Visual search and recognition of objects, scenes and people

Josef Sivic

Habilitation à diriger des recherches

École Normale Suprieure

Spécialité: Informatique

Josef Sivic

Visual search and recognition of objects, scenes and people

Rapporteurs:

M. Martial **HEBERT**	Carnegie Mellon University
M. David **LOWE**	University of British Columbia
M. Luc **VAN GOOL**	Eidgenössische Technische Hochschule Zürich

Membres du jury:

M. Francis **BACH**	INRIA
M. Patrick **PÉREZ**	Technicolor
M. Jean **PONCE**	ENS
Mme. Cordelia **SCHMID**	INRIA
M. Andrew **ZISSERMAN**	University of Oxford

Acknowledgements

I would like to thank my colleagues and collaborators who either directly contributed to, or indirectly influenced this work: Francis Bach, Leon Bottou, Ondrej Chum, Alyosha Efros, Bill Freeman, Michael Isard, Mark Everingham, Abhinav Gupta, Guillaume Obozinski, Tomas Pajdla, Bryan Russell, Cordelia Schmid, Akihiko Torii, Antonio Torralba, Andrew Zisserman, and specially Ivan Laptev and Jean Ponce. It has been a real pleasure to work with you.

I would also like to thank the reviewers of this manuscript David Lowe, Luc Van Gool and Martial Hebert as well as the jury members Francis Bach, Patrick Perez, Jean Ponce, Cordelia Schmid and Andrew Zisserman for their time and support.

Over the years, I had the chance to interact with many excellent students and post-docs. I would like to specially mention Karteek Alahari, Mathieu Aubry, Piotr Bojanowski, Vincent Delaitre, Carl Doersch, Olivier Duchenne, Jan Knopp, Petr Gronat, Maxime Oquab, James Philbin and Guillaume Seguin who significantly contributed to the research presented in this manuscript.

Finally, I would like to thank my family and my partner Jana for their endless support.

Contents

1 Introduction **1**
- 1.1 The objective . . . 1
- 1.2 Motivation . . . 2
- 1.3 Challenges . . . 2
- 1.4 Outline and contributions . . . 4

2 Representations for instance-level visual search **6**
- 2.1 Improving the bag-of-visual-words representation . . . 6
- 2.2 Leveraging the structure of image database . . . 8
- 2.3 Beyond bags-of-visual-words: Painting-to-3D Model Alignment . . . 11
- 2.4 Discussion . . . 16

3 Learning models for category-level recognition **17**
- 3.1 Unsupervised learning: discovering objects in image collections . . . 18
- 3.2 Weakly supervised learning of mid-level visual representations . . . 20
- 3.3 Transferring mid-level representations . . . 22
- 3.4 Beyond bounding boxes: predicting 3D models from images . . . 26
- 3.5 Discussion . . . 29

4 Modeling and recognition of people and their activities in video **30**
- 4.1 Learning person-specific classifiers from video and text . . . 30
- 4.2 Learning human actions from from video and text . . . 33
- 4.3 Joint learning of actors and actions in movies . . . 36
- 4.4 Discussion . . . 39

5 Discussion and outlook **40**

Chapter 1

Introduction

1.1 The objective

The objective of this work is to make a step towards an artificial system with human-like visual intelligence capabilities. We consider the following three visual recognition problems. First, we wish to identify the same object or scene instance in a large database of images despite significant changes in appearance due to viewpoint, illumination but also ageing, seasonal changes, or depiction style, as illustrated in figure 1.1(a). Second, we consider recognition of object classes such as "chairs" or "windows" (as opposed to a specific instance of a chair or a window). We wish to predict which object classes are present in the image, identify their locations as well as predict their approximate 3D model and fine-grained style ("Is this a bar stool or a folding chair?"; "Is this a bay window or a French window?"), as shown in figure 1.1(b). In particular, we investigate different levels of supervision for this task starting from just observing images without any supervision to having millions of labelled images or a set of full 3D models. Finally, we consider recognition of people and their actions in unconstrained videos such as TV or feature length films. In detail, we wish to identify individual people in the video using their faces ("Who is this?") as well as recognize what they do ("Is this person walking or sitting?"), as shown in figure 1.1(c). While these visual recognition tasks can be easily achieved by a human, they present a major challenge for a computer vision system.

(a) Instance-level recognition (b) Category-level recognition (c) People and actions

Figure 1.1: **Examples of visual recognition results described in this thesis.** (a) Aligning painting (left) to a 3D model of an architectural site (right), described in chapter 2. (b) We detect a 3D object ("chair") in the input image (left) as well a predict its fine-grained style and pose (right), as described in chapter 3. (c) Joint person identification and action recognition in video, described in chapter 4.

1.2 Motivation

Visual data is all around us. Our planet is covered by street-level imagery[1]; nearly 8 years worth of video content is uploaded to Youtube every day[2]; public archives such as INA[3] and the BBC capture and store all broadcasted content in France and the UK; and it is estimated that 4.2 million surveillance cameras capture visual data 24 hours a day in the UK alone[4]. Even more data will become available in the near future: cameras will be built in most manufactured cars[5] and (some) people will continuously capture their daily lives using wearable mobile devices such as Google Glass[6]. Breakthrough progress on the artificial visual intelligence capabilities would have major impact on our everyday lives as well as science and commerce. Example high-impact applications include:

Visual search of public archives. Imagine an automatic visual search through all public image archives to find historical imagery of a specific place over time for applications in archeology, history or architecture.

Robotics. The visual sensing capabilities of current robotics systems are still limited. Imagine a robot that can recognize and manipulate 3D objects to clean and maintain homes and public places; or that can understand human actions and gestures to interact with a child.

Surveillance and security. While surveillance cameras are wide-spread they have only very limited visual intelligence capabilities. Imagine an early warning system that can identify dangerous events in crowded scenes that may lead to disasters such as crowd panic or stampede. Or what if we could extract statistics of human behaviors over time across a city to help improve road safety, urban planning or commerce?

Personal and wearable cameras. Imagine "HowTo glasses" that help people achieve complex tasks such as change tires of a rented car or assemble Ikea furniture by recognizing individual parts as well as providing detailed visual instructions and feedback. Or what if we had "Expert glasses" that can make anyone an expert in a specific domain such as plant recognition or architecture by helping them identify different plant species or architecture elements and styles?

1.3 Challenges

The range of possible applications is exciting. However, there are major challenges that need to be addressed.

Appearance variation due to viewpoint, illumination, aging and depiction style. The imaged appearance of specific objects and scenes can vary depending on the viewpoint, scale, illumination and partial occlusion by other objects. Additional major sources of appearance variation are changes in the scene due to, e.g. season (snow coverage, coloring of trees) as well as aging and structural changes (building destroyed) over time. Finally, in case of non-photographic imagery

[1] Google Street-view or Bing maps

[2] http://www.youtube.com/t/faq

[3] Institute National Audiovisual – the French public audio and video archive.

[4] http://www.channel4.com/news/articles/society/factcheck+how+many+cctv+cameras/2291167.html

[5] http://kschang.hubpages.com/hub/How-Many-Cameras-Will-Your-Next-Car-Have-How-Cameras-are-Making-Cars-Smarter

[6] http://en.wikipedia.org/wiki/Project_Glass

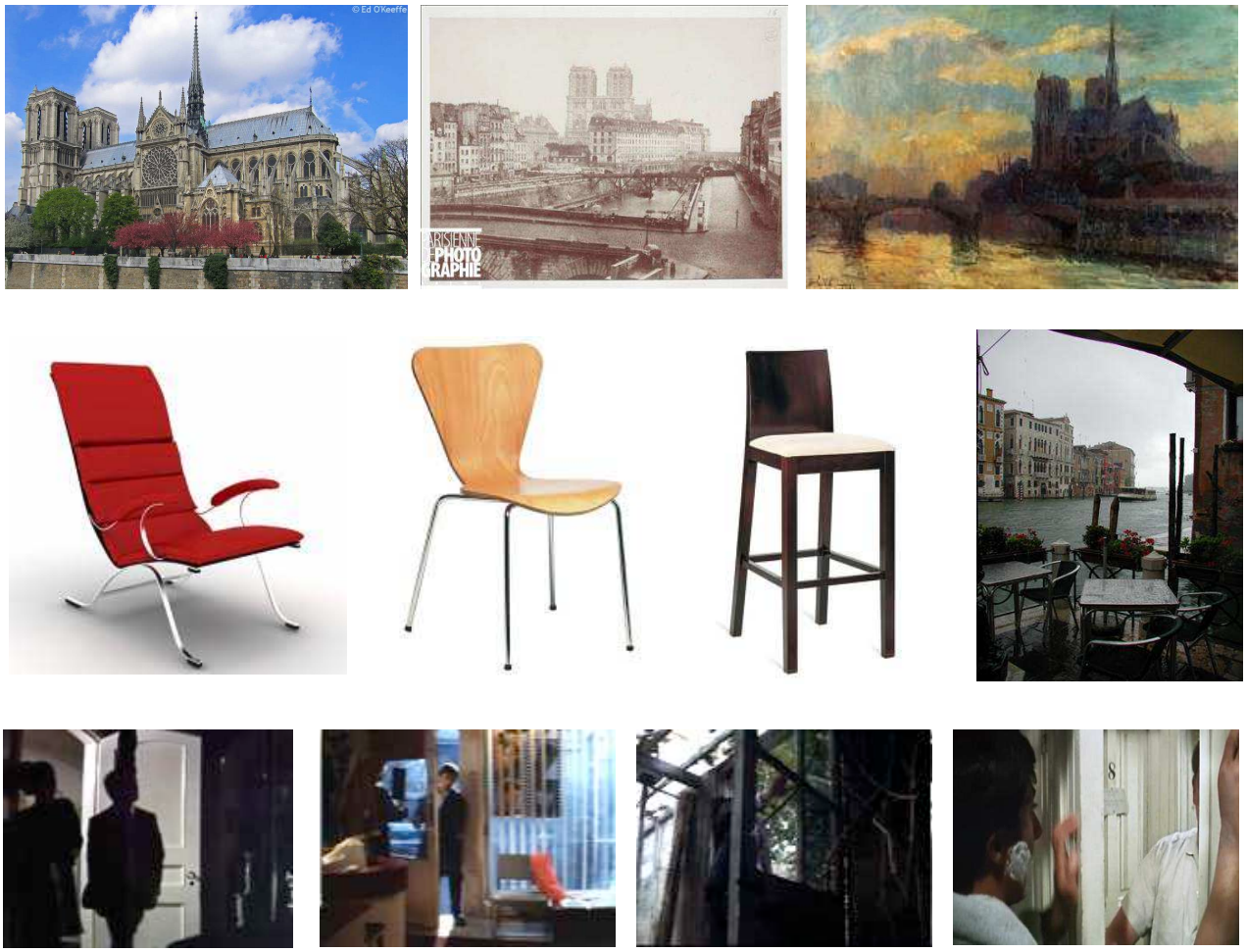


Figure 1.2: **Challenges of visual recognition. Top: Instance-level recognition.** The imaged appearance of the same place (the "Notre Dame" cathedral in Paris) varies significantly due to changes in viewpoint and illumination (left vs. middle), changes over time (e.g. destroyed buildings, left vs. middle) and depiction style (e.g. painting, right). **Middle: Category-level recognition.** Large intra-class variation in appearance of object class "chair". Right: objects (chairs and tables) are embedded in cluttered scenes with complex mutual occlusions. **Bottom: Dynamic scenes involving people.** Four instances a "person opening door" action. Note the huge variability in imaged appearance.

such as paintings or drawings, the appearance can be very different due to specific depiction style and drawing errors, as illustrated in figure 1.2(top).

Intra-class variation. For object classes, there is the additional difficulty of intra-class variation as illustrated in figure 1.2(middle). Moreover, objects are embedded in cluttered scenes and undergo complex mutual occlusions.

Variability of dynamic scenes involving people. For dynamic scenes involving people, there is the additional difficulty of modeling temporal information and the human (inter-)actions with the scene. The same action performed by two different people can result in a very different appearance in the video (see figure 1.2(bottom)). For example, the people may have different clothing, the camera viewpoint is different and the action is performed in a different manner, for example, faster or slower.

Available data and annotations. As discussed above the variation in appearance of objects and scenes is huge. Many currently successful methods address this challenge by designing trainable models that can be learnt from data using statistical machine learning. However, manually labeling training data is expensive. For example, common human actions such as hand shaking or kissing occur only a few times per movie on average, thus collecting a fair-sized number of action samples for training requires annotation of tens or hundreds of hours of video. Do we need to collect large amount of training data for every new visual recognition task? Can we learn from readily-available annotations such as shooting scripts for movies or GPS positions of images?

Granularity of representation. There are fundamental issues regarding the appropriate granularity of the visual representation. While, for example, natural text vocabulary and grammar are rather well defined, there is no accepted equivalent in the visual domain. Should person entering a car be considered as a single action or rather as a sequence of more basic action elements such as "opening a car door", "sitting in a car", etc. Should opening car door be considered the same action as opening door of the house? Similar issues arise in object and scene recognition. For modeling urban scenes, should all windows be grouped in a single class "window" or should we distinguish different window types, and if yes, how many?

Big data challenges. With the unprecedented amount of visual data becoming available, one of the key challenges becomes the computational and memory efficiency. What is the appropriate representation of images and videos that is compact, efficient, and yet sufficiently rich to enable accurate visual recognition?

1.4 Outline and contributions

This thesis addresses the above challenges and in the following we outline its main contributions. For each contribution we list the corresponding publication, the publication venue and, if applicable, the number of citations[7]. Related work is discussed in detail in each chapter.

The first chapter outlines several representations for **instance-level visual search**. First, in section 2.1 we discuss extensions improving the bag-of-visual-words model [178]. In particular, we describe scaling-up visual search to large image collections by building *approximate vocabularies* [141] (CVPR'07, 911 citations), improving recall using *query expansion* [41] (ICCV'07, 309 citations) and reducing quantization effects by *soft-assignment* [141] (CVPR'08, 440 citations), recently extended to repetitive structures in [189] (CVPR'13). Second, in section 2.2 we describe two image representations that leverage the *structure of the image database* to improve large-scale retrieval in image collections organized on a map [76, 99] (CVPR'13 and ECCV'10, 47 citations). This work is also related to structured graph-based image database representations [95, 144, 188] and learning more accurate descriptors for retrieval [143] omitted from this document due to space constraints. Finally, in section 2.3 we describe a new *mid-level image representation of 3D architectural sites* [14] (TOG'13) that is able to match historic photographs and non-photographic depictions across large changes in appearance, where standard local invariant features fail.

The second chapter investigates different models of **object classes** learnable from visual data with varying amounts of supervision. First, in section 3.1 we discuss a probabilistic generative model able to learn a visual hierarchy of objects together with their approximate segmentations in an *unsupervised* manner [177] (CVPR'08, 126 citations). This work builds on our initial efforts

[7]The citation counts were obtained from Google Scholar on 30 Nov 2013.

towards unsupervised modeling of image collections [154, 176] and is also related to our later work on energy-based data-driven image segmentation [156]. Second, in section 3.2 we describe a discriminative clustering model that incorporates *weak supervision* in the form of location labels automatically derived from GPS tags [54] (SIGGRAPH'12, 32 citations). We demonstrate that the model can be used to mine representative visual (architectural) elements automatically from a large dataset of street-level imagery. Third, in section 3.3 we investigate a *transfer learning* approach based on convolutional neural networks [135]. We demonstrate that a mid-level image representation learnt on a task with a large amount of fully labelled image data (ImageNet) can significantly improve visual recognition performance on related tasks where supervision is limited. Finally, in section 3.4 we describe an example-based *3D representation of object classes* with large intra-class variability (such as "chairs") that explicitly models the large variations in style and viewpoint [13]. We demonstrate prediction of 3D object models from 2D still images. This work is also related to our efforts (omitted due to space constraints) extending the range of possible visual recognition outputs from object labels and bounding boxes to more complex visual outputs such as prediction of future events [119], scene geometry [72], or scene appearance [96, 175].

The third chapter describes models of **people and their actions** that aim to overcome the major challenge of manually collecting annotated training video data and can be automatically learnt from video together with **readily-available** but weak **text supervision**. First, in section 4.1 we show that *person specific* face based *classifiers* can be learnt from video together with text annotation in the form subtitles and transcripts [60, 61, 174] (BMVC'06, IVC'08, CVPR'09, together 465 citations). Second, in section 4.2 we develop a discriminative clustering model for *temporal human action detection* in feature length movies [56] (ICCV'09, 77 citations) weakly supervised by text. Third, in section 4.3 we combine person identification and action recognition, and develop a joint model of actors and actions in movies, supervised by coarsely aligned movie scripts, that can *localize individual actors* in video *and recognize their action* [29] (ICCV'13) . This line of research is also related to our efforts (omitted due to space constraints) to develop novel mid-level representations of video that capture long-term temporal relations [114], localize and segment people [6], model person-object interactions [51, 52] or explicitly model density and motion of people in crowded scenes [151, 152].

Chapter 2

Representations for instance-level visual search

In this chapter we address the problem of visual search and recognition of particular objects and places. The key challenge is to develop a rich image representation that is able to deal with major changes in appearance of the object due to viewpoint, illumination, as well as, for example, changes of season, depiction style (drawing, painting or sketch) or structural changes (buildings destroyed) over time. At the same time the representation has to be compact to enable efficient search and recognition in large image collections. First, in section 2.1 we describe several extensions of the efficient bag-of-visual-words representation. Then, in section 2.2 we describe two image representations that leverage the structure of the image database to improve large-scale retrieval in image collections organized on a map. Finally, in section 2.3 we describe a compact representation of 3D scenes, where an entire architectural site is represented by a small set of discriminative visual elements that are automatically learnt from rendered views. We demonstrate the learnt 3D visual elements can be used to match and localize historical and non-photographic imagery where the standard image representations based on local invariant features fail.

2.1 Improving the bag-of-visual-words representation

In this section we breifly describe several extensions of the basic bag-of-visual-words system for large-scale retrieval [178]. We begin by scaling-up visual search to large image collections using *approximate vocabularies*, then discuss reducing quantization effects using *soft-assignment* and finally outline how to improve recall using *query expansion*.

Scaling-up: retrieval with large approximate vocabularies

In our initial work on object retrieval [178, 179], small vocabularies of 10K and 6K clusters were generated using k-means, which was sufficient to represent 5K-10K keyframes from one or two feature length movies. However, it was shown in [133, 141] that for large scale image / object retrieval a larger and more discriminative vocabulary is necessary. The time complexity of the k-means algorithm is $O(kN)$, where N is the number of data points and k is the number of cluster centers (visual words). Such a time complexity is feasible for small values of k, but renders the algorithm intractable for large vocabularies ($k > 10^5$). To address this issue we have developed an approximate k-means algorithm by replacing the exact nearest neighbour search of k-means by an

(a) Soft-assignment (b) Adaptive soft-assignment with repetitive structures

Figure 2.1: **(a) Illustration of soft-assignment of descriptors to multiple close-by visual words.** Points A - E represent cluster centers (visual words), and points 1–4 are features. Here we demonstrate two benefits of soft assignment: (i) In hard assignment, features 3 and 4 will never be matched as they are assigned to different visual words despite being close in descriptor space. Using soft-assignment, words 3 and 4 will be assigned to A, B and C (with certain weights) and can be matched strongly as they are close in the descriptor space; (ii) In hard assignment, features 1–3 are all assigned to word A equally and there is no way of distinguishing that 2 and 3 are closer than 1 and 3. Soft-assignment provides a way of recording this information, and subsequently giving more weight to the closer matches and less to the further. **(b) Adaptive soft-assignment for repetitive structures.** (Left) Groups of repeated image elements are identified in the image. Different groups of repeated patterns are shown in different colors. (Right) Visualization of the descriptor space. Points A – C represent cluster centres. Different occurrences of the same visual element (shown as red dots) are often quantized to different visual words (here A and C) naturally representing the noise in the description and quantization process as well as other non-modeled effects such as complex illumination or perspective deformation. Descriptors with many repetitions (red) are assigned each to only the nearest cluster center (red arrows). Descriptors with fewer or no repetitions (blue) are assigned to several near clusters in the feature space (blue arrows), here A, B and C.

approximate nearest neighbour search using a forest of KD-trees [128, 171]. This reduces the k-means time complexity to $O(N \log k)$ and enables creating vocabularies of 10^5 to 10^6 visual words. We have further demonstrated that the proposed approximate k-means (AKM) quantization [141] outperforms (measured by average precision in object retrieval) the alternative *hierarchical* k-means (HKM) [133] that solves a nested sequence of smaller clustering problems also reducing time complexity to $O(N \log k)$. This might be attributed to the quantization errors of the hierarchical k-means algorithm, where the data points can suffer (and never recover) from bad initial splits close to the root of the hierarchy. As a result, descriptors arising from the same object/scene region in different images are assigned to different clusters and not matched. While quantization errors seem to be more prominent for the HKM algorithm, they affect performance of any quantization method. In the following we describe two techniques that address this problem: the soft-assignment and query expansion.

Reducing quantization errors by soft-assignment

Visual words generated through descriptor clustering often suffer from quantization errors, where local feature descriptors that should be matched but lie close to the Voronoi boundary are incorrectly assigned to different visual words. To overcome this issue, we have proposed to soft-assign [142] each descriptor to several (typically 3) closest cluster centers in the feature space with weights set according to $\exp -\frac{d^2}{2\sigma^2}$, where d is the Euclidean distance of the descriptor from the cluster center and σ is a parameter of the method. The soft-assignment strategy is illustrated in figure 2.1 (a).

Details are given in [142], where we further show that soft-assignment improves recall in object retrieval at the moderate increase in computational and memory requirements. Recently, we have shown that this simple soft-assignment strategy can be further improved by taking the advantage of repeated features in the image, as described next.

Repeated structures such as building facades, fences or road markings often represent a significant challenge for object retrieval and specially large-scale place recognition applications [35, 47, 99, 146, 162]. Repeated structures are notoriously hard for establishing correspondences using multi-view geometry. Even more importantly, they violate the feature independence assumed in the bag-of-visual-words representation which often leads to over-counting evidence and significant degradation of retrieval performance. Previous methods treat repeated structures as nuisance and downweight their influence at the indexing stage [87, 99, 161, 164]. In contrast, in [189] we show that repetitions can be beneficial for retrieval and design an *adaptive soft-assignment* strategy, where we take advantage of the fact that multiple occurrences of repeated elements in the same image provide a natural and accurate soft-assignment of features to visual words. This is implemented by assigning the more frequently repeated visual words to fewer cluster centers, as illustrated in figure 2.1(b). Details of the procedure are given in [189] where we also show that the adaptive soft-assignment significantly outperforms the standard soft-assignment together with several other baseline methods in large-scale place recognition applications.

Query expansion

In the text retrieval literature a standard method for improving retrieval quality is query expansion [33]. A number of the highly ranked documents from the original query are reissued as a new query. In this way, additional relevant terms can be added to the query. This is a form of blind relevance feedback and it can fail if outlier (false positive) documents are included in the reissued query. In [40], we have brought query expansion into the visual domain. A strong spatial constraint between the query image and each result [141] allows for an accurate verification of each return, suppressing the false positives which typically ruin text-based query expansion. These verified images are then used to learn a latent feature model to enable controlled construction of expanded queries. The query enhancement adds additional visual words to the query, which were missed, e.g. due to noise in the detection/quantization process or extreme viewpoint and lighting changes, which are not modeled by the extracted image descriptors. The success of the method is based on two key elements: (i) The image database contains multiple images of the same object/place, some of which can be easily retrieved using the original query image; and (ii) a careful spatial filtering step ensures that non-relevant images/objects are not added to the query, effectively preventing polluting the query by non-relevant visual words. In [41] we show that visual query expansion can significantly improve retrieval results, and in particular recall. Example results before and after query expansion are shown in figure 2.2. Recently, several improvements to query expansion were proposed in [12, 38].

2.2 Leveraging the structure of image database

In object and image retrieval, discussed in the previous section, the database is typically an unstructured collection of images. However, in some applications and notably large scale visual place recognition, image databases are structured: images have geotags, are localized on a map or depict a consistent 3D world. Knowing or discovering the structure of the database can lead to significant

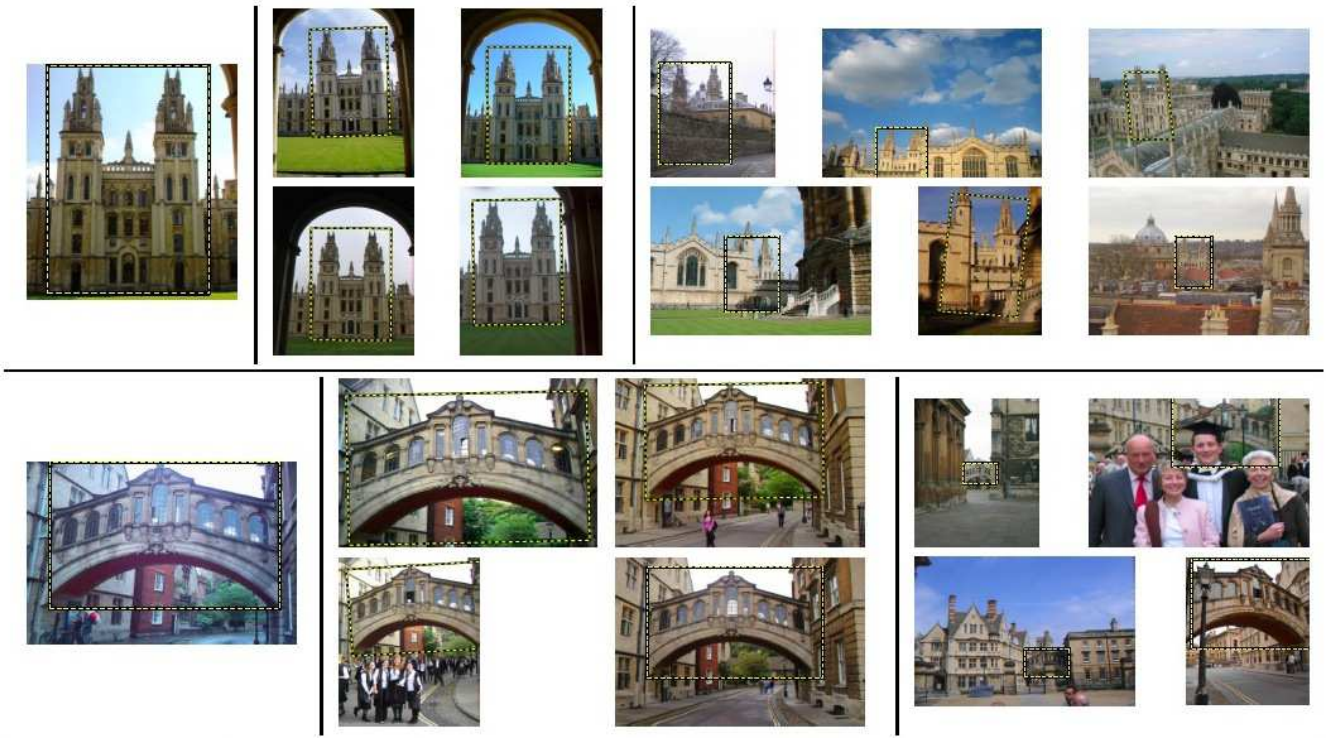


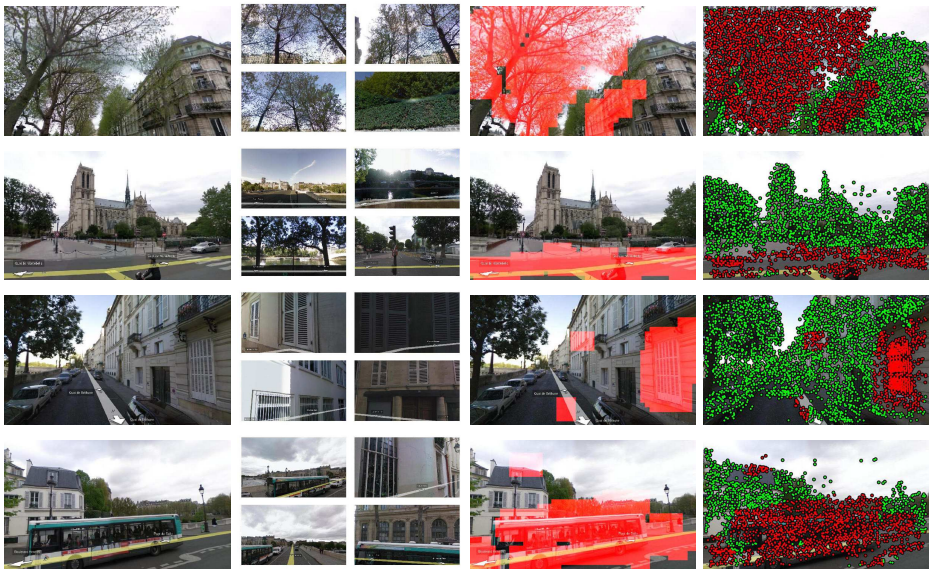
Figure 2.2: **Examples of large scale object retrieval with query expansion.** The image to the left shows the original query image with the outlined query region. The four images in the middle show the first four results returned by the bag-of-visual-words retrieval with spatial verification [141]. The images to the right show additional true positive images returned after query expansion [40]. Figure from [40]. Demo of the corresponding system can be found at: http://www.robots.ox.ac.uk/~vgg/research/oxbuildings/index.html.

improvements in both retrieval speed or accuracy. Examples include: (i) constructing an image graph [193], (ii) using the geotagged data as a form of supervision [162] to optimize the database for better recognition performance, or (iii) using the 3D structure of the scene to obtain accurate camera position [116, 117, 160], improve generalization over viewpoint [16, 85] or match across large changes in appearance [14].

In our work, we have explored image graphs to discover significant objects in Internet image collections [144] and to improve the accuracy of place recognition [188]. In section 2.3 we describe a method that uses the 3D structure of the scene to match across large changes in appearance where representations based on local invariant features fail. In the remainder of this section we discuss two methods [76, 99] that use geotags as a form of supervision to learn features characteristic for a place.

Avoiding confusing features in place recognition

In [99], we develop a method for automatic detection of image-specific and spatially-localized groups of confusing features, and demonstrate that suppressing them significantly improves place recognition performance while reducing the database size. In particular, we detect, in each database image, spatially localized groups of local invariant features, which are matched to images far from the geospatial location of the database image. The result is a segmentation of each image into a confusing layer, represented by groups of spatially localized invariant features occurring at other



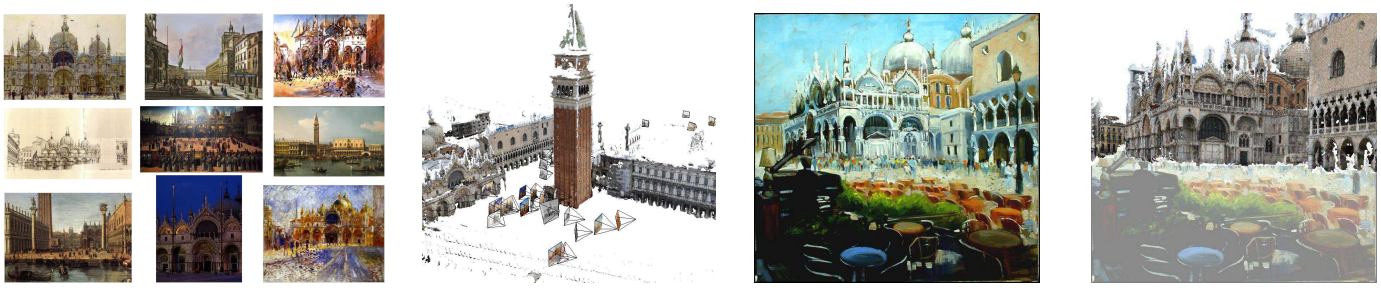
(a) (b) (c) (d)

Figure 2.3: **Detecting and removing confusing objects for place recognition.** Examples of detected confusing regions which are obtained by finding local features in original image (a) frequently mismatched to similar images of different places shown in (b). (c) Detected confusing image regions. (d) Features within the confusing regions are erased (red) and the rest of features are kept (green). Note that confusing regions are spatially localized and fairly well correspond to real-world objects, such as trees, road, bus or a window blind.

places in the database, and a layer characteristic for a place, as illustrated in figure 2.3. Further, we demonstrate that suppressing such confusing features, while keeping the characteristic features for each image, significantly improves place recognition performance while reducing the database size. Details and results are in [99].

Learning per-location classifiers for visual place recognition

In [99] we have manually designed a criteria, which features should be removed, and which features should be kept in each database image. This approach can be further improved by designing an appropriate objective function. In particular, in our recent work [76] we cast place recognition as a classification task and use the available geotags to train a classifier for each location in the database in a similar manner to per-exemplar SVM [124] in object recognition. This is beneficial as each classifier can learn, which features are discriminative for a particular place. The classifiers are learnt offline. At query time, the query photograph is localized by transferring the GPS tag of the best scoring location classifier. While learning classifiers for each place is appealing, calibrating outputs of the individual classifiers is a critical issue. In object recognition [124], it is addressed in a separate calibration stage on a held-out set of training data. This is not possible in the place recognition set-up as only a small number, typically one to five, of positive training images are available for each location (e.g. street-view images viewing the same building facade). To address this issue, we have developed a calibration procedure inspired by the use of p-values in statistics and based on ranking the score of a query image amongst scores of other images in the database. Detailed results of the proposed place recognition approach are in [76] and demonstrate that this method outperforms the hand-designed criteria for selecting characteristic features for each place described in our previous work [99].



(a) Internet paintings. (b) Painting viewpoints. (c) Aligned painting to 3D model.

Figure 2.4: Our method automatically aligns and recovers the viewpoint of paintings, drawings, and historical photographs to a 3D model of an architectural site.

2.3 Beyond bags-of-visual-words: Painting-to-3D Model Alignment

In this section we describe a technique that can reliably align arbitrary 2D depictions of an architectural site, including drawings, paintings and historical photographs, with a 3D model of the site as illustrated in figure 2.4. Why is this task important? First, non-photographic depictions are plentiful and comprise a large portion of our visual record. We wish to reason about them, and aligning such depictions to reference imagery (via a 3D model in this case) is an important step towards this goal. Second, such technology would open up a number of exciting applications that currently require expensive manual alignment of 3D models to various forms of 2D imagery. Examples include interactive visualization of a 3D site across time and different rendering styles [49, 113], model-based image enhancement [100], annotation transfer for augmented reality [180], inverse procedural 3D modeling [7, 130] or computational re-photography [148, 19]. Finally, reliable automatic image to 3D model matching is important in domains where reference 3D models are often available, but may contain errors or unexpected changes (e.g. something built/destroyed) [30], such as urban planning, civil engineering or archaeology.

Related work. *Local invariant features* and descriptors such as SIFT [121] represent a powerful tool for matching photographs of the same at least lightly textured scene despite changes in viewpoint, scale, illumination, and partial occlusion. Example applications include 3D reconstruction [180], image mosaicing [183], visual search [178], visual localization [163], and camera tracking [20] to list but a few. Large 3D scenes, such as a portion of a city [116], can be represented as a 3D point cloud with associated local feature descriptors extracted from the corresponding photographs [160]. However, appearance changes beyond the modeled invariance, such as significant perspective distortions, non-rigid deformations, non-linear illumination changes (e.g. shadows), weathering, change of seasons, structural variations or a different depiction style (photograph, painting, sketch, drawing) cause local feature-based methods to fail [81, 155, 170]. Greater insensitivity to appearance variation can be achieved by matching the geometric or symmetry pattern of local image features [37, 81, 168], rather than the local features themselves. However, such patterns have to be detectable and consistent between the matched views.

Contour-based 3D to 2D alignment methods [122, 84] rely on detecting edges in the image and aligning them with projected 3D model contours. Such approaches are successful if object contours can be reliably extracted both from the 2D image and the 3D model. A recent example is the work

Figure 2.5: **Matching as classification. (a)** Given a region and its HOG descriptor q in a rendered view (top left) the aim is to find the corresponding region in a painting (top right). This is achieved by training a linear HOG-based sliding window classifier using q as a single positive example and a large number of negative data. The classifier weight vector w is visualized by separately showing the positive (+) and negative (-) weights at different orientations and spatial locations. The best match x in the painting is found as the maximum of the classification score. **(b)** The entire architectural site is summarized by a set of mid-size 3D discriminative visual elements that are used to find correspondences between the input scene depiction (left) and the 3D model (right).

on photograph localization using semi-automatically extracted skylines matched to clean contours obtained from rendered views of digital elevation models [17, 16]. Contour matching was also used for aligning paintings to 3D meshes reconstructed from photographs [155]. However, contours extracted from paintings and real-world 3D meshes are noisy. As a result, the method requires a good initialization with a close-by viewpoint. In general, reliable contour extraction is a hard and yet unsolved problem.

Modern image representations developed for *visual recognition*, such as HOG descriptors [48], represent 2D views of objects or object parts [67] by a weighted spatial distribution of image gradient orientations. The weights are learnt in a discriminative fashion to emphasize object contours and de-emphasize non-object, background contours and clutter. Such a representation can capture complex object boundaries in a soft manner, avoiding hard decisions about the presence and connectivity of imaged object edges. Learnt weights have also been shown to emphasize visually salient image structures matchable across different image domains, such as sketches and photographs [170]. Similar representation has been used to learn architectural elements that summarize a certain geo-spatial area by analyzing (approximately rectified) 2D street-view photographs from multiple cities [54].

Building on discriminatively trained models for object detection, we develop a compact representation of 3D scenes suitable for alignment to 2D depictions. In contrast to [54, 170] who analyze 2D images, our method takes advantage of the knowledge and control over the 3D model to learn a set of mid-level 3D scene elements robust to a certain amount of viewpoint variation and capable of recovery of the (approximate) camera viewpoint. We show that the learnt mid-level scene elements are reliably detectable in 2D depictions of the scene despite large changes in appearance and rendering style.

Approach overview. The key idea of this work is to formulate image matching as a discriminative classification task. The aim is to match a given rectangular image patch q (represented by a HOG descriptor [48]) in a rendered view to its corresponding image patch in the painting, as illustrated in figure 2.5. Instead of finding the best match measured by the Euclidean distance between the descriptors, we train a linear classifier with q as a single positive example (with label $y_q = +1$) and a large number of negative examples x_i for i =1 to N (with labels $y_i = -1$). The matching is then performed by finding the patch x^* in the painting with the highest classification score

$$s(x) = w^\top x + b, \tag{2.1}$$

where w and b are the parameters of the linear classifier. Compared to the Euclidean distance, the classification score (2.1) measures a form or similarity, i.e. a higher classification score indicates higher similarity between x and q. In addition, the learnt vector w weights each component of x differently. This is in contrast to the standard Euclidean distance where all components of x have the same weight. Note that a similar idea was used in learning per-exemplar distances [73] or per-exemplar SVM classifiers [124] for object recognition and cross-domain image retrieval [170]. Here, we build on this work and apply it to image matching using mid-level image structures.

Parameters w and b are obtained by minimizing a cost function of the following form

$$E\left(w, b\right) = L\left(1, w^T q + b\right) + \frac{1}{N}\sum_{i=1}^{N} L\left(-1, w^T x_i + b\right), \tag{2.2}$$

where the first term measures the loss L on the positive example q (also called "exemplar") and the second term measures the loss on the negative data. Note that for simplicity we ignore in (2.2) the regularization term $||w||^2$, but the regularizer can be easily added in a similar manner to [18, 74]. A particular case of the exemplar based classifier is the exemplar-SVM [124, 170], where the loss $L(y, s(x))$ between the label y and predicted score $s(x)$ is the hinge-loss $L(y, s(x)) = \max\{0, 1-ys(x)\}$ [25]. For exemplar-SVM cost (2.2) is convex and can be minimized using iterative algorithms [63, 166]. Note that the optimal value of the cost (2.2) characterizes the separability of a particular candidate visual element q from the (fixed) negative examples $\{x_i\}$ and hence can be used for measuring the degree of discriminability of q. However, when using a hinge-loss as in exemplar SVM, optimizing (2.2) would be expensive to perform for thousands of candidate elements in each rendered view. Instead, similarly to [18, 74], we take advantage of the fact that in the case of square loss $L(y, s(x)) = (y - s(x))^2$ the w_{LS} and b_{LS} minimizing (2.2) and the optimal cost E^*_{LS} can be obtained in closed form. In turn, this enables efficient training of candidate visual element detectors corresponding to image patches that are densely sampled in each rendered view. Only visual elements q with low training cost (2.2), i.e. those that are *discriminative* w.r.t. the generic set of background patches are retained. As a result the entire architectural site is represented by a small set of discriminative visual element detectors $\{q\}$ learnt from rendered views. At query time, the element detectors are used to establish correspondences between the 3D model and the input depiction as illustrated in figure 2.5(b). In turn those correspondences are then used to recover the approximate viewpoint of the painting with respect to the 3D model. Details are given in [14].

Results. We have collected a set of human-generated 3D models from Google Sketchup for the following architectural landmarks: Notre Dame of Paris, Trevi Fountain, and San Marco's Basilica. The Sketchup models for these sites consist of basic primitive shapes and have a composite texture

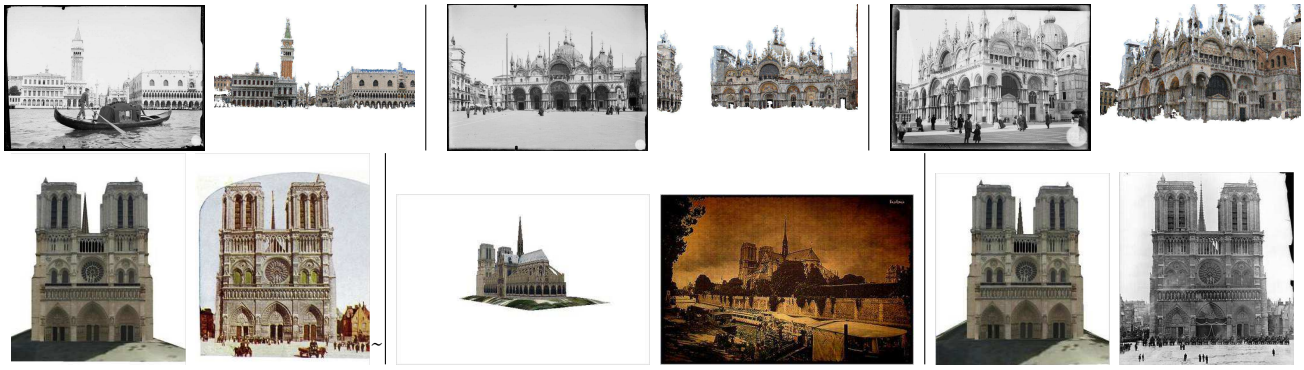


Figure 2.6: Alignment of historical photographs of San Marco's Square (top) and Notre Dame of Paris (bottom) to their respective 3D models.

	Good match	Coarse match	No match
(i) SIFT on rendered views	40%	26%	33%
(ii) Viewpoint retrieval [155]	1%	39%	60%
(iii) Exemplar SVM [170]	34%	18%	48%
(iv) mid-level painting visual elements	33%	29%	38%
3D discrim. visual elements (ours)	**51%**	**21%**	**28%**

Table 2.1: Viewpoint similarity user study – comparison with baselines on the "San Marco Square" 3D site.

from a set of images. In addition to the Sketchup models, we also consider one of the Rome-in-a-day [3] 3D models of San Marco's Square that was reconstructed from a set of photographs using dense multi-view stereo. To test the developed method we have collected from the Internet 85 historical photographs and 252 non-photographic depictions (watercolors, oil paintings, pastels, drawings and engravings) of the sites.

Figures 2.6 and 2.7 show example alignments of historical photographs and non-photographic depictions, respectively. Notice that the depictions are reasonably well-aligned, with regions on the 3D model rendered onto the corresponding location for a given depiction. We are able to cope with a variety of viewpoints with respect to the 3D model as well as different depiction styles. Our approach succeeds in recovering the approximate viewpoint in spite of these challenging appearance changes and the varying quality of the 3D models. Figure 2.8 shows the camera frusta for the recovered approximate painting viewpoints. Notice that our system is able to recover viewpoints that are to the rear of the main facade of the Notre Dame cathedral, which has not been possible in prior work [180] due to the lack of reconstructed structure in these areas.

To quantitatively evaluate the goodness of our alignments, we have conducted a user study via Amazon Mechanical Turk. The workers were asked to judge the viewpoint similarity of the resulting alignments to their corresponding input depictions by categorizing the viewpoint similarity as either a (a) Good match, (b) Coarse match, or (c) No match. We asked five different workers to rate the viewpoint similarity for each depiction and we report the majority opinion. Table 2.1 compares the performance of our algorithm to several baseline methods for the 141 depictions of San Marco Square – the largest 3D model in our dataset with 45K sampled viewpoints. We compare our algorithm against the following four baselines: (i) SIFT on rendered views, (ii) viewpoint retrieval

Figure 2.7: Example alignments of non-photographic depictions to 3D models. Notice that we are able to align depictions rendered in different styles and having a variety of viewpoints with respect to the 3D models.

(a) Notre Dame of Paris. (b) Trevi Fountain.

Figure 2.8: Google Sketchup models and camera frusta depicting the recovered viewpoints of the paintings.

(corresponding to the coarse alignment step of [155]), (iii) exemplar SVM [170], and (iv) mid-level painting visual elements that, similar to [172], learns mid-level visual elements directly from paintings, rather than the 3D model. The implementation details of each baseline are given in [14]. Our method significantly outperforms all baselines. Note that the (i) 'SIFT on rendered views' baseline is similar in spirit to matching with Viewpoint Invariant Patches (VIP) [200], except no depth or rectification is needed for the paintings. This is the best performing baseline, having 40% good alignments compared with 51% for our algorithm. Note that the good performance is largely due to alignments of historical photographs (70% vs. 50% for our method). However, if historical photographs are removed from the dataset, the SIFT on rendered views baseline drops to 27% good alignments, while our algorithm still achieves 52% good alignments.

2.4 Discussion

In this chapter, we have introduced several extensions of the bag-of-visual-words model and shown their benefits for large-scale object retrieval. Next we plan to scale-up the search to very large image collections. For example, we estimate that France alone is covered by more than 60 million street-view images. Building on our work an the recently developed aggregated image codes [88, 89], can we learn *a compact but discriminative* image representation for *planet scale* place recognition? Finally, we have introduced a mid-level representations of 3D models that can be matched across large changes in appearance where the standard local features fail. Designing an efficient and compact *mid-level indexing* technique for this new powerful representation remains an exciting open problem.

Chapter 3

Learning models for category-level recognition

In this chapter we address the problem of category-level object recognition. Compared to instance-level recognition discussed in the previous chapter, the main additional challenge is modeling the intra-class appearance and shape variation. For example, what is the appropriate model of the various appearances and shapes of "chairs"? This challenge is usually addressed by designing some form of a parametric model of the object's appearance and shape. The parameters of the model are then learnt from a set of instances using statistical machine learning techniques. In turn, availability of appropriate training data becomes a critical issue. In this chapter we discuss several models of object classes which require different types and amounts of supervision. In section 3.1, we describe an *unsupervised* latent generative model, which learns a visual hierarchy of objects together with their approximate segmentation from an unlabeled image collection. While unsupervised learning is appealing, the output is often limited only to frequently occurring and visually consistent objects. At the same time, many image collections contain some form of readily-available annotations such as text or geotags, which is often ignored in unsupervised learning. To address these limitations, in section 3.2, we describe a discriminative clustering model that learns object models from images and incorporates the readily available supervision as constraints. We go even further in section 3.3 and investigate whether an object model learnt on one visual recognition task, where large amount of fully supervised data is available, can be transferred to another task where supervision is relatively scarce. This is interesting as large datasets of millions of images labelled with thousands of object classes are now becoming available [53]. However, collecting fully labelled datasets of similar scale for any new given task may be difficult. Finally, in section 3.4 we introduce an object model that requires as input not only object labels and their locations, but a *library of 3D models* of hundreds of instances of the object class spanning the variability of shapes and styles. As a result, recognition can proceed despite large changes in viewpoint and the output is a 3D model aligned with the test image rather than just the object label or the object bounding box.

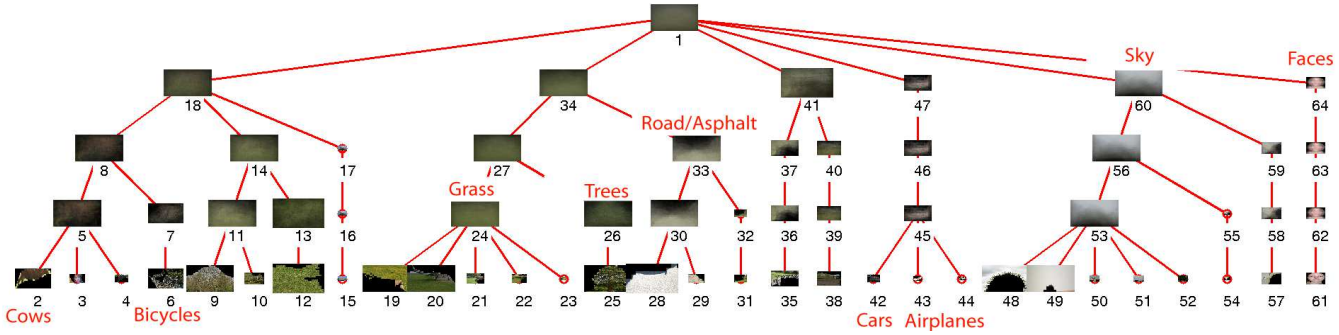


Figure 3.1: **A 5-level hLDA hierarchy learned on the MSRC-B1 dataset.** Branches with less than 5 image segments (in distinct images) are removed from the tree. Branches in the tree were manually labelled by object class names (shown in red) based on visual inspection. Node numbers are shown in black below each node. Note that, for example, 'grass' and 'trees' or 'cars' and 'airplanes' appear close together in the hierarchy. Some of the discovered topics are shown in more detail in figure 3.2. Note that 8 (out of 9) object classes are discovered. We do not find any building topics as buildings seem to have less consistent segmentations across the dataset. Using automatic segmentations enables discovering new object classes not labelled in the data (here a 'road/asphalt' topic, node 28, shown in figure 3.2).

3.1 Unsupervised learning: discovering objects in image collections

In this section we pose the following question: given a collection images, "Is it possible to learn visual object classes simply from looking at images?" That is, if our data set contains many instances of (visually similar) object classes, can we discover these object classes, identify their segmentation in the image and organize the objects in a hierarchical structure based on their visual appearance without any supervision? As discussed in section 1.3, training data is essential for many machine vision tasks, including object categorization and scene recognition. The information used for training can be labelled or unlabelled. In the case of labelled data, objects or their properties are given along with the original visual data. This is the most useful form of training data, but is also the most expensive to obtain, and the quantities of such datasets are often quite limited. Hand-labelled data will include any biases or mistakes on the part of the labellers. Moreover, as discussed in section 1.3, recent large-scale object labeling efforts [59, 157, 199] have demonstrated the difficulties in deciding on the granularity of the categories to be labeled. For example, if cars and buses are two separate categories, shouldn't commercial and military airplanes be separated as well? A categorization or labelling of the world thought up by one person may not in fact be the most useful for training a machine how to see. In contrast, an unlabelled training set comes virtually free; one only needs to point a camera out at the world to obtain an unlimited supply of training images. There has been research interest in learning from unlabelled data, including unsupervised algorithms for object categorization [75, 176, 112] and segmentation [112, 154, 186]. Unsupervised learning has been also studied as a way to initialize large convolutional neural networks, as discussed in more detail in section 3.3.

In general, learning from unlabelled data will be much slower and more difficult than for labelled data. However, working with unlabelled data can bring benefits. One might hope to learn common structures or organizations of the visual world by analyzing unlabelled collections of images. A hierarchy is a natural structure to consider. While objects in the world can be arranged into a hierarchy based on their semantic meaning (e.g. organism – animal – feline – cat). Here we consider

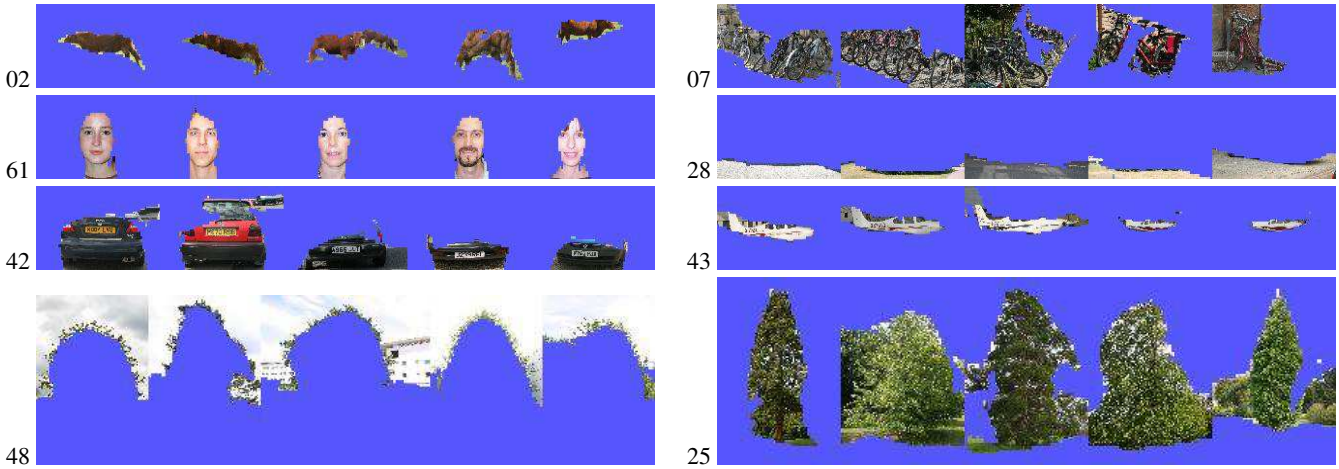


Figure 3.2: **Unsupervised learning of objects and their segmentation from images.** Selected nodes from the hierarchy, shown in figure 3.1. Each node is illustrated by a montage of the top five segments. Node numbers, referring to figure 3.1, are shown to the left of each montage. Note that the hierarchy of objects and their segmentation were automatically learned from an unlabelled set of images without supervision.

visual hierarchy, that is we ask the question what is the visual hierarchy of the objects we see in the world?

Related work. Vision researchers have used hierarchical models for visual object [22, 70, 182, 197] and scene categories [182, 195], but mainly in a supervised or semi-supervised setting, where object labels for (at least some) images are available. Unsupervised learning has been restricted mainly to part hierarchies for individual object categories [58, 91, 134, 206]. In the context of supervised object category recognition and detection, object/part hierarchies have been shown to improve generalization for small sample sets by sharing features/parts between objects [21, 182, 191]. Combining classifiers learnt from images at different levels of a (handcrafted) object hierarchy was shown to improve object classification performance [208]. An object hierarchy, learnt in an unsupervised way from a small set of images, was shown to improve supervised classification and object segmentation in unseen images [5]. Hierarchical models have been also applied to unsupervised learning of visual scene categories [23]. Multi-level (deep) hierarchical representations are also common in the neural networks literature, which we review in more detail in section 3.3.

Approach overview. We build on a hierarchical model developed for text analysis – the hierarchical Latent Dirichlet Allocation (hLDA) [27]. This model is a generalization of the (flat) LDA [28] model. Like LDA, it generates a document as a superposition of topics, but in hLDA the topics are composed during a path through a tree becoming ever more specialized from root to leaf. The great merit of the hLDA model is that both the topics and the *structure* of the tree are learnt from the training data – it is not necessary to specify the structure of the tree in advance. In this work we investigate whether the hLDA model can be adapted for discovering object hierarchies in the visual domain. Images are modeled using quantized densely sampled local image regions as analogues to words in text. Details of the model and the learning procedure are given in [177].

Results. We applied our hierarchical model to the the MSRC-B1 dataset (240 images, 9 object classes). First, building on our previous work [154] multiple overlapping segmentations of each im-

age are obtained by varying parameters of a bottom-up segmenter based on Normalized Cuts [169]. Second, object categories (and their rough segmentations) are learnt, in an unsupervised way, by finding image segments consistently segmented throughout the dataset using the hLDA topic discovery model, where each image segment is treated as a separate 'document'. The resulting 5-level object hierarchy is shown in figure 3.1. The top 5 segments for selected nodes in the tree are shown in figure 3.2. Further details of the experiments and the quantitative evaluation are given in [177]. The quantitative results demonstrate that the hierarchical model outperforms the flat LDA topic model [154] in the quality of resulting object segmentations.

3.2 Weakly supervised learning of mid-level visual representations

In the previous section, we have demonstrated that generative probabilistic latent models from text analysis can be applied to unsupervised discovery of objects in image collections. However, the method suffers from several drawbacks. First, due to the expensive sampling-based learning procedure the model is hard to scale to large datasets. Second, being unsupervised, this model (together with other related unsupervised object discovery methods [154, 39, 97, 111, 112, 172, 146]) is limited to discovering only objects that are both very common and highly visually consistent.

In contrast, here we propose an object discovery model that uses a *discriminative* cost that is further *constrained* by readily available, though only weak (image-level), supervision. We apply the model to a dataset of street-level imagery from Google street-view using the readily available supervision in the form of location labels automatically derived from GPS tags. We demonstrate that the model can be used to mine representative visual (architectural) elements automatically from this large online image dataset. Not only are the resulting visual elements geographically discriminative (i.e. they occur only in a given locale), but they also typically look meaningful to humans, making them suitable for a variety of geo-data visualization applications.

Approach overview. Our goal is to discover visual elements which are characteristic of a given geographical locale (e.g. the city of Paris). That is, we seek patterns that are both *frequently occurring* within the given locale, **and** *geographically discriminative*, i.e. they appear in that locale and do not appear elsewhere. Note that neither of these two requirements by itself is enough: sidewalks and cars occur frequently in Paris but are hardly discriminative, whereas the Eiffel Tower is very discriminative, but too rare to be useful ($< 0.0001\%$ in our data). In this work, we will represent visual elements by square image patches at various resolutions, and mine them from our large image database. The database will be divided into two parts: (i) the positive set containing images from the location whose visual elements we wish to discover (e.g. Paris); and (ii) the negative set containing images from the rest of the world (in our case, the other 11 cities in the dataset). We assume that many frequently occurring but uninteresting visual patterns (trees, cars, sky, etc.) will occur in both the positive and negative sets, and should be filtered out. Our biggest challenge is that the overwhelming majority of our data is uninteresting, so matching the occurrences of the rare interesting elements is like finding a few needles in a haystack.

More formally, we are given a large set of N image patches represented by their (HOG [48]) descriptors $\mathbf{x}_i$, $i = 1, \ldots, N$ together with a binary annotation y_i for each patch whether the patch belongs to the positive set (a particular city of interest) or negative set (other cities). The aim is to find a mapping $\hat{z}_i = f(\mathbf{x}_i)$ that will output for each patch a discrete visual element label $\hat{z}_i$ from

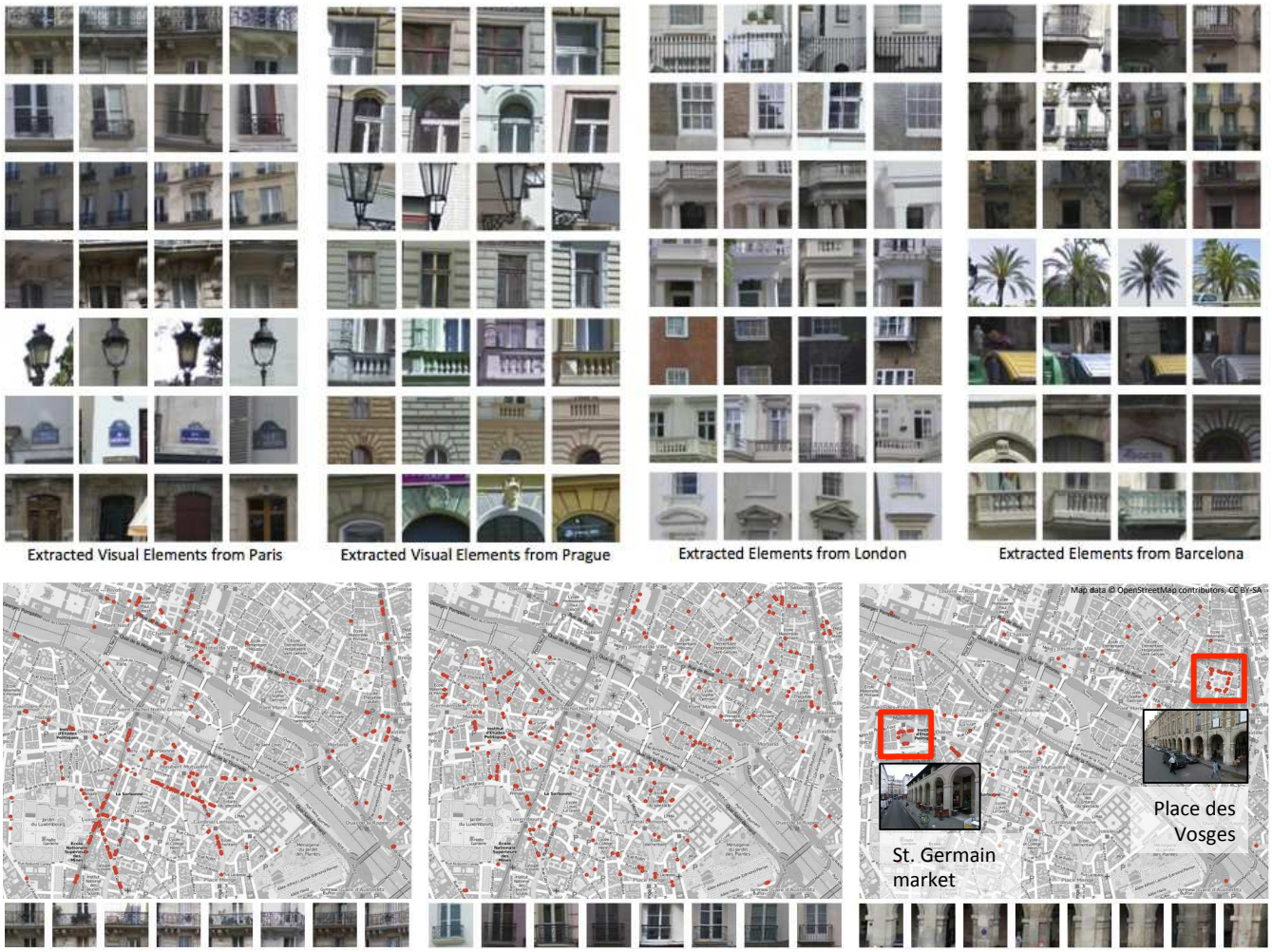


Figure 3.3: **Weakly-supervised learning of architectural elements from geotagged street-view images [54]. Top:** Examples of architectural visual elements characteristic for Paris, Prague, London and Barcelona automatically learnt by analyzing thousands of Street-view images. **Bottom:** Examples of geographic patterns in Paris (shown as red dots on the maps) for three discovered visual elements (shown below each map). Balconies with cast-iron railings are concentrated on the main boulevards (left). Windows with railings mostly occur on smaller streets (middle). Arch supporting columns are concentrated on Place des Vosges and the St. Germain market (right).

a "visual vocabulary" of Z elements or "background". These labels will correspond to the different discovered architectural elements or objects. This is achieved by solving the following optimization problem

$$\min_{f,\mathbf{z}} \quad \sum_{i=1}^{N} \ell(z_i, f(\mathbf{x}_i)) + \Omega(f) \tag{3.1}$$

$$\text{s.t.} \quad \mathbf{g}(\mathbf{z}, \mathbf{y}) \geq 0 \tag{3.2}$$

where the first term in (3.1) is a discriminative loss on the data, which measures the discrepancy between prediction $f(\mathbf{x}_i)$ and vocabulary labels z_i and the second term in (3.1) is the regularizer on f to prevent overfitting. The constraints (3.2) insure that the vocabulary labels $\mathbf{z}$ are indeed representative for a certain geo-location, i.e. mostly occur in the positive set and do not occur

in the negative locations. Note that we wish optimize over both the mapping f and vocabulary labels $\mathbf{z}$. In other words, we wish to find a discriminative classifier f and a grouping of patches x_i into clusters (visual elements), given by $\mathbf{z}$, such that the regularized discriminative loss (3.1) is minimized. This is a form of a *discriminative clustering* problem [18, 202]. This type of approach have been used, e.g., for image co-segmentation [92, 93]. We have also applied discriminative clustering to temporal action detection [56] and recently joint learning of actors and actions in videos as described in chapter 4.

In practice, rather than solving one large multi-label problem, we solve a set of Z binary problems, learning a set of 1-vs-all binary classifiers (detectors) f_j, one for each visual element j. Each detector is a linear SVM classifier, i.e. $f_j = \mathbf{w}_j{}^\top x + b_j$, where $\mathbf{w}_j$, b_j are the parameters of the classifier. Constraints (3.2) ensure that a certain proportion of the top scoring detections for each classifier are within the positive set. The optimization is solved by alternatively optimizing the label assignments $\mathbf{z}$ and learning the set of classifiers f_j. Initialization of $\mathbf{z}$ is important and is done by finding discriminative clusters of descriptors x using Euclidean distance. Details are given in [54].

Results. Figure 3.3(top) shows the results of running the developed algorithm on a dataset of approximately 10,000 Google Street-view images from 12 cities. Results are shown for a subset of four cities (due to space limitations, a subset of elements was selected manually to show variety; see the project webpage[1] for the full list). For example, in Paris, the top-scoring elements find some of the main features that make Paris look like Paris: doors, balconies, windows with railings, street signs and special Parisian lampposts. Figure 3.3(bottom) shows geographical locations for the top-scoring detections for 3 different visual elements in Paris (a sampling of detections are shown below each map), revealing interestingly non-uniform distributions. Automatically discovering such architectural patterns may be useful to both architects and urban historians. More results are in [54].

3.3 Transferring mid-level representations

In the previous section we have shown that a mid-level visual representation of a city (visual detectors for hundreds of characteristic architectural elements) can be learnt in a weakly supervised manner from large amounts of images with weak annotations. In this section we investigate whether a rich mid-level image representation learned on one visual recognition task, where large amount of *fully supervised* data is available, can be transferred to another task where supervision is relatively scarce. This is interesting as large datasets of millions of images labelled with thousands of object classes are now becoming available [53]. However, collecting fully labelled datasets of similar scale for any new given task may be difficult. In particular, we investigate image representations learned using convolutional neural networks (CNN).

Convolutional neural networks (CNN) are high-capacity classifiers with very large numbers of parameters that must be learned from training examples. While CNNs have been advocated beyond character recognition for other vision tasks [137, 194] including generic object recognition [110], their performance was limited by the relatively small sizes of standard object recognition datasets. This situation has changed with the appearance of the large-scale ImageNet dataset [53] and the rise of GPU computing. Krizhevsky *et al.* [101] achieve a performance leap in image classification on

[1] http://graphics.cs.cmu.edu/projects/whatMakesParis/

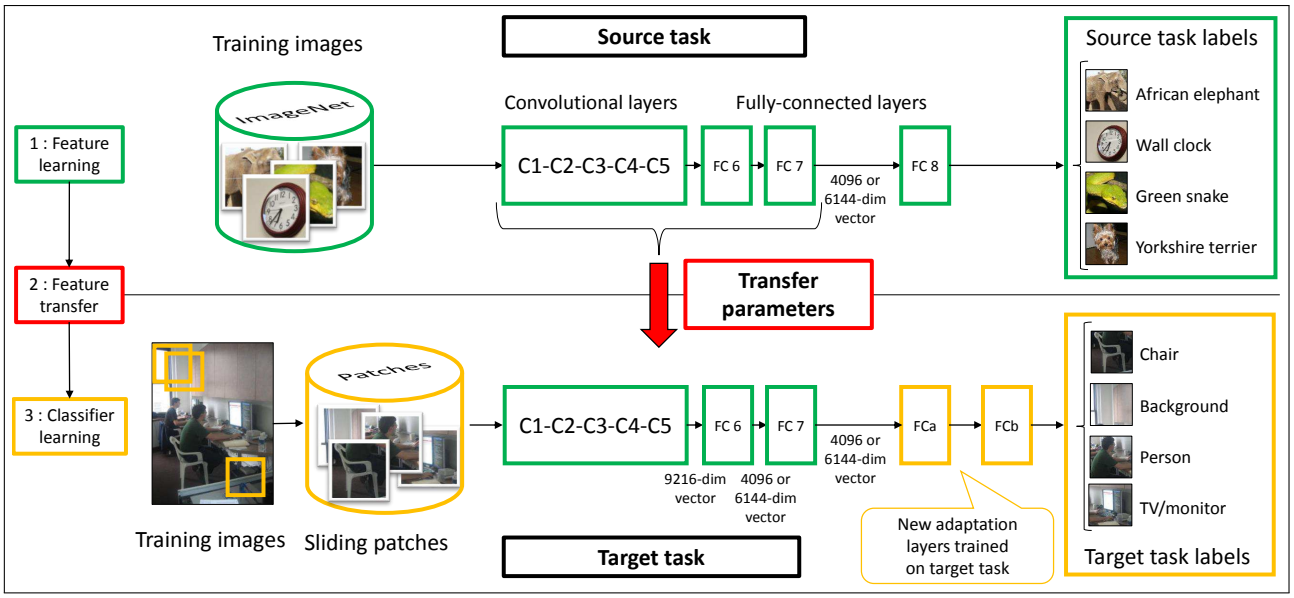


Figure 3.4: **Transferring parameters of a convolutional neural network (CNN).** First, the network is trained on the source task (ImageNet classification, top row) with a large amount of available labelled images. Pre-trained parameters of the internal layers of the network (C1-FC7) are then transferred to the target tasks (Pascal VOC object or action classification, bottom row). To compensate for the different image statistics (type of objects, typical viewpoints, imaging conditions) of the source and target data we add an adaptation layer (fully connected layers FCa and FCb) and train them on the labelled data of the target task.

the ImageNet 2012 Large-Scale Visual Recognition Challenge (ILSVRC-2012), and further improve the performance by training a network on all 15 million images and 22,000 ImageNet classes. As much as this result is promising and exciting, it is also worrysome. Will we need to collect millions of annotated images for each new visual recognition task in the future?

It has been argued that computer vision datasets have significant differences in image statistics [190]. For example, while objects are typically centered in Caltech256 and ImageNet datasets, other datasets such as Pascal VOC and LabelMe are more likely to contain objects embedded in a scene. Differences in viewpoints, scene context, "background" (negative class) and other factors, inevitably affect recognition performance when training and testing across different domains [145, 158, 190]. Similar phenomena have been observed in other areas such as NLP [90]. Given the "data-hungry" nature of CNNs and the difficulty of collecting large-scale image datasets, the applicability of CNNs to tasks with limited amount of training data appears as an important open problem.

Related Work. Our work is related to numerous works on transfer learning, image classification, and deep learning, which we briefly discuss below.

Transfer learning. Transfer learning aims to transfer knowledge between related *source* and *target* domains [139]. In computer vision, examples of transfer learning include [15, 187] which try to overcome the deficit of training samples for some categories by adapting classifiers trained for other categories. Other methods aim to cope with different data distributions in the source and target domains for the same categories, e.g. due to lighting, background and view-point variations [66, 98, 158]. These and other related methods adapt classifiers or kernels while using standard image features. Differently to this work, we here transfer image representations trained on the source task.

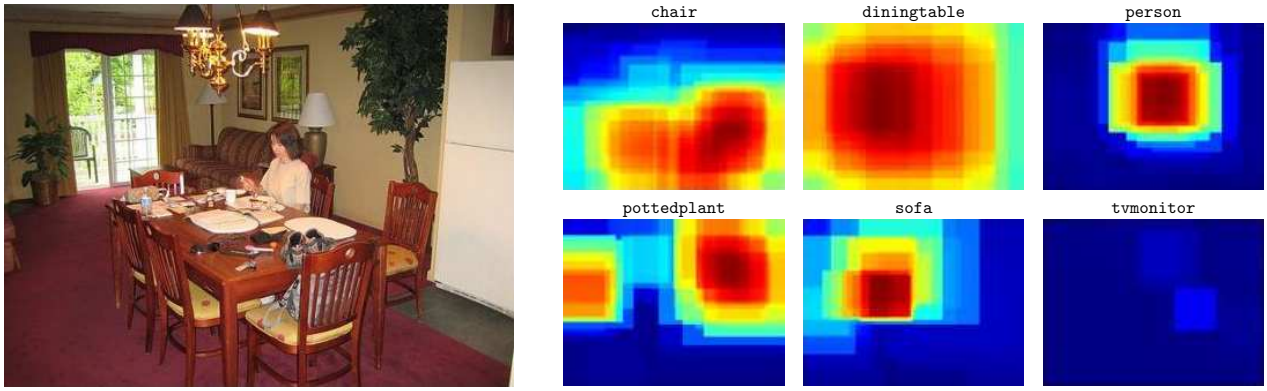


Figure 3.5: Recognition and localization results of our method for a Pascal VOC test image. Output maps are shown for six object categories with the highest responses.

More similar to our work, [4] trains CNNs on unsupervised pseudo-tasks. Differently to [4] we pre-train the convolutional layers of CNNs on a large-scale supervised task and address variations in scale and position of objects in the image. Transfer learning with CNNs has been also explored for Natural Language Processing [43] in a manner closely related to our approach.

Visual object classification. Most of the recent image classification methods follow the bag-of-features pipeline [46]. Densely-sampled SIFT descriptors [121] are typically quantized using unsupervised clustering (k-means, GMM). Histogram encoding [46, 178], spatial pooling [107] and more recent Fisher Vector encoding [140] are common methods for feature aggregation. While such representations have been shown to work well in practice, it is unclear why they should be optimal for the task. This question raised considerable interest in the subject of mid-level features [32, 94, 172], and feature learning in general [109, 149, 185]. The goal of this work is to show that convolutional network layers provide generic mid-level image representations that can be transferred to new tasks.

Deep Learning. The recent revival of interest in multilayer neural networks was triggered by a growing number of works on learning intermediate representations, either using unsupervised methods, as in [83, 108], or using more traditional supervised techniques, as in [64, 101].

Approach overview. We propose to transfer image representations learned with convolutional neural networks on large datasets to other visual recognition tasks with limited training data. In particular, we design a method that uses ImageNet-trained layers of the neural network architecture recently proposed in [101] to compute efficient mid-level image representation for images in Pascal VOC. The CNN architecture of [101] contains more than 60 million parameters. Directly learning so many parameters from only a few thousand training images is problematic. The key idea of this work is that the internal layers of the CNN can act as a *generic extractor of mid-level image representation*, which can be pre-trained on one dataset (the *source task*, here ImageNet) and then re-used on other *target tasks* (here object and action classification in Pascal VOC), as illustrated in Figure 3.4. However, this is difficult as the labels and the distribution of images (type of objects, typical viewpoints, imaging conditions, etc.) in the source and target datasets can be very different. To address these challenges we (i) design an architecture that explicitly remaps the class labels between the source and target tasks (see figure 3.4), and (ii) develop training and test procedures, inspired by sliding window detectors, that explicitly deal with different distributions of object sizes, locations and scene clutter in source and target tasks. Details are given in [135].

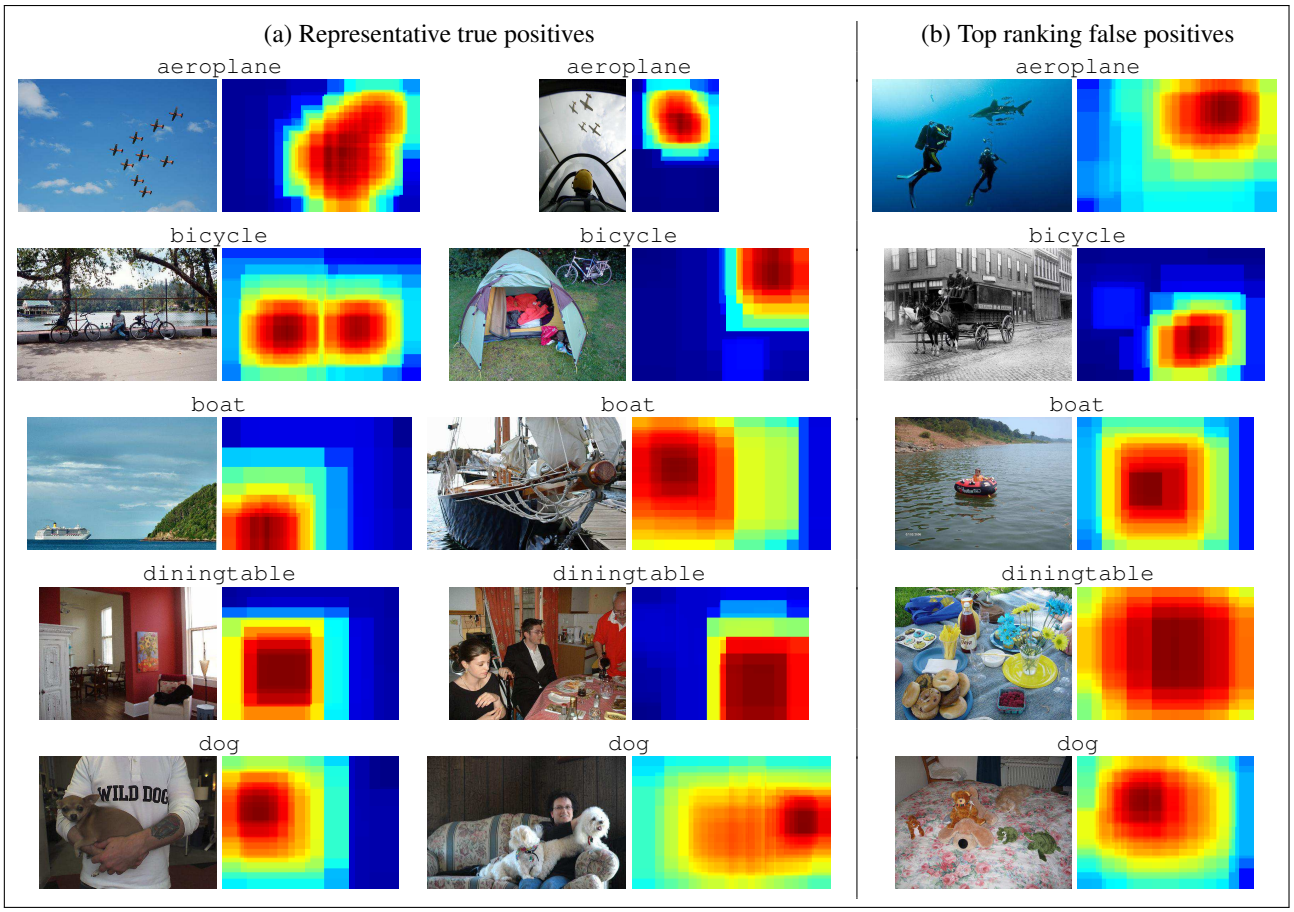


Figure 3.6: Response maps on representative images of seven categories of the VOC 2012 classification test set. The rightmost column contains the highest-scoring false positive (according to our judgement) for each of these categories. Correct estimates of object locations and scales are provided by the score maps.

	pln	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	**mAP**
INRIA [126]	77.5	63.6	56.1	71.9	33.1	60.6	78.0	58.8	53.5	42.6	54.9	45.8	77.5	64.0	85.9	36.3	44.7	50.6	79.2	53.2	59.4
NUS-PSL [181]	82.5	79.6	64.8	73.4	**54.2**	75.0	77.5	79.2	46.2	62.7	41.4	74.6	**85.0**	76.8	91.1	53.9	61.0	**67.5**	83.6	70.6	70.5
PRE-1000C	**88.5**	**81.5**	**87.9**	**82.0**	47.5	**75.5**	**90.1**	**87.2**	**61.6**	**75.7**	**67.3**	**85.5**	83.5	**80.0**	**95.6**	**60.8**	**76.8**	58.0	**90.4**	**77.9**	**77.7**

Table 3.1: Per-class results for object classification on the VOC2007 test set (average precision %).

Results. We apply our mid-level feature transfer scheme to the Pascal VOC 2007 object classification task. Results are reported in Table 3.1. Our transfer technique (PRE-1000C) demonstrates significant improvements over previous results on this data outperforming the 2007 challenge winners [126] (INRIA) by 18.3% and the more recent work of [181] (NUS-PSL) by 7.2%. Results on the Pascal VOC 2012 object classification and action recognition tasks outperforming the current state-of-the-art on this data are shown in [135].

Although our method has not been explicitly designed for the task of localization, we have observed strong evidence of object and action localization provided by the network at test time. For qualitative assessment of localization results, we compute an output map for each category by averaging the scores of all the testing patches covering a given pixel of the test image. Examples of such output maps are given in Figures 3.5 and 3.6. This visualization clearly demonstrates that the system knows the size and locations of target objects within the image. Addressing the detection task seems within reach.

3.4 Beyond bounding boxes: predicting 3D models from images

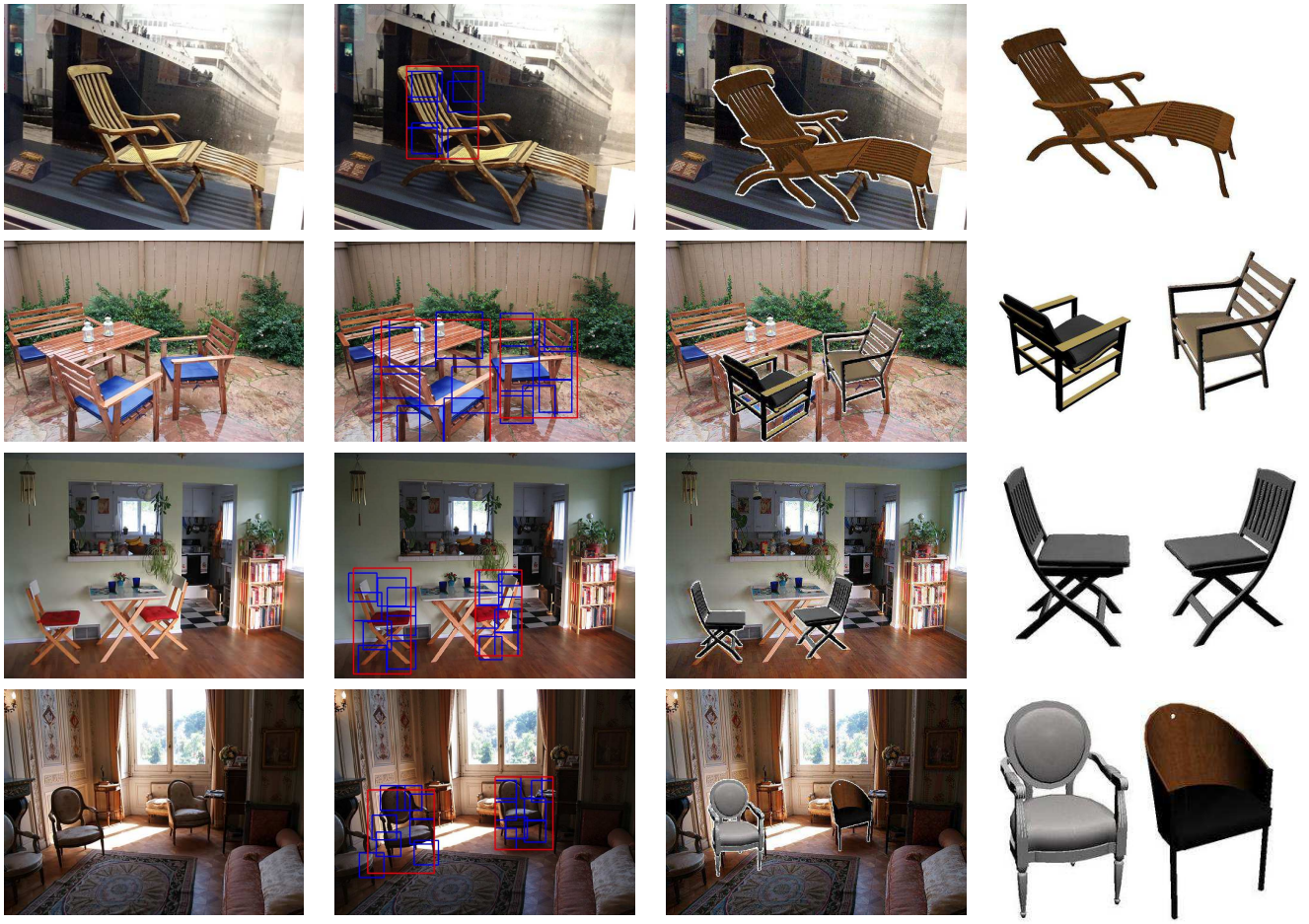
Methods discussed in previous sections of this chapter learn appearance based models from training example images depicting 2D views of objects with varying level of supervision. While their results are encouraging, their output is only a name of the object found in the image (e.g. "chair") or a bounding box / segmentation giving its rough location (cf. figure 3.5 in the previous section). While this type of result is reasonable for tasks such as retrieval (e.g. "find all chairs in this dataset"), it is rather unsatisfying for doing any deeper reasoning about the scene (e.g. "what's the pose of the chair?", "can I sit on it?", "where can I grasp it?", "what is this chair occluding?", etc). All these questions could be much more easily answered, if only we had a 3D model of the chair aligned with the image.

The work presented in this section aims to combine some of the benefits of the 3D model-based instance alignment methods (e.g. [84, 122, 120] and methods discussed in chapter 2) with the modern, appearance-based object category tools (e.g., [48, 67, 196] and methods discussed in previous sections of this chapter) towards getting a best-of-both-worlds object recognition engine. The idea is to use a large library of textured 3D object models that have become publicly available on the Internet, to implicitly represent both the 3D shape of the object class, as well as its view-dependent 2D appearance. Examples of 3D models output by our method are shown in figure 3.7.

We picked the "chair" category as the running example in this work because: 1) it is very hard even for the state-of-the-art methods [67], achieving only 0.13–0.20 average precision (AP) on PASCAL VOC [1]; 2) it is a category well-represented in the publically-available 3D model collections (e.g. Google 3D Warehouse), 3) chairs have huge intra-class variation – whereas there are perhaps only hundreds of models of cars ever made, there are thousands of different types of chairs!

Related work. From its very beginnings [150] and up until the early nineties [129], object recognition research has been heavily geometry-centric. The central tenet of the time was *alignment*, and the act of recognition was posed as correctly aligning a 3D model of an object with its 2D depiction in the test image. The parameters recovered during alignment (object pose, object scale, etc.) served as the output of the recognition process, to be used, for instance, in the perception-manipulation loop in robotics applications. Unfortunately, the success of these 3D model-based methods was largely limited to instance recognition tasks for objects with well-pronounced rectilinear structures (e.g. staplers were a favorite example). As the field moved toward category recognition and objects with more complex appearance, 3D model-based object recognition has been replaced by the new 2D appearance-based methods (e.g. [48, 67, 196]). These methods forgo 3D and operate directly on the 2D image plane. Thus, instead of a 3D model of an object, they use a large dataset of 2D views of the object class from different viewpoints, as the model. These methods have shown steadily improving performance on a number of challenging tasks, such as the PASCAL VOC dataset [59].

This work is part of an emerging trend towards reclaiming some of the early successes in 3D recognition, and combining them with modern visual recognition tools. 3D geometry with multi-view constraints has served as a strong alignment oracle for *instance-level* recognition [116, 153] and retrieval [41]. Images are typically represented using local invariant features such as SIFT [121], which work best for textured objects such as building facades [116, 141, 160]. More recent work has seen the re-emergence of contour-based representations for matching skylines [16] and smooth



(a) Input images (b) DPM [13] output (c) Our aligned outputs (d) Retrieved 3D chairs

Figure 3.7: Comparison of our algorithm output with the deformable parts model (DPM) [67]. While the DPM correctly predicts the 2D location of the depicted chairs, along with the 2D location of its parts, our algorithm is able to predict the 3D pose and style of the chair.

objects such as furniture pieces [118] or statues [11], however, these efforts have also largely focused on instance recognition.

In *category-level* recognition, recent work has explored recognition, alignment and fine pose estimation for cars and bicycles in outdoor scenes [82, 207] using low-dimensional parametric deformable 3D models combined with a small number of learnt part detectors. Others have explored simplified 3D cuboid models for reasoning about outdoor scenes [79] or detecting and estimating pose for box-like objects such as printers, beds or sofas [69, 201], often using simplifying 3D box-layout constraints for indoor scenes [36, 50]. In a separate line of research, a discriminative, exemplar-based object detection framework [124] demonstrated results for *transferring* 3D object geometry onto a test image, once 2D-to-2D alignment with an example was established. However, [124] models an object with a single global template, thus requiring a huge amount of exemplars to represent categories with high intra-class variation. Non-parametric representations have also shown promising results for 3D to 2D matching of indoor scenes [159], but again, due to the use of a global scene descriptor, this method works only for highly structured scenes, such as (tidy) bedrooms.

Addressing the need for descriptors that capture less than a full object/scene, but are more

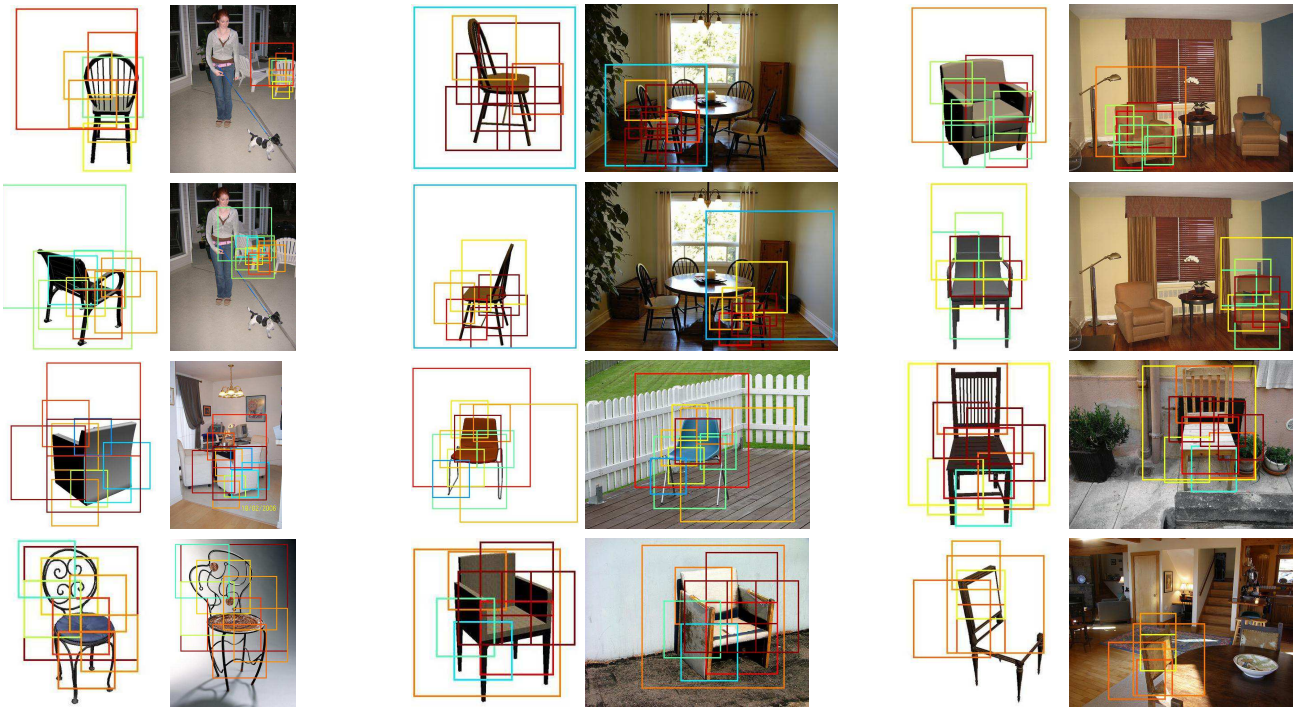


Figure 3.8: The output alignments. Matched parts are colored based on their detection confidence (warmer colors are more confident).

descriptive than low-level features such as SIFT [121], are a new breed of *mid-level* visual elements. Recently, discriminatively trained mid-level representations have shown great promise for various visual recognition tasks, both in a supervised regime [31], as well as unsupervised [86, 94, 172], including instance-level 3D to 2D alignment described in chapter 2. While here we build on the latter work, we address the much more difficult problem of category-level recognition and alignment of 3D objects.

Approach overview. In this work we propose a non-parametric, instance-based 3D representation for object categories with large intra-class variability, using "chairs" as the running example. Our representation consists of more than 1,300 CAD models to capture both, different styles of chairs, as well as the wide range of possible viewpoints. Chair detection in new images is accomplished by finding an alignment between the 2D chair and the most similar 3D chair model rendered at the most appropriate camera viewpoint, as shown on figure 3.7. Explicitly representing and synthesizing the fine grained style and viewpoint of the 3D object category significantly simplifies the difficult task of 3D to 2D alignment. Nevertheless, reliably matching a synthesized view of an object to a real photograph is still very challenging due to differences in, e.g., texture, materials, color, illumination or geometry. Furthermore, it is well known that computer generated images have different statistical properties than real photographs. To address these issues, we cast the matching problem as a classification task, and represent the collection of 3D models using a large set of more than 800,000 *mid-level visual elements* – linear classifiers over HOG features learnt from the rendered views in a discriminative fashion. This is similar to the 3D to 2D alignment approach described in chapter 2, but here applied to more than thousand 3D CAD models of chairs. As each of the 800,000 visual elements is learnt individually, calibrating their matching scores becomes a critical issue and we address this by learning a linear calibrating function for each element on a common dataset of negative images. Finally, we wish to be tolerant to small geometric defor-

mations (such as a chair with longer legs or shorter armrests) and hence we develop a matching procedure that allows for small deformations in the spatial configurations of the matched visual elements while preserving consistent viewpoint and style. Details are described in [13].

Results. In Figure 3.8 we show example output alignments of our algorithm. Notice that our algorithm can detect many different styles of chairs in different poses. For many cases, the predicted chair matches quite well the input depicted chair style and pose. When the style is not exactly correct, in many cases a similar style is returned, often retrieving an exact partial match to the depicted chair. Moreover, our approach shows some robustness to background clutter and partial occlusion and cropping. In Figure 3.7 we compare the output of our algorithm with the Deformable Parts Model (DPM) [67]. While the DPM correctly predicts the 2D location of the depicted chairs, along with the 2D locations of its parts, our algorithm produces a more informative result. The aligned 3D chair pose and style allows for true 3D reasoning of the input scene.

We evaluate the detection accuracy of our algorithm on the PASCAL VOC 2012 dataset. We report detection precision-recall on images marked as non-occluded, non-truncated, and not-difficult in the chairs validation set. While this is an easier set compared to the full validation set, nonetheless it is very challenging due to the large intra-class variation, chair poses, and background clutter. We note that even removing these difficult images results in a set with partially occluded and truncated chairs, as seen in figure 3.8. The resulting set contains 179 images with 247 annotated chairs. We compare the proposed approach against two baselines: (i) DPM [67] and (ii) a root template detector, using the LDA version of Exemplar-SVM [124]. During detection, we run the template in sliding window fashion across the input image. The full precision-recall curves for all three methods are in [13]. Our approach achieves an average precision (AP) of 0.339 on this task. The DPM and root template exemplar detector baselines achieve AP 0.410 and 0.055, respectively. Our performance is noteworthy as it does not use any of the PASCAL VOC training images. We investigated combining our algorithm with DPM for the detection task. For this we estimated an affine transformation for the DPM scores to calibrate it in the range of our returned scores. For overlapping detected windows for the two methods, we give one twice the confidence and discard the other. Combining our approach with DPM yields an AP of 0.452, which significantly out-performs the state-of-the-art DPM baseline. The increase in performance for the combined approach is due to the additional detected chairs at higher recall levels. More results, including a quantitative evaluation of the recovered chair style and viewpoint, can be found in [13].

3.5 Discussion

In this chapter we have investigated object category models learnable from different types and amounts of supervision ranging from images with no labels to a library of full 3D models. The results suggest that including even weak supervision seems beneficial over fully unsupervised techniques. In particular, weakly supervised learning using a discriminative clustering cost appears as a powerful promising approach, and we come back to it in the next chapter. However, the key open question is how to incorporate the variety of readily-available annotations into the optimization problem as different types of constraints. Finally, we have shown that object representations learnt using convolutional neural networks (CNNs) can be transferred across different visual recognition tasks, making CNNs a powerful and rich object representation. However, the current network architectures are only feed-forward. How to incorporate high-level reasoning about the entire scene, e.g. in the form of feed-back loops, still remains an open issue.

Chapter 4

Modeling and recognition of people and their activities in video

In this chapter we address the problem of recognizing people and their actions in video. As discussed in chapter 1, the key challenge lies in modeling the huge (intra-class) variation of dynamic scenes. At the same time, the amount of annotated training video data is severely limited (at least compared to object recognition in still images). To address these issues we introduce several models that are trainable from weak, but readily-available annotations in the form of text (e.g. a movie script) coarsely aligned with the video. First, in section 4.1 we show that person-specific face-based classifiers can be learnt from video with weak text supervision in the form of subtitles and transcripts. Second, in section 4.2 we develop a discriminative clustering model for temporal action localization that is learnt from video with coarsely aligned shooting scripts. Finally, in section 4.3 we combine person identification and action recognition, and develop a joint model of actors and actions in movies, supervised only by video with coarsely aligned text.

4.1 Learning person-specific classifiers from video and text

We investigate the problem of automatically labelling appearances of characters in TV or film material with their names. This is tremendously challenging due to the huge variation in imaged appearance of each character and the weakness and ambiguity of available annotation. However, we demonstrate that high precision can be achieved by combining multiple sources of information, both visual and textual.

We build on previous approaches which have matched frontal faces in order to "discover cast lists" in movies [71] or retrieve shots in a video containing a particular character [10, 173] based on image queries. The novelty we bring is to employ readily available textual annotation for TV and movie footage, in the form of subtitles and transcripts, to *automatically* assign the correct name to each face image.

Alone, neither the script nor the subtitles contain the required information to label the identity of the people in the video – the subtitles record *what* is said, but not by *whom*, whereas the script records *who* says *what*, but not *when*. However, by automatic alignment of the two sources, it is possible to extract *who* says *what* and *when*. Knowledge that a character is speaking then gives a very weak cue that the person may be visible in the video. A key to the success of our method is to leverage this cue by *visually* detecting which (if any) character in the video corresponds to the speaker. This gives us sufficient annotated data from which it is possible to learn to recognize the

other instances of the character.

In addition to effective exploitation of cues from textual annotation, success depends on robust computer vision methods for multi-view face processing in video: (i) obtaining face tracks for individual people in the video; (ii) speaker detection (as discussed above) to determine if a name proposed by the aligned transcript should be used to label a face track; and (iii) classification, where the unlabelled face tracks are labelled by a classifier trained on the labelled tracks.

A particular strength of our approach is seamless tracking, integration and recognition of *profile* and *frontal* face detections. The outcome is two-fold. First, harvesting both frontal and profile views of people from video improves *coverage* (the number of characters that can be identified and the number of frames over which they are tracked). Second, an integrated treatment of multiple detectors enables learning across viewpoints e.g. (weakly) labelled profile views contribute to learning frontal appearance via tracking. For instance, if there is supervisory information available for a profile view and this profile is connected to a frontal view (e.g. the character turns their face) then the supervision can be transferred to the frontal view, harvesting additional labelled faces. The result is improved *accuracy* correctly identifying characters in the video.

Related work. Previous work on the recognition of characters in TV or movies has often ignored the availability of textual annotation. In the "cast list discovery" problem [71, 9], faces are clustered by appearance, aiming to collect all faces of a particular character into a few pure clusters (ideally one), which must then be assigned a name manually. It remains a challenging task to obtain a small number of clusters per character without merging multiple characters into a single cluster. Other work [62] has addressed finding particular characters specified a priori by building a model of a character's appearance from user-provided training data. Our initial efforts focused on the efficient retrieval of characters based on example face images [173].

Assigning names given a combination of faces and textual annotation has similarities to the "Faces in the News" labelling of [24]. In that work, faces appearing in images accompanying news stories are tagged with names by making use of the names appearing in the news story text. A clustering approach is taken, initialized by cases for which the news story contains a single name and the accompanying image contains a single (detected) face. Here we are also faced with similar problems in establishing the correspondence between text and faces: ambiguity can arise from deficiencies in the face detection, e.g. there may be several characters in a frame but not all their faces are detected, or there may be false positive detections; ambiguity can also arise from the annotation, e.g. in a reaction shot the person speaking (and therefore generating a subtitle) may not be shown. The combination of face detection and text has also been applied previously to face recognition in video, but the focus has been typically on the more constrained setup of news footage [203, 138, 204]. Profile faces have been detected and tracked in uncontrolled video (TV, movies) at the level of difficulty considered here [115, 147], but these papers have not dealt with the problem of *recognizing* faces in profile.

There has been considerable work on discriminative classification of faces. In the case of images from web pages (at the level of difficulty of faces in the wild) others have shown the benefit of the discriminative approach both for identity [127], and in the Facetracer project [103] for other attributes such as age, ethnic origin, and gender. Indeed, others [8, 102] have improved on the classification performance of our work [60, 61]. Our aims differ from related work which has made more extensive use of script text [44] in that we aim to build models which allow labelling of faces with *no* associated text, rather than selecting from candidate names obtained from a full transcript.

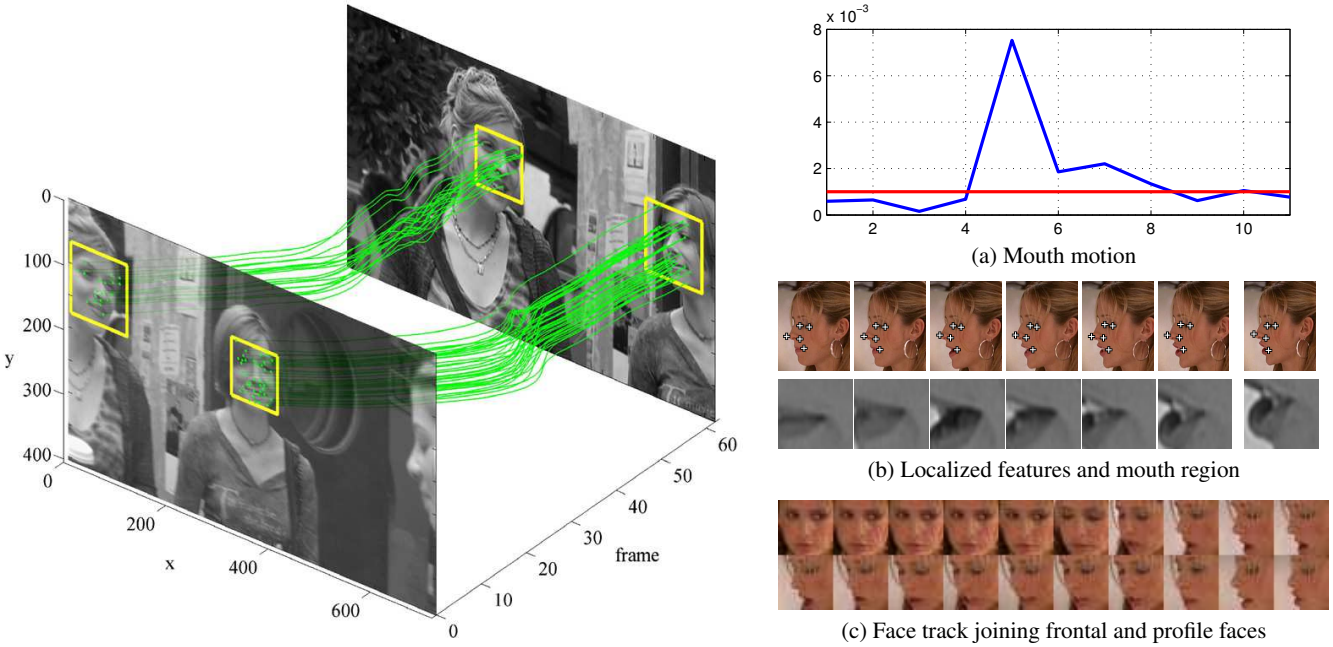


Figure 4.1: **Left: Face tracking by point tracking.** Trajectories of points tracked on the actors faces shown as curves in the video volume between the first and last frame of a 63 frame long video sequence. The challenges include moving camera as well as changing expressions and the head pose of actors.
Right: Speaker identification for profile faces. (a) Inter-frame differences for 11 frames of a face track. The horizontal line indicates the speaking threshold. (b) Above: Extracted face detections with facial feature points overlaid for frames 39. Below: Corresponding extracted mouth regions. (c) Original face track from frontal to profile views. A label is proposed for the automatically identified profile speaker detection. Note the character does not speak while in a frontal pose.

Approach overview. We automatically label cast members in video using only the video stream and textual information in the form of sub-titles and aligned transcripts. The method consists of three stages: (i) visual processing to obtain face tracks for individual people in the video; (ii) speaker detection to determine if a name proposed by the aligned transcript should be used to label a face track; and (iii) classification, where the unlabelled face tracks are labelled by a classifier trained on the labelled tracks. We briefly summarize these three stages. The details are given in [60, 61, 174].

First, both frontal and profile faces are detected in every frame of the video and tracked throughout a shot, such that each track is of a single character. The tracking algorithm associates face detections by point tracks as illustrated in figure 4.1(left). The aim of the subsequent algorithm is to label (associate names with) each track. Each face in the track is represented by a histogram of gradients (HOG) feature vector, computed from a face region rectified using automatically detected facial features (based on eyes, nose and mouth). A track is represented by this set of feature vectors.

Second, a transcript is aligned with the subtitles using dynamic time warping, so that speaker names appearing in the transcript are associated with a time interval of the video. This is weak supervision because the person speaking may not be visible or detected, and there may be other faces in the scene. To strengthen the supervision, names are only associated with tracks where the face is detected as speaking (both in frontal and profile views). Nevertheless, this is still noisy supervision. Speaker detection is based only on visual information and is illustrated in figure 4.1(right). The outcome of this stage is that some of the tracks are labelled (these are

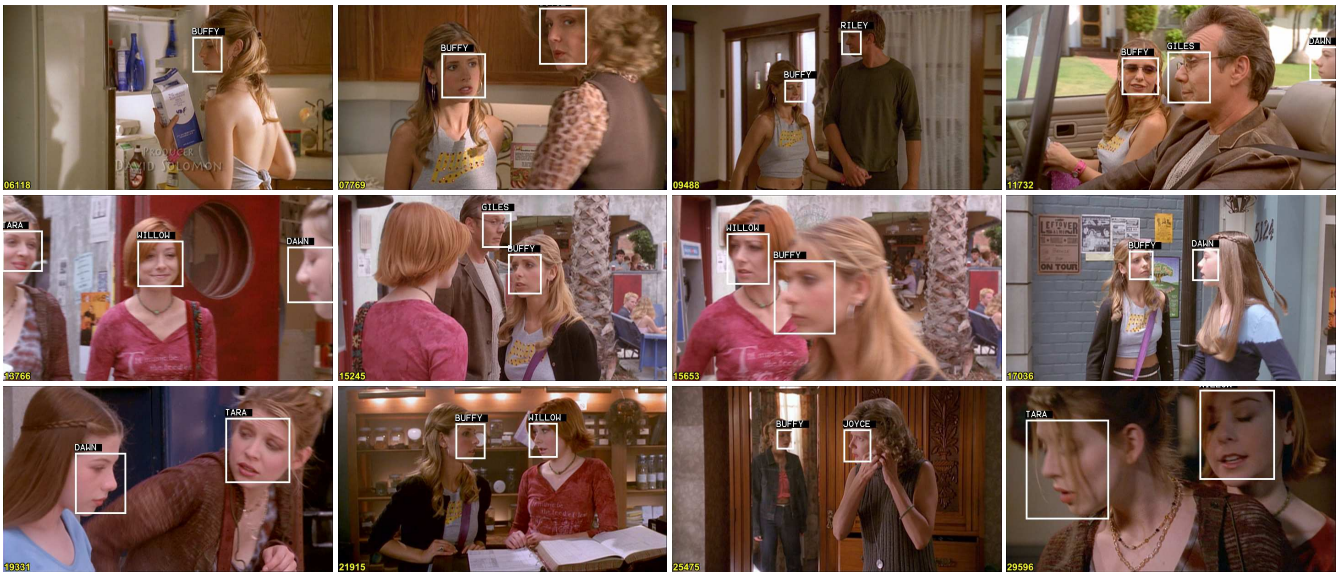


Figure 4.2: Examples of correct detection and naming throughout episodes 2 and 5 of the TV series "Buffy the Vampire Slayer". Note that correct naming is achieved over a wide range of scale, pose, facial expression and lighting.

referred to as exemplars).

Finally, The exemplar tracks are used to label the remaining tracks using their visual descriptors. This is formulated as a discriminative classification task, where we learn a multiple-kernel support vector machine (SVM) classifier for each character to discriminate the tracks of that person from the tracks of others.

Results. The proposed method was tested on seven episodes from season 5 of "Buffy the Vampire Slayer": episodes 1–6 and 13. Each episode contains around 40,000 frames and 33,000 – 42,000 face detections grouped into 1,500-2,200 face tracks. The number of main characters varies between 13-19 depending on the episode. Examples of correctly detected and named characters are shown in Figure 4.2. . Detailed quantitative evaluation on all seven episodes is given in [174]. The results demonstrate that seamless integration of frontal and profile face detections throughout the face recognition pipeline can increase the proportion of video frames labelled as well as the labeling accuracy due to more harvested training data.

4.2 Learning human actions from from video and text

In the previous section we demonstrated that appearance based models of people's faces can be learned from videos with aligned text. In this section we investigate whether video with aligned text can be used to learn models of human actions.

Action recognition has a long history of research with significant progress reported over the last years. Most of recent works, however, address the problem of action classification, i.e., "what actions are present in the video?" in contrast to "where?" and "when?" they occur. In this work, similar to [57, 105, 167, 205], we aim at identifying both the classes and the temporal location of actions in video, as illustrated in figure 4.3 (left).

Current work on action recognition report impressive results for evaluations in controlled settings such as in Weizman [26] and in KTH [165] datasets. At the same time, state-of-the-art

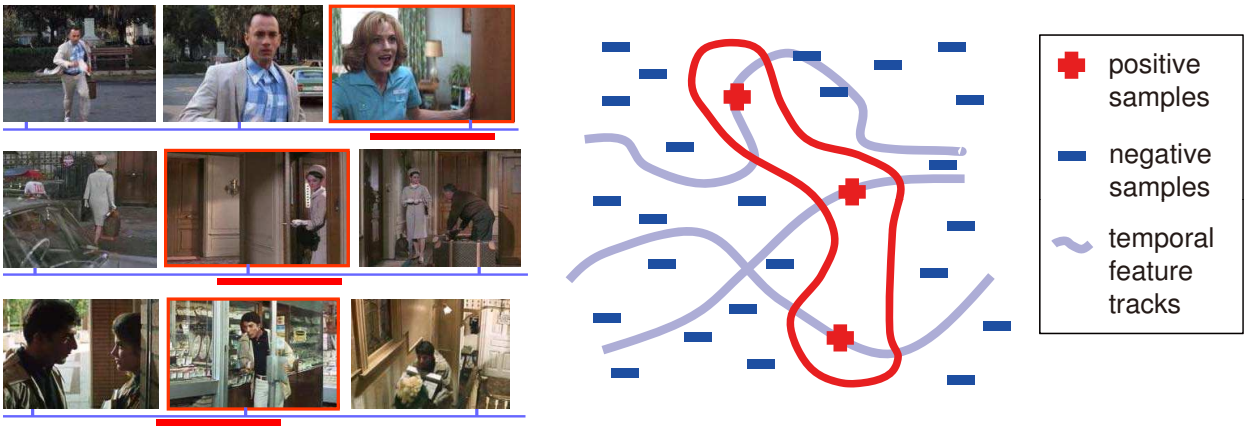


Figure 4.3: **Left: Video clips with *OpenDoor* actions provided by automatic script-based annotation.** Selected frames illustrate both the variability of action samples within a class as well as the imprecise localization of actions in video clips. **Right: In feature space**, positive samples are constrained to be located on temporal feature tracks corresponding to consequent temporal windows in video clips. Background (non-action) samples provide further constrains on the clustering.

methods only achieve limited performance in real scenarios such as movies and surveillance videos as demonstrated in, e.g. [104, 192]. This emphasises the importance of realistic video data with human actions for the training and evaluation. To this end we demonstrate results on several challenging feature length films. To avoid the prohibitive cost of manual annotation, we propose an automatic solution for training with only minimal manual supervision. Similar to previous section, this is achieved by using as supervision the textual description of the video content in the form of a shooting script coarsely aligned with video using timestamps from subtitles. However, these coarsely aligned scripts provide only an imprecise annotation, where the action is often located in the video with an error of up to 1,600 frames. In addition, the annotation is incomplete, where some actions that occur in the video are not described in the text, and contains errors, where some actions in the text are not depicted in the video (i.e. happen off-screen).

Related work. This work is related to several recent research directions. With respect to human action recognition, similar to [55, 104, 132, 165] and others, we adopt a bag-of-features framework, represent actions by histograms of quantized local space-time features and use an SVM to train action models. Our automatic video annotation is based on video alignment of scripts described in the previous section and also used in [44, 104]. Similar to [104], we use scripts to find coarse temporal locations of actions in the training data. Unlike [104] we use clustering to discover precise action boundaries from video.

Unsupervised action clustering and localization has been addressed in [205] by means of normalized cuts [169]. Whereas this direct clustering approach works in simple settings [205], we find it is not well suited for actions with large intra-class variation and propose an alternative discriminative clustering approach. [132] deals with unsupervised learning of action classes but only considers actions in simple settings and does not address temporal action localization as we do in this work. Recently and independently of our work, Buehler *et al.* [34] considered learning sign language from weak TV video annotations using multiple instance learning.

Our weakly supervised clustering is also related to the work on learning object models from weakly annotated images [42, 106]. The temporal localization of training samples in videos, ad-

dressed in this work, is also similar in spirit to weakly supervised learning of object part locations in the context of object detection [68]. Discriminative clustering methods have recently been used also for image and video co-segmentation [93, 92], object recognition (as described in section 3.2) and, independently of our work, for action recognition [131]. More detailed review of discriminative clustering methods is in section 4.3.

Several previous methods address temporal localization of actions in video. Whereas most of them evaluate results in simple settings [57, 167, 205], our work is more related to [105] that detects actions in a real movie. Differently to [105] our method is weakly supervised and enables the learning of actions with minimal manual supervision.

Approach overview. Our goal is to jointly segment video clips containing a particular action—that is, we aim at separating what is common within the video clips (i.e., the particular action) from what is different among these (i.e, the background frames). Our setting is however simpler than general co-segmentation in the image domain since we only perform *temporal* segmentation. That is, we look for segments that are composed of contiguous frames. For simplicity, we further reduce the problem to separating one segment per video clip (the action segment) from a set of *background video segments*, taken from the same movie or other movies, and which are unlikely to contain the specific action. We thus have the following learning problem: We are given M video clips $c_1, \ldots, c_M$ containing the action of interest but at unknown position within the clip. Each clip c_i is represented by n_i temporally overlapping segments centered at frames $1, \ldots, n_i$ represented by histograms $h_i[1], \ldots, h_i[n_i]$ in $\mathbb{R}^N$. Each histogram captures the ℓ_1-normalized frequency counts of quantized space-time interest points [55, 104, 132, 165], i.e. it is a positive vector in $\mathbb{R}^N$ whose components sum to 1. We are also given P background video segments represented by histograms $h_1^b, \ldots, h_P^b \in \mathbb{R}^N$. Our goal is to find in each of the M clips i one specific video segment centered at frame $f_i \in \{1, \ldots, n_i\}$ so that the set of M histograms $h_i[f_i]$, $i = 1, \ldots, M$ form one cluster while the P background histograms form another cluster as illustrated in figure 4.3 (right).

We formulate the above clustering problem as a minimization of the following discriminative cost function [202]

$$J(f, w, b) = C_+ \sum_{i=1}^{M} \max\{0, 1 - w^\top \Phi(h_i[f_i]) - b\} + C_- \sum_{i=1}^{P} \max\{0, 1 + w^\top \Phi(h_i^b) + b\} + \|w\|^2, \quad (4.1)$$

where the first two terms represent the hinge loss on positive and negative training data weighted by factors C_+ and C_- respectively, and the last term is the regularizer of the classifier. Further, $w \in \mathcal{F}$ and $b \in \mathbb{R}$ are parameters of the classifier and Φ is the implicit feature map from $\mathbb{R}^N$ to feature space $\mathcal{F}$, corresponding to the intersection kernel between histograms. We wish to minimize (4.1) with respect to both the parameters of the classifier, w, b, and the locations, $\{f_i\}$, of the actions in the positive clips. This is achieved using a coordinate descent algorithm where we alternate between (i) optimizing w, b, given fixed action segmentation $\{f_i\}$, and (ii) finding the best action segmentation $\{f_i\}$, given fixed parameters of the classifier w and b. The algorithm starts by learning the classifier, where the segmentation is initialized to entire clips. Details are given in [56].

Results. We train and test classifiers for two actions: "OpenDoor" and "SitDown". To train each action classifier we use fifteen feature length movies[1] containing 31 and 44 clips for the two actions,

[1]Our fifteen training movies were selected based on their availability as well as the quality of script alignment. The titles of the movies are: American Beauty; Being John Malkovich; Casablanca; Forrest Gump; Get Shorty; Its

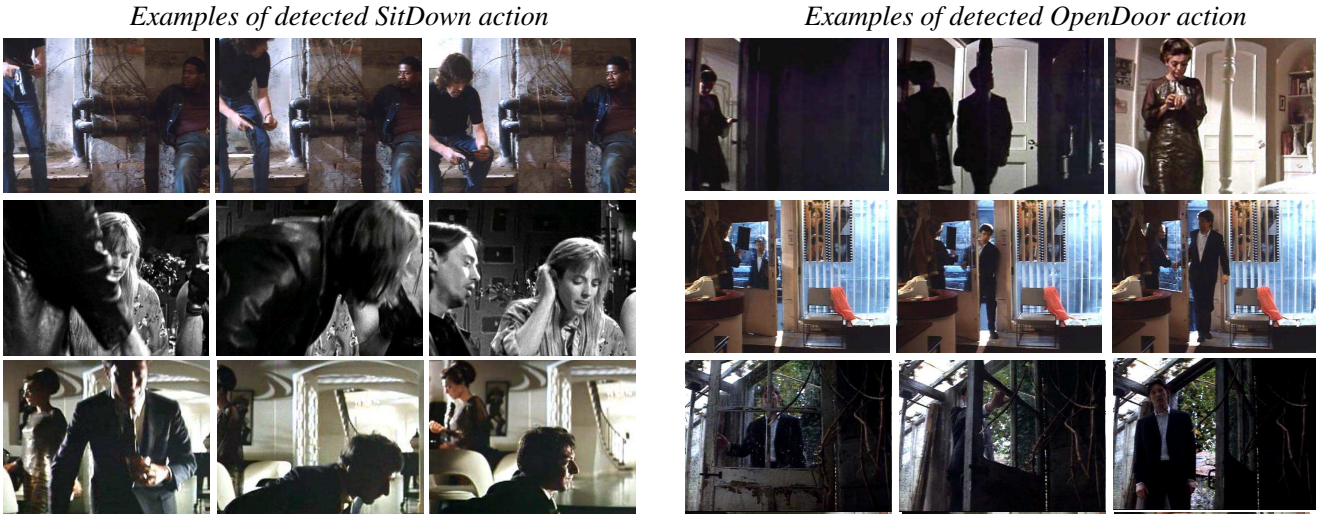


Figure 4.4: **Automatic annotation of actions in video.** Examples of action samples detected with the automatically trained action detector in three test movies.

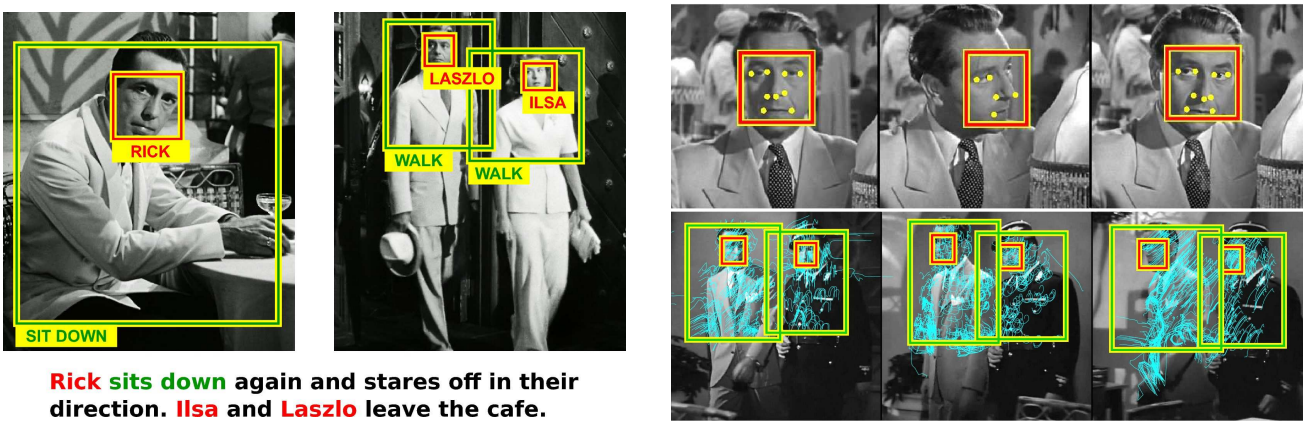
respectively. We use these clips as input to the discriminative clustering algorithm and obtain as output segments with temporally localized action boundaries. The output segments are passed as positive training samples to train an SVM action classifier. To test the detection performance we manually annotated all 93 OpenDoor and 86 SitDown actions in three movies: Living in oblivion, The crying game and The graduate. The average precision (AP) of the two resulting classifiers on this extremely challenging test data is 0.121 and 0.141 for OpenDoor and SitDown, respectively. Detailed quantitative evaluation is given in [56]. The results demonstrate that action segments provided by our clustering method clearly outperform classifiers trained on the entire video clips without the automatic temporal segmentation (AP of 0.016 and 0.029, respectively). Moreover, the performance of automatically trained action detectors is comparable to the detectors trained on the ground truth segmented data (AP of 0.139 and 0.144, respectively). We emphasize the large amount (450.000 frames in total) and high complexity of our test data illustrated with a few detected action samples in figure 4.4.

4.3 Joint learning of actors and actions in movies

In section 4.1 and 4.2 we saw how videos with coarsely aligned text can be used to train identity and action classifiers, respectively, with no manual supervision. In this section, we investigate how the two problems – (i) action recognition and (ii) person identification – can be formulated jointly. The intuition is that objects, people and actions often co-occur in the video. Knowing that "Rick sits down" in a video can help annotating a sitting down action if we can localize Rick and vice versa, see Figure 4.5(a). Action recognition can particularly help person identification for rare subjects and subjects facing away from the camera (e.g., Ilsa walks away to the door). Recognizing actors, on the other hand, can be most useful for learning rare events (e.g. hand shaking).

We follow this intuition and address *joint* weakly supervised learning of actors and actions by exploiting their co-occurrence in movies. We build on methods developed in this chapter as well as

a Wonderful Life; Jackie Brown; Jay and Silent Bob Strike Back; Light Sleeper; Men in Black; Mumford; Ninotchka; The Hustler; The Naked City and The Night of the Hunter.



(a) (b)

Figure 4.5: **Joint learning of actors and actions from video with aligned text. (a) Example result** of automatic detection and annotation of characters and their actions in the movie Casablanca. The automatically resolved correspondence between video and script is color-coded. **(b) Representing video.** Top: face track together with extracted facial features. An appearance descriptor is extracted from each face in the track (cf. section 4.1). Bottom: Actions are represented by motion features based on dense point trajectories [198] extracted from tracked upper body bounding boxes.

previous work by others [45, 104, 125, 184], and use movie scripts as a source of weak supervision. Differently from this prior work, we use actor-action co-occurrences derived from scripts to constrain the weakly supervised learning problem. In particular, in this section we present the following contributions. First, we consider a richer use of textual information for video and learn from pairs of names and actions co-occurring in the text. Second, we formulate the problem of finding characters and their actions as weakly supervised structured classification of pairs of action and name labels. Third, we develop a new discriminative clustering model jointly learning both actions and names and incorporating text annotations as constraints. The corresponding optimization is formulated as a quadratic program under linear constraints. Finally, we demonstrate the validity of the model on two feature-length movies and corresponding movie scripts, and demonstrate improvements over earlier weakly supervised methods.

Related work. Learning from images and text has been addressed in the context of automatic annotation of images with keywords [22, 77, 198] or labeling faces with names in news collections [24]. Berg *et al.* [24] label detected faces in news photographs with names of people obtained from text captions. Recent work has looked at learning spatial relations (such as "on top of") from prepositions [78] or generating entire sentence-level captions for images [65, 136]. A generative model of faces and poses (such as "Hit Backhand") was learnt from names and verbs in manually provided captions for news photographs [123]. While the goal of this work is related to ours, we focus on learning from video with sparse, noisy and imprecise annotations extracted from scripts. To deal with the ambiguity of annotations, we develop a new discriminative weakly supervised clustering model of video and text.

In video, manually provided text descriptions have been used to learn a causal model of human actions in the constrained domain of sports events [80]. Other work has looked at learning from videos with readily-available text, but names (section 4.1 as well as e.g. [45]) and actions (section 4.2 and [104]) have been so far considered separately. The ambiguity and errors of readily

available annotations present a major challenge for any learning algorithm. These problems have been addressed by designing appropriate loss functions [45] or explicitly finding the corresponding instances in video using discriminative clustering similar to multiple instance learning as described in section 4.2. Others have looked at convex relaxations of discriminative clustering with hidden variables for image co-segmentation [93, 92].

Approach overview. We formulate the problem of jointly detecting actors and actions as discriminative clustering [18, 92]: grouping samples into classes so that an appropriate loss is minimized. We incorporate text-based knowledge as a suitable set of constraints on the cluster membership.

Let us suppose that we have two label sets, $\mathcal{P}$ and $\mathcal{A}$, representing person identities and action classes, respectively. We denote the number of labels as $P = |\mathcal{P}|$ and $A = |\mathcal{A}|$. Our data is organized into sets, that we refer to as bags, and which are indexed by $i \in I$. Every bag has a set of samples $\mathcal{N}_i$ and a set of annotations Λ_i. In our application, $\mathcal{N}_i$ is the group of person tracks appearing in a scene while Λ_i can be thought of as a set of sentences specifying who is doing what obtained from the movie script, as illustrated in figure 4.5(a). We denote by $N = \sum_{i \in I} |\mathcal{N}_i|$ the total number of person tracks in the video.

For every sample (person track) $n \in \mathcal{N}_i$ we have a feature vector $x_n \in \mathbb{R}^{1 \times d}$ representing both the face appearance as well as the upper body motion/appearance, as illustrated in figure 4.5(b). Every sample belongs to a class in $\mathcal{P}$ and a class in $\mathcal{A}$. For each sample we therefore define a pair of latent variables z_n in $\{0,1\}^{1 \times P}$ and t_n in $\{0,1\}^{1 \times A}$ indicating to which person and action class it belongs. We define X to be a $N \times d$ data matrix with rows x_n, Z is a $N \times P$ matrix with person labels in rows z_n and T is a $N \times A$ matrix with action labels in rows t_n. The p-th element of a vector z_n is written z_{np}.

The aim is to recover latent variables z_n, t_n for every sample x_n and, at the same time, learn two multi-class classifiers $f : \mathbb{R}^d \to \mathbb{R}^P$ and $g : \mathbb{R}^d \to \mathbb{R}^A$ for persons and actions respectively, given weak supervision in the form of constraints on Z and T obtained from the text annotations. This is formulated as the following minimization

$$\min_{T,Z,f} \quad \frac{1}{N} \sum_{i \in I} \sum_{n \in \mathcal{N}_i} \ell(z_n, f(\phi(x_n))) + \frac{1}{N} \sum_{i \in I} \sum_{n \in \mathcal{N}_i} \ell(t_n, g(\psi(x_n))) + \Omega(f) + \Omega(g) \tag{4.2}$$

$$\text{s.t.} \quad \forall i \in I \ , \ h_i(Z,T) \geq 0, \tag{4.3}$$

where the first two terms in (4.2) correspond to the discriminative loss for faces and actions, respectively, and $\Omega(f)$, $\Omega(g)$ are regularizers. ϕ and ψ are feature maps corresponding to kernels used for faces and actions, respectively. Constraints (4.3) on latent variables Z and T encode the joint occurrences of actions and person names the text script. In particular, for every person-action pair (p, a) found in the script we construct a bag i containing samples $\mathcal{N}_i$ corresponding to person tracks in the temporal proximity of (p, a). What we want to model is the following: "if a person-action pair is mentioned in the script, it should appear at least once in the bag". This can be translated into a constraint on the sum of latent variables of tracks within the bag as: $\sum_{n \in \mathcal{N}_i} z_{np}\, t_{na} \geq 1$. Similar inequality constraints can be defined for independent occurrences of actions and names in the script as outlined in detail in [29]. However, it is the joint constraints that provide the coupling between the action and person recognition problems.

When Z and T take binary values, solving the minimization problem defined above is NP hard. We thus relax it by considering real-valued positive matrices. The relaxed problem is not jointly

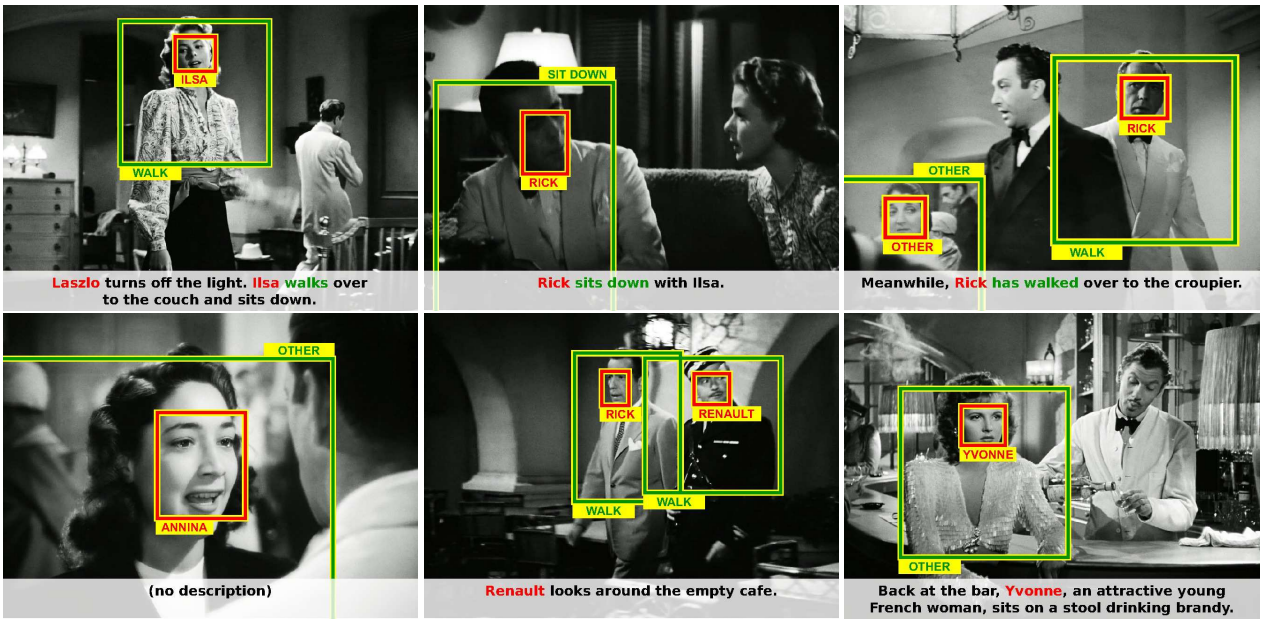


Figure 4.6: **Examples of automatically assigned names and actions in the movie Casablanca.** Note that even very infrequent characters are correctly classified (Annina and Yvonne). See more examples on the project web-page [2].

convex in Z and T because of the coupling constraint in eq. (4.3). Once we fix one of the two matrices, the coupling constraint becomes linear in the other latent variable. We, therefore, perform a block coordinate descent and alternate optimization by solving for one of the matrices Z, T while fixing the other. Each of the two steps is a convex quadratic program under linear constraints. Details are given in [29].

Results. We report results for movies Casablanca and American Beauty. For both movies we extract person tracks and associated descriptors. For Casablanca, we obtain 1,273 person tracks containing 124,423 face detections while for American Beauty we use 1,330 person tracks containing 131,741 face detections. Example results for automatic labeling names of actors and actions using our method are illustrated figure 4.6. Detailed quantitative evaluation is in [29] and demonstrates that (i) the proposed approach outperforms the method described in section 4.1 as well as the weakly supervised learning of [45] for learning face classifiers from video, and (ii) that joint learning of faces and actions together outperforms weakly supervised learning of actions only.

4.4 Discussion

In this chapter, we have shown that person and action models can be learnt from movies with coarsely aligned shooting scripts. While face recognition in unconstrained videos is reaching a certain level of maturity, action recognition in unconstrained video is still extremely challenging. The key challenge appears to lie in designing mid- and high-level representations that can capture the extremely high appearance variability of dynamic scenes. Building on our work on object recognition described in chapter 3, a possible avenue of research is to investigate representations based on convolutional neural networks that could be trained on related (possibly still image) tasks.

Chapter 5

Discussion and outlook

In this thesis we have made a step towards automated visual intelligence. Our work has been part of the great progress in visual recognition in the recent years: large scale instance-level visual search is running on our mobile phones[1]; category-level image classification techniques are beginning to organize our pictures online[2]; and we can search video collections based on facial appearance of individual actors[3]. What will be next?

Despite the great progress, the current automated visual intelligence is still far away from the immense capabilities of the human brain. For example, people constantly draw on past visual experiences to anticipate future events and better understand, navigate, and interact with their environment, for example, when seeing an angry dog or a quickly approaching car. Currently there is no artificial system with a similar level of visual analysis and prediction capabilities. My work in the next four years will be focused on the goals of my ERC starting grant LEAP that aims to make a step in that direction. The project will leverage the emerging collective visual memory formed by the unprecedented amount of visual data available in public archives, on the Internet and from surveillance or personal cameras – a complex evolving net of dynamic scenes, distributed across many different data sources, and equipped with plentiful but noisy and incomplete metadata. The goal of this project is to analyze dynamic patterns in this shared visual experience in order (i) to find and quantify their trends; and (ii) learn to predict future events in dynamic scenes. With ever expanding computational resources and this extraordinary data, the main scientific challenge is now to invent new and powerful models adapted to its scale and its spatio-temporal, distributed and dynamic nature. Breakthrough progress on this problem would have profound implications on our everyday lives as well as science and commerce, with safer cars that anticipate the behavior of pedestrians on streets; tools that help doctors monitor, diagnose and predict patients health; and smart glasses that help people react in unfamiliar situations enabled by the advances from this project.

[1] E.g. Google Goggles, Bing scan or Koaaba.com.

[2] http://googleresearch.blogspot.com.au/2013/06/improving-photo-search-step-across.html

[3] VideoSurf.com (acquired by Microsoft). https://www.microsoft.com/en-us/news/press/2011/nov11/11-22xboxnovemberpr.aspx

Bibliography

[1] http://pascallin.ecs.soton.ac.uk/challenges/voc/voc2012/, 2012.

[2] http://www.di.ens.fr/willow/research/actoraction, 2013.

[3] S. Agarwal, Y. Furukawa, N. Snavely, B. Curless, S. Seitz, and S. Szeliski. Reconstructing rome. *Computer*, 43(6):40–47, 2010.

[4] A. Ahmed, K. Yu, W. Xu, Y. Gong, and E. Xing. Training hierarchical feed-forward visual recognition models using transfer learning from pseudo-tasks. In *ECCV*, 2008.

[5] N. Ahuja and S. Todorovic. Learning the taxonomy and models of categories present in arbitrary images. In *ICCV*, 2007.

[6] K. Alahari, G. Seguin, J. Sivic, and I. Laptev. Pose estimation and segmentation of people in 3D movies. In *ICCV*, 2013.

[7] D. G. Aliaga, P. A. Rosen, and D. R. Bekins. Style grammars for interactive visualization of architecture. *Visualization and Computer Graphics, IEEE Transactions on*, 13(4), 2007.

[8] N. E. Apostoloff and A. Zisserman. Who are you? – real-time person identification. In *Proc. BMVC.*, 2007.

[9] O. Arandjelovic and R. Cipolla. Automatic cast listing in feature-length films with anisotropic manifold space. In *CVPR*, 2006.

[10] O. Arandjelovic and A. Zisserman. Automatic face recognition for film character retrieval in feature-length films. In *CVPR*, 2005.

[11] R. Arandjelović and A. Zisserman. Smooth object retrieval using a bag of boundaries. In *ICCV*, 2011.

[12] R. Arandjelović and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012.

[13] M. Aubry, B. Russell, A. Efros, and J. Sivic. Seeing 3D chairs: exemplar part-based 2D-3D alignment using a large dataset of CAD models. Technical report, INRIA. Available at http://www.di.ens.fr/∼josef/publications/TR/Aubry13b.pdf, 2013.

[14] M. Aubry, B. Russell, and J. Sivic. Painting-to-3D model alignment via discriminative visual elements. Technical Report hal-00863615, INRIA. Accepted for publication in ACM Transactions on Graphics (TOG). Pre-print available at http://www.di.ens.fr/willow/research/painting_to_3d/texts/Aubry13.pdf, 2013.

[15] Y. Aytar and A. Zisserman. Tabula rasa: Model transfer for object category detection. In *ICCV*, 2011.

[16] G. Baatz, O. Saurer, K. Köser, and M. Pollefeys. Large scale visual geo-localization of images in mountainous terrain. In *ECCV*, 2012.

[17] L. Baboud, M. Cadik, E. Eisemann, and H.-P. Seidel. Automatic photo-to-terrain alignment for the annotation of mountain pictures. In *CVPR*, 2011.

[18] F. Bach and Z. Harchaoui. DIFFRAC : a discriminative and flexible framework for clustering. In *NIPS*, 2007.

[19] S. Bae, A. Agarwala, and F. Durand. Computational rephotography. *ACM Transactions on Graphics*, 29(3), 2010.

[20] L. Ballan, G.J. Brostow, J. Puwein, and M. Pollefeys. Unstructured video-based rendering: Interactive exploration of casually captured videos. *SIGGRAPH*, 2010.

[21] A. Bar Hillel and D. Weinshall. Subordinate class recognition using relational object models. In *NIPS*, 2006.

[22] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. Jordan. Matching words and pictures. *JMLR*, 2003.

[23] E. Bart, I. Porteous, P. Perona, and M. Welling. Unsupervised learning of visual taxonomies. In *CVPR*, 2008.

[24] T. Berg, A. Berg, J. Edwards, R. White, Y. W. Teh, E. Learned-Miller, and D. Forsyth. Names and faces in the news. In *CVPR*, pages 848–854, 2004.

[25] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[26] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005.

[27] D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *NIPS*, 2004.

[28] D. M. Blei, T. Griffiths, M. I. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *NIPS*, 2003.

[29] P. Bojanowski, F. Bach, I. Laptev, Ponce J., C. Schmid, and J. Sivic. Finding actors and actions in movies. In *ICCV*, 2013.

[30] F. Bosché. Automated recognition of 3D CAD model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction. *Advanced engineering informatics*, 24(1):107–118, 2010.

[31] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, 2009.

[32] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *CVPR*, 2010.

[33] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using smart. In *TREC-3 Proc.*, 1995.

[34] P. Buehler, M. Everingham, and A. Zisserman. Learning sign language by watching tv (using weakly aligned subtitles). In *CVPR*, 2009.

[35] D. Chen, G. Baatz, et al. City-scale landmark identification on mobile devices. In *CVPR*, 2011.

[36] W. Choi, Y. Chao, C. Pantofaru, and S. Savarese. Understanding indoor scenes using 3d geometric phrases. In *CVPR*, 2013.

[37] O. Chum and J. Matas. Geometric hashing with local affine frames. In *CVPR*, 2006.

[38] O. Chum, A. Mikulik, M. Perdoch, and J. Matas. Total recall II: Query expansion revisited. In *CVPR*, 2011.

[39] O. Chum, M. Perdoch, and J. Matas. Geometric min-hashing: Finding a (thick) needle in a haystack. In *CVPR*, 2009.

[40] O. Chum, J. Philbin, M. Isard, and A. Zisserman. Scalable near identical image and shot detection. In *CIVR*, 2007.

[41] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total Recall: Automatic query expansion with a generative feature model for object retrieval. In *ICCV*, 2007.

[42] O. Chum and A. Zisserman. An exemplar model for learning object classes. In *CVPR*, 2007.

[43] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *JMLR*, 12:2493–2537, 2011.

[44] T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar. Movie/script: Alignment and parsing of video and text transcription. In *ECCV*, 2008.

[45] T. Cour, B. Sapp, C. Jordan, and B. Taskar. Learning from ambiguously labeled images. In *CVPR*, 2009.

[46] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop*, 2004.

[47] M. Cummins and P. Newman. Highly scalable appearance-only SLAM - FAB-MAP 2.0. In *Proceedings of Robotics: Science and Systems*, Seattle, USA, June 2009.

[48] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, 2005.

[49] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs. In *SIGGRAPH*, 1996.

[50] L. Del Pero, J. Bowdish, B. Kermgard, E. Hartley, and K. Barnard. Understanding bayesian rooms using composite 3d object models. In *CVPR*, 2013.

[51] V. Delaitre, D. Fouhey, I. Laptev, J. Sivic, A. Gupta, and A. Efros. Scene semantics from long-term observation of people. In *ECCV*, 2012.

[52] V. Delaitre, J. Sivic, and I. Laptev. Learning person-object interactions for action recognition in still images. In *NIPS*, 2011.

[53] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.

[54] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros. What makes Paris look like Paris? *ACM Transactions on Graphics (TOG)*, 31(4):101, 2012.

[55] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, 2005.

[56] O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce. Automatic annotation of human actions in video. In *ICCV*, 2009.

[57] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, 2003.

[58] B. Epshtein and S. Ullman. Feature hierarchies for object classification. In *ICCV*, 2005.

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

[60] M. Everingham, J. Sivic, and A. Zisserman. "Hello! My name is... Buffy" – Automatic naming of characters in TV video. In *Proc. BMVC.*, 2006.

[61] M. Everingham, J. Sivic, and A. Zisserman. Taking the bite out of automatic naming of characters in TV video. *Image and Vision Computing*, 27(5), 2009.

[62] M. Everingham and A. Zisserman. Identifying individuals in video by combining 'generative' and discriminative head models. In *ICCV*, 2005.

[63] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin. Liblinear: A library for large linear classification. *JMLR*, 9(1):1871–1874, 2008.

[64] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE PAMI*, 2013.

[65] A. Farhadi, M. Hejrati, A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: generating sentences for images. In *ECCV*, 2010.

[66] A. Farhadi, M.K. Tabrizi, I. Endres, and D. Forsyth. A latent model of discriminative aspect. In *ICCV*, 2009.

[67] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE PAMI*, 32(9), 2010.

[68] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[69] S. Fidler, S. Dickinson, and R. Urtasun. 3D object detection and viewpoint estimation with a deformable 3d cuboid model. In *NIPS*, 2012.

[70] S. Fidler and A. Leonardis. Towards scalable representations of object categories: Learning a hierarchy of parts. In *CVPR*, 2007.

[71] A. W. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. In *ECCV*, volume 3, pages 304–320, 2002.

[72] D. Fouhey, V. Delaitre, A. Gupta, A. Efros, I. Laptev, and J. Sivic. People watching: Human actions as a cue for single-view geometry. In *ECCV*, 2012.

[73] A. Frome, Y. Singer, F. Sha, and J. Malik. Learning globally-consistent local distance functions for shape-based image retrieval and classification. In *ICCV*, 2007.

[74] M. Gharbi, T. Malisiewicz, S. Paris, and F. Durand. A Gaussian approximation of feature space for fast image similarity. Technical report, MIT, 2012.

[75] K. Grauman and T. Darrell. Unsupervised learning of categories from sets of partially matching image features. In *CVPR*, 2006.

[76] P. Gronat, G. Obozinski, J. Sivic, and T. Pajdla. Learning and calibrating per-location classifiers for visual place recognition. In *CVPR*, 2013.

[77] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *CVPR*, 2009.

[78] A. Gupta and L. S. Davis. Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In *ECCV*, 2008.

[79] A. Gupta, A. Efros, and M. Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *ECCV*, 2010.

[80] A. Gupta, P. Srinivasan, J. Shi, and L. Davis. Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In *CVPR*, 2009.

[81] D. C. Hauagge and N. Snavely. Image matching using local symmetry features. In *CVPR*, 2012.

[82] M. Hejrati and D. Ramanan. Analyzing 3d objects in cluttered images. In *NIPS*, 2012.

[83] G.E. Hinton. Learning multiple layers of representation. *Trends in cognitive sciences*, 11(10):428–434, 2007.

[84] D. P. Huttenlocher and S. Ullman. Object recognition using alignment. In *International Conference on Computer Vision*, 1987.

[85] A. Irschara, C. Zach, J.-M. Frahm, and H. Bischof. From structure-from-motion point clouds to fast location recognition. In *CVPR*, 2009.

[86] A. Jain, A. Gupta, M. Rodriguez, and L. Davis. Representing videos using mid-level discriminative patches. In *CVPR*, 2013.

[87] H. Jegou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *CVPR*, 2009.

[88] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE PAMI*, 33(1):117–128, 2011.

[89] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE PAMI*, 34(9):1704–1716, 2012.

[90] J. Jiang and C. Zhai. Instance weighting for domain adaptation in NLP. In *ACL*, 2007.

[91] Y. Jin and S. Geman. Context and hierarchy in a probabilistic image model. In *CVPR*, 2006.

[92] A. Joulin, F. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *CVPR*, 2010.

[93] A. Joulin, F. Bach, and J. Ponce. Multi-class cosegmentation. In *CVPR*, 2012.

[94] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013.

[95] B. Kaneva, J. Sivic, A. Torralba, S. Avidan, and W. T. Freeman. Infinite images: Creating and exploring a large photorealistic virtual space. *Proceedings of the IEEE*, 98(8):1391–1407, 2010.

[96] B. Kaneva, J. Sivic, A. Torralba, S. Avidan, and W. T. Freeman. Matching and predicting street level images. In *ECCV 2010 Workshop on Vision for Cognitive Tasks*, 2010.

[97] L. Karlinsky, M. Dinerstein, and S. Ullman. Unsupervised feature optimization (ufo): Simultaneous selection of multiple features with their detection parameters. In *CVPR*, 2009.

[98] A. Khosla, T. Zhou, T. Malisiewicz, A. A. Efros, and A. Torralba. Undoing the damage of dataset bias. In *ECCV*, 2012.

[99] J. Knopp, J. Sivic, and T. Pajdla. Avoiding confusing features in place recognition. In *ECCV*, 2010.

[100] J. Kopf, B. Neubert, B. Chen, M. Cohen, D. Cohen-Or, O. Deussen, M. Uyttendaele, and D. Lischinski. Deep photo: Model-based photograph enhancement and viewing. *ACM Transactions on Graphics*, 27(5), 2008.

[101] A. Krizhevsky, I. Sutskever, and G.E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[102] M. P. Kumar, P. H. S. Torr, and A. Zisserman. An invariant large margin nearest neighbour classifier. In *ICCV*, 2007.

[103] N. Kumar, P. Belhumeur, and S. Nayar. Facetracer: A search engine for large collections of images with faces. In *ECCV*, 2008.

[104] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[105] I. Laptev and P. Pérez. Retrieving actions in movies. In *ICCV*, 2007.

[106] S. Lazebnik, C. Schmid, and J. Ponce. Semi-local affine parts for object recognition. In *Proc. BMVC.*, 2004.

[107] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[108] Q. Le, M. Ranzato, R. Monga, M. Devin, K. Chen, G. Corrado, J. Dean, and A. Ng. Building high-level features using large scale unsupervised learning. In *ICML*, 2012.

[109] Q. V. Le, W. Zou, S. Y. Yeung, and A. Y. Ng. Learning hierarchical spatio-temporal features for action recognition with independent subspace analysis. In *CVPR*, 2011.

[110] Y. LeCun, L. Bottou, and J. HuangFu. Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR*, 2004.

[111] Y. J. Lee and K. Grauman. Shape discovery from unlabeled image collections. In *CVPR*, 2009.

[112] Y. J. Lee and K. Grauman. Learning the easy things first: Self-paced visual category discovery. In *CVPR*, 2011.

[113] G. Levin and P. Debevec. Rouen revisited – interactive installation, 1999.

[114] J. Lezama, K. Alahari, J. Sivic, and I. Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *CVPR*, 2011.

[115] P. Li, H. Ai, Y. Li, and C. Huang. Video parsing based on head tracking and face recognition. In *CIVR*, 2007.

[116] Y. Li, N. Snavely, D. Huttenlocher, and P. Fua. Worldwide pose estimation using 3D point clouds. In *ECCV*, 2012.

[117] Y. Li, N. Snavely, and D. P. Huttenlocher. Location recognition using prioritized feature matching. In *ECCV*, 2010.

[118] J. Lim, H. Pirsiavash, and A. Torralba. Parsing IKEA objects: Fine pose estimation. In *ICCV*, 2013.

[119] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. SIFT flow: dense correspondence across different scenes. In *ECCV*, 2008.

[120] D. Lowe. Local feature view clustering for 3D object recognition. In *CVPR*, 2001.

[121] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[122] D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3):355–395, 1987.

[123] J. Luo, B. Caputo, and V. Ferrari. Who's doing what: Joint modeling of names and verbs for simultaneous face and pose annotation. In *NIPS*, 2009.

[124] T. Malisiewicz, A. Gupta, and A. Efros. Ensemble of exemplar-SVMs for object detection and beyond. In *ICCV*, 2011.

[125] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009.

[126] M. Marszalek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning object representations for visual object class recognition. In *Visual Recognition Challange workshop, ICCV*, 2007.

[127] T. Mensink and J. Verbeek. Improving people search using query expansions: How friends help to find people. In *ECCV*, 2008.

[128] M. Muja and D. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP*, 2009.

[129] J. L. Mundy. Object recognition in the geometric era: A retrospective. In *Toward Category-Level Object Recognition, volume 4170 of Lecture Notes in Computer Science*, pages 3–29. Springer, 2006.

[130] P. Musialski, P. Wonka, D.G. Aliaga, M. Wimmer, L. van Gool, W. Purgathofer, N.J. Mitra, M. Pauly, M. Wand, D. Ceylan, et al. A survey of urban reconstruction. In *Eurographics 2012-State of the Art Reports*, 2012.

[131] M. Nguyen, L. Torresani, F. De la Torre, and C. Rother. Weakly supervised discriminative localization and classification: a joint learning process. In *ICCV*, 2009.

[132] J.C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *Proc. BMVC.*, 2006.

[133] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[134] B. Ommer and J. Buhmann. Learning compositional categorization models. In *ECCV*, 2006.

[135] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. Technical Report hal-00911179, INRIA. Available at http://www.di.ens.fr/willow/research/cnn/, 2013.

[136] V. Ordonez, G. Kulkarni, and T.L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011.

[137] R. Osadchy, M. Miller, and Y. LeCun. Synergistic face detection and pose estimation with energy-based model. In *NIPS*, 2005.

[138] D. Ozkan and P. Duygulu. Finding people frequently appearing in news. In *CIVR*, 2006.

[139] S. J. Pan and Q. Yang. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345–1359, 2010.

[140] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010.

[141] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007.

[142] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, 2008.

[143] J. Philbin, M. Isard, J. Sivic, and A. Zisserman. Descriptor learning for efficient retrieval. In *ECCV*, 2010.

[144] J. Philbin, J. Sivic, and A. Zisserman. Geometric latent dirichlet allocation on a matching graph for large-scale image datasets. *IJCV*, 2010.

[145] H. Pirsiavash and D. Ramanan. Detecting activities of daily living in first-person camera views. In *CVPR*, 2012.

[146] T. Quack, B. Leibe, and L. Van Gool. World-scale mining of objects and events from community photo collections. In *CIVR*, 2008.

[147] D. Ramanan, S. Baker, and S. Kakade. Leveraging archival video for building face datasets. In *ICCV*, 2007.

[148] J. B. Rapp. A geometrical analysis of multiple viewpoint perspective in the work of Giovanni Battista Piranesi: an application of geometric restitution of perspective. *The Journal of Architecture*, 13(6), 2008.

[149] X. Ren and D. Ramanan. Histograms of sparse codes for object detection. In *CVPR*, 2013.

[150] L. Roberts. Machine perception of 3-d solids. In *PhD. Thesis*, 1965.

[151] M. Rodriguez, I. Laptev, J. Sivic, and J.-Y. Audibert. Density-aware person detection and tracking in crowds. In *ICCV*, 2011.

[152] M. Rodriguez, J. Sivic, I. Laptev, and J.-Y. Audibert. Data-driven crowd analysis in videos. In *ICCV*, 2011.

[153] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3D object modeling and recognition using affine-invariant patches and multi-view spatial constraints. In *CVPR*, 2003.

[154] B. C. Russell, A. A. Efros, J. Sivic, W. T. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, 2006.

[155] B. C. Russell, J. Sivic, J. Ponce, and H. Dessales. Automatic alignment of paintings and photographs depicting a 3D scene. In *IEEE Workshop on 3D Representation for Recognition (3dRR-11), associated with ICCV*, 2011.

[156] B. C. Russell and A. Torralba. Building a database of 3D scenes from user annotations. In *CVPR*, 2009.

[157] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: a database and web-based tool for image annotation. *IJCV*, 77(1-3):157–173, 2008.

[158] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010.

[159] S. Satkin, J. Lin, and M. Hebert. Data-driven scene understanding from 3D models. In *Proc. BMVC.*, 2012.

[160] T. Sattler, B. Leibe, and L. Kobbelt. Fast image-based localization using direct 2D-to-3D matching. In *ICCV*, 2011.

[161] F. Schaffalitzky and A. Zisserman. Automated location matching in movies. *CVIU*, 92:236–264, 2003.

[162] G. Schindler, M. Brown, and R. Szeliski. City-scale location recognition. In *CVPR*, 2007.

[163] G. Schindler, F. Dellaert, and S.B. Kang. Inferring temporal order of images from 3D structure. In *CVPR*, 2007.

[164] C. Schmid and R. Mohr. Local greyvalue invariants for image retrieval. *IEEE PAMI*, 19(5):530–534, May 1997.

[165] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *ICPR*, 2004.

[166] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter. Pegasos: Primal Estimated sub-GrAdient SOlver for SVM. *Mathematical Programming, Series B*, 127(1):3–30, 2011.

[167] E. Shechtman and M. Irani. Space-time behavior based correlation. In *CVPR*, 2005.

[168] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *CVPR*, 2007.

[169] J. Shi and J. Malik. Normalized cuts and image segmentation. In *CVPR*, 1997.

[170] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven visual similarity for cross-domain image matching. In *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 2011.

[171] C. Silpa-Anan and R. Hartley. Localization using an image-map. In *ACRA*, 2004.

[172] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012.

[173] J. Sivic, M. Everingham, and A. Zisserman. Person spotting: video shot retrieval for face sets. In *International Conference on Image and Video Retrieval (CIVR 2005), Singapore*, 2005.

[174] J. Sivic, M. Everingham, and A. Zisserman. "Who are you?" - Learning person specific classifiers from video. In *CVPR*, 2009.

[175] J. Sivic, B. Kaneva, A. Torralba, S. Avidan, and W. T. Freeman. Creating and exploring a large photorealistic virtual space. In *Proceedings of the First IEEE Workshop on Internet Vision, Ancorage*, 2008.

[176] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their location in images. In *ICCV*, 2005.

[177] J. Sivic, B. C. Russell, A. Zisserman, W. T. Freeman, and A. A. Efros. Unsupervised discovery of visual object class hierarchies. In *CVPR*, 2008.

[178] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[179] J. Sivic and A. Zisserman. Efficient visual search cast as text retrieval. *IEEE PAMI*, 31(4):591–606, 2009.

[180] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3D. *SIGGRAPH*, 2006.

[181] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan. Contextualizing object detection and classification. In *CVPR*, 2011.

[182] E. Sudderth and M. Jordan. Shared segmentation of natural scenes using dependent Pitman-Yor processes. In *NIPS*, 2008.

[183] R. Szeliski. Image alignment and stitching: A tutorial. *Foundations and Trends in Computer Graphics and Vision*, 2(1):1–104, 2006.

[184] M. Tapaswi, M. Bauml, and R. Stiefelhagen. "Knock! Knock! Who is it?" probabilistic person identification in tv-series. In *CVPR*, 2012.

[185] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. Convolutional learning of spatio-temporal features. In *ECCV*, 2010.

[186] S. Todorovic and N. Ahuja. Extracting subimages of an unknown category from a set of images. In *CVPR*, 2006.

[187] T. Tommasi, F. Orabona, and B. Caputo. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *CVPR*, 2010.

[188] A. Torii, J. Sivic, and T. Pajdla. Visual localization by linear combination of image descriptors. In *Proceedings of the 2nd IEEE Workshop on Mobile Vision, with ICCV*, 2011.

[189] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi. Visual place recognition with repetitive structures. In *CVPR*, 2013.

[190] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *CVPR*, 2011.

[191] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *CVPR*, 2004.

[192] Trecvid evaluation for surveillance event detection, National Institute of Standards and Technology (NIST), 2008.

[193] P. Turcot and D. Lowe. Better matching with fewer features: The selection of useful features in large database recognition problem. In *WS-LAVD, ICCV*, 2009.

[194] R. Vaillant, C. Monrocq, and Y. LeCun. Original approach for the localisation of objects in images. *IEE Proc on Vision, Image, and Signal Processing*, 141(4):245–250, 1994.

[195] N. Vasconcelos. Image indexing with mixture hierarchies. In *CVPR*, 2001.

[196] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple classifiers. In *CVPR*, 2001.

[197] G. Wang, Y. Zhang, and L. Fei-Fei. Using dependent regions for object categorization in a generative framework. In *CVPR*, 2006.

[198] Y. Wang and G. Mori. A discriminative latent model of image region and object tag correspondence. In *NIPS*, 2010.

[199] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *ICCV*, 2005.

[200] C. Wu, B. Clipp, X. Li, J.-M. Frahm, and M. Pollefeys. 3D model matching with viewpoint invariant patches (VIPs). In *CVPR*, 2008.

[201] J. Xiao, B. Russell, and A. Torralba. Localizing 3d cuboids in single-view images. In *NIPS*, 2012.

[202] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans. Maximum margin clustering. In *NIPS*, 2004.

[203] J. Yang, A. Hauptmann, and M-Y. Chen. Finding person X: Correlating names with visual appearances. In *CIVR*, 2004.

[204] J. Yang, Y. Rong, and A. Hauptmann. Multiple instance learning for labeling faces in broadcasting news video. In *Proceedings of the ACM International Conference on Multimedia*, 2005.

[205] L. Zelnik-Manor and M. Irani. Event-based analysis of video. In *CVPR*, 2001.

[206] L. Zhu, Y. Chen, and A. Yuille. Unsupervised learning of a probabilistic grammar for object detection and parsing. In *NIPS*, 2006.

[207] M. Zia, M. Stark, B. Schiele, and K. Schindler. Detailed 3d representations for object recognition and modeling. *IEEE PAMI*, 2013.

[208] A. Zweig and D. Weinshall. Exploiting object hierarchy: Combining models from different category levels. In *ICCV*, 2007.