correspondence filtering for scalable 3d object recognition. In *Proc. CVPR*, 2013. 20, 67

Bharath Hariharan, Jitendra Malik, and Deva Ramanan. Discriminative decorrelation for clustering and classification. In *ECCV*. 2012. 23, 86, 87, 88

R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 7

G. Häusler and D. Ritter. Feature-based object recognition and localization in 3D-space, using a single video image. 73(1):64–81, January 1999. 20, 66

Mohsen Hejrati and Deva Ramanan. Analyzing 3d objects in cluttered images. In *NIPS*, 2012. 10, 23, 84, 86

S. Hinterstoisser, V. Lepetit, S. Ilic, P. Fua, and N. Navab. Dominant orientation templates for real-time detection of texture-less objects. In *Proc. CVPR*, pages 2257–2264, 2010. 77

R. Horaud. New methods for matching 3-D objects with single perspective views. *PAMI*, 9(3):401–412, May 1987. 20, 66

Berthold Klaus Paul Horn. *Robot Vision*. the MIT Press, 1986. 77

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators, 1989. 30

Edward Hsiao, Alvaro Collet, and Martial Hebert. Making specific features less discriminative to improve point-based 3D object recognition. In *Proc. CVPR*, pages 2653–2660. IEEE, 2010. 21, 68

Wenze Hu and S Zhu. Learning 3d object templates by quantizing geometry and appearance spaces. *PAMI*, 2014. 10, 23, 83, 86

Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3D human sensing in natural environments. *PAMI*, 36(7):1325–1339, 2014. 17, 24, 108, 110, 117, 120, 122

S. Izadi, R.A. Newcombe, D. Kim, O. Hilliges, D. Molyneaux, S. Hodges, P. Kohli, J. Shotton, A.J. Davison, and A. Fitzgibbon. Kinectfusion: real-time dynamic 3D surface reconstruction and interaction. In *ACM SIGGRAPH*, volume 23, 2011. 69

A. Jain, J. Tompson, M. Andriluka, G.W. Taylor, and C. Bregler. Learning human pose estimation features with convolutional networks. In *ICLR*, 2014. 24, 109

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. 89, 117

Hao Jiang. 3D human pose reconstruction using millions of exemplars. 2010. 24, 108

Hao Jiang, Stella X Yu, and David R Martin. Linear scale and rotation invariant matching. *PAMI*, 33(7):1339–1355, 2011. 23, 86, 95

Daniel Kahneman. *Thinking, fast and slow*. Macmillan, 2011. 4

Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Gaussian processes for object categorization. *IJCV*, 2010. 19, 44

Iasonas Kokkinos. Rapid deformable object detection using dual-tree branch-and-bound. In *NIPS*, 2011. 9, 18, 19, 23, 43, 44, 86

H.S. Koppula, A. Anand, T. Joachims, and A. Saxena. Semantic labeling of 3D point clouds for indoor scenes. In *NIPS*, 2011. 22, 68

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 27, 29, 31, 87, 88

K. Lai, L. Bo, X. Ren, and D. Fox. A scalable tree-based approach for joint object and pose recognition. 2011. 22, 68

Christoph H Lampert. An efficient divide-and-conquer cascade for nonlinear object detection. In *Proc. CVPR*. IEEE, 2010. 18, 43

Christoph H Lampert, Matthew B Blaschko, and Thomas Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. In *Proc. CVPR*, pages 1–8. IEEE, 2008. 18, 43

S. Lazebnik, A. Sethi, C. Schmid, D.J. Kriegman, J. Ponce, and M. Hebert. On pencils of tangent planes and the recognition of smooth 3D shapes from silhouettes. In *Proc. ECCV*, pages III: 651–665, 2002. 20, 67

Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989. 29

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. 27

Hsi-Jian Lee and Zen Chen. Determination of 3D human body postures from a single view. *Computer Vision, Graphics and Image Processing*, 30(2):148–168, 1985. 25, 106, 109

Alain Lehmann, Peter V Gehler, and Luc J Van Gool. Branch&rank: Non-linear object detection. In *BMVC*, 2011a. 19, 43

Alain Lehmann, Bastian Leibe, and Luc Van Gool. Fast prism: Branch and bound hough transform for object class detection. *IJCV*, 94(2):175–197, 2011b. 18, 43

Hongsheng Li, Junzhou Huang, Shaoting Zhang, and Xiaolei Huang. Optimal object matching via convexification and composition. In *ICCV*, 2011. 23, 86, 95

Sijin Li and Antoni B. Chan. 3D human pose estimation from monocular images with deep convolutional neural network. 2014. 24, 108

Sijin Li, Weichen Zhang, and Antoni B. Chan. Maximum-margin structured learning with deep networks for 3D human pose estimation. In *Proc. ICCV*, 2015. 24, 108, 118, 120, 122

J. Liebelt and C. Schmid. Multi-view object class detection with a 3D geometric model. In *Proc. CVPR*, pages 1688–1695, 2010. 20, 67

J. Liebelt, C. Schmid, and K. Schertler. Viewpoint-independent object class detection using 3D Feature Maps. In *Proc. CVPR*, pages 1–8, 2008a. 20, 67

Joerg Liebelt, Cordelia Schmid, and Klaus Schertler. Viewpoint-independent object class detection using 3d feature maps. In *CVPR*, 2008b. 10, 23, 83, 86

Joseph J Lim, Hamed Pirsiavash, and Antonio Torralba. Parsing ikea objects: Fine pose estimation. In *ICCV*, 2013. 23, 86

Yen-Liang Lin, Vlad I Morariu, Winston Hsu, and Larry S Davis. Jointly optimizing 3d model fitting and fine-grained classification. In *ECCV*, 2014. 10, 14, 23, 84, 85, 86, 95, 96

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proc. CVPR*, 2015. 116

Chien-Ping Lu, Gregory D Hager, and Eric Mjolsness. Fast and globally convergent pose estimation from video images. *PAMI*, 22(6):610–622, 2000. 124

João Maciel and João P Costeira. A global solution to sparse correspondence problems. *PAMI*, 25(2):187–199, 2003. 23, 86

T.B. Moeslund, A. Hilton, and V. Kruger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2-3): 90–126, 2006. 24, 108

Greg Mori and Jitendra Malik. Recovering 3D human body configurations using shape contexts. *PAMI*, 28(7):1052–1062, 2006. 24, 108

Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 2007. 113

Andrew Y. Ng and Michael I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 841–848. MIT Press, 2002. 27

Neal Parikh and Stephen Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):123–231, 2013. 32, 33, 34

Hyun Soo Park and Yaser Sheikh. 3D reconstruction of a smooth articulated trajectory from a monocular image sequence. In *Proc. ICCV*, pages 201–208, 2011. 25, 109

N. Payet and S. Todorovic. From contours to 3D object detection and pose estimation. In *Proc. ICCV*, pages 983–990, 2011. 21, 68

Marco Pedersoli, Andrea Vedaldi, and Jordi Gonzalez. A coarse-to-fine approach for fast deformable object detection. In *Proc. CVPR*, pages 1353–1360. IEEE, 2011. 19, 43

B. Pepik, M. Stark, P. Gehler, and B. Schiele. Teaching 3D geometry to deformable part models. In *CVPR*, pages 3362–3369, 2012a. 20, 67

Bojan Pepik, Peter Gehler, Michael Stark, and Bernt Schiele. 3d2pm–3d deformable part models. In *ECCV*, 2012b. 10, 23, 83, 86

Bojan Pepik, Michael Stark, Peter Gehler, and Bernt Schiele. Teaching 3D geometry to deformable part models. In *CVPR*, 2012c. 10, 83, 101, 102, 103

T. Pfister, J. Charles, and A. Zisserman. Flowing convnets for human pose estimation in videos. In *Proc. ICCV*, 2015. 24, 109, 117, 127

Harry Plantinga and Charles R Dyer. Visibility, occlusion, and the aspect graph. *International Journal of Computer Vision*, 5(2):137–160, 1990. 6

John Platt et al. Sequential minimal optimization: A fast algorithm for training support vector machines. 1998. 28

Victor Adrian Prisacariu, Aleksandr V Segal, and Ian Reid. Simultaneous monocular 2D segmentation, 3D pose recovery and 3D reconstruction. pages 593–606. Springer, 2013. 21, 67

Esa Rahtu, Juho Kannala, and Matthew Blaschko. Learning a category independent object detection cascade. In *Proc. ICCV*, 2011. 19, 44

Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Reconstructing 3d human pose from 2d image landmarks. In *ECCV*, 2012. 23, 25, 86, 109, 110

Frank Rosenblatt. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Technical report, DTIC Document, 1961. 29

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988. 29, 30

Radu Bogdan Rusu and Steve Cousins. 3D is here: Point cloud library (PCL). pages 1–4, 2011. 22, 68

R.B. Rusu, G. Bradski, R. Thibaux, and J. Hsu. Fast 3D recognition and pose using the viewpoint feature histogram. pages 2155–2162, 2010. 22, 68

Mathieu Salzmann and Raquel Urtasun. Implicitly constrained gaussian process regression for monocular non-rigid pose estimation. In *NIPS*, 2010. 24, 108

Benjamin Sapp, Alexander Toshev, and Ben Taskar. Cascaded models for articulated pose estimation. In *Proc. ECCV*. Springer, 2010. 9, 19, 43, 44

Silvio Savarese and Li Fei-Fei. 3D generic object categorization, localization and pose estimation. In *Proc. ICCV*, pages 1–8, 2007. 10, 21, 68

Silvio Savarese and Fei-Fei Li. 3d generic object categorization, localization and pose estimation. In *ICCV*, 2007. 23, 86

A. Sethi, D. Renaudie, D.J. Kriegman, and J. Ponce. Curve and surface duals and the recognition of curved 3D objects from their silhouettes. *IJCV*, 58(1):73–86, June 2004. 20, 67

Gregory Shakhnarovich, Paul A. Viola, and Trevor Darrell. Fast pose estimation with parameter-sensitive hashing. In *Proc. ICCV*, 2003. 24, 108

Jamie Shotton, Andrew W. Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *Proc. CVPR*, 2011. 16, 106

Leonid Sigal, Alexandru O. Balan, and Michael J. Black. HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 87(1-2):4–27, 2010. 125

Leonid Sigal, Michael Isard, Horst W. Haussecker, and Michael J. Black. Loose-limbed people: Estimating 3D human pose and motion using non-parametric belief propagation. *IJCV*, 98(1):15–48, 2012. 24, 108

Edgar Simo-Serra, Ariadna Quattoni, Carme Torras, and Francesc Moreno-Noguer. A Joint Model for 2D and 3D Pose Estimation from a Single Image. In *Proc. CVPR*, 2013. 125

Saurabh Singh, Abhinav Gupta, and Alexei A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012. URL http://arxiv.org/abs/1205.3137. 23, 86, 87

Cristian Sminchisescu. 3D human motion analysis in monocular video techniques and challenges. In *AVSS*, 2007. 24, 108

Cristian Sminchisescu and Bill Triggs. Kinematic jump processes for monocular 3D human tracking. In *Proc. CVPR*, 2003. 24, 108

Min Sun, Gary Bradski, Bing-Xin Xu, and Silvio Savarese. Depth-encoded hough voting for joint object detection and shape recovery. In *ECCV*, 2010. 23, 86

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 29

Raphael Sznitman, Carlos Becker, Francois Fleuret, and Pascal Fua. Fast object detection with entropy-driven evaluation. In *Proc. CVPR*, June 2013. 18, 19, 43, 44

C.J. Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. 80(3):349–363, 2000. 25, 109

Bugra Tekin, Xiaolu Sun, Xinchao Wang, Vincent Lepetit, and Pascal Fua. Predicting people's 3D poses from short sequences. *arXiv preprint arXiv:1504.08200*, 2015. 24, 108, 118, 120, 122, 124

T. Tieleman and G. Hinton. Rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning.Technical report*, 2012. 31

Jonathan J. Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *NIPS*, 2014. 24, 108, 109

A. Toshev, B. Taskar, and K. Daniilidis. Shape-based object detection via boundary structure segmentation. *IJCV*, 99(2):123–146, 2012. 13, 22, 65, 69, 72

Alexander Toshev and Christian Szegedy. DeepPose: Human pose estimation via deep neural networks. In *Proc. CVPR*, 2014. 24, 106, 108, 127

S. Ullman and R. Basri. Recognition by linear combinations of models. *PAMI*, 13:992–1006, 1991. 21, 68

Jack Valmadre and Simon Lucey. Deterministic 3D human pose estimation using rigid structure. In *Proc. ECCV*, 2010. 25, 109

Vladimir Naumovich Vapnik and Samuel Kotz. *Estimation of dependences based on empirical data*, volume 40. Springer-Verlag New York, 1982. 27, 28

S. Vijayanarasimhan and K. Grauman. Efficient region search for object detection. In *Proc. CVPR*, pages 1401–1408, 2011. 22, 69

M. Villamizar, H. Grabner, J. Andrade-Cetto, A. Sanfeliu, L. Van Gool, F. Moreno-Noguer, and KU Leuven. Efficient 3D object detection using multiple pose-specific classifiers. In *Proc. BMVC*, 2011. 20, 67

Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. CVPR*. IEEE, 2001. 19, 43

Chunyu Wang, Yizhou Wang, Zhouchen Lin, Alan L Yuille, and Wen Gao. Robust estimation of 3d human poses from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 17, 25, 109

David Weiss, Benjamin Sapp, and Ben Taskar. Structured prediction cascades. *arXiv preprint arXiv:1208.3279*, 2012. 9, 19, 43

Yu Xiang and Silvio Savarese. Estimating the aspect layout of object categories. In *CVPR*, 2012. 10, 23, 86

Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV*, 2014. 95, 96, 101, 102

Bruce Xiaohan Nie, Caiming Xiong, and Song-Chun Zhu. Joint action recognition and pose estimation from video. In *Proc. CVPR*, 2015. 106

Pingkun Yan, Saad M Khan, and Mubarak Shah. 3d model based object class detection in an arbitrary view. In *ICCV*, 2007. 23, 86

Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Proc. CVPR*, 2011. 10, 106, 126, 127, 128, 139

Tsz-Ho Yu, Tae-Kyun Kim, and Roberto Cipolla. Unconstrained monocular 3D human pose estimation by action detection and cross-modality regression forest. In *Proc. CVPR*, 2013. 24, 108

Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *Proc. ICCV*, 2013. 17, 110, 117, 118

Ziming Zhang, Jonathan Warrell, and Philip HS Torr. Proposal generation for object detection using cascaded ranking svms. In *Proc. CVPR*, pages 1497–1504. IEEE, 2011. 19, 43

Feng Zhou and Fernando De la Torre. Spatio-temporal matching for human detection in video. In *ECCV*, 2014. 23, 86

Feng Zhou and Fernando De la Torre. Spatio-temporal matching for human detection in video. In *Proc. ECCV*, 2014. 25, 107, 109

Xiaowei Zhou, Spyridon Leonardos, Xiaoyan Hu, and Kostas Daniilidis. 3d shape estimation from 2d landmarks: a convex relaxation approach. *CVPR*, 2015a. 92, 94

Xiaowei Zhou, Spyridon Leonardos, Xiaoyan Hu, and Kostas Daniilidis. 3D shape estimation from 2D landmarks: A convex relaxation approach. In *Proc. CVPR*, 2015b. 25, 109, 110, 115, 116

Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, and Kostas Daniilidis. Sparse representation for 3D shape estimation: A convex relaxation approach. *arXiv preprint arXiv:1509.04309*, 2015c. 25, 34, 109, 115, 116, 118

Menglong Zhu, Nilokay Atanasov, George Pappas, and Kostas Daniilidis. Active Deformable Part Models Inference. In *European Conference on Computer Vision*, 2014a. 17

Menglong Zhu, Konstantinos G Derpanis, Yinfei Yang, Samarth Brahmbhatt, Mabel Zhang, Cody Phillips, Matthieu Lecce, and Kostas Daniilidis. Single image 3d object detection and pose estimation for grasping. 2014b. 17

Menglong Zhu, Xiaowei Zhou, and Kostas Daniilidis. Single image pop-up from discriminatively learned parts. In *International Conference on Computer Vision*, 2015. 17

Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition*, pages 2879–2886. IEEE, 2012. 10

Yingying Zhu, Dong Huang, Fernando De la Torre Frade, and Simon Lucey. Complex non-rigid motion 3d reconstruction by union of subspaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014c. 25, 109

M Zeeshan Zia, Michael Stark, Bernt Schiele, and Konrad Schindler. Detailed 3d representations for object recognition and modeling. *PAMI*, 35(11):2608–2623, 2013. 10, 14, 23, 83, 85, 86, 92