

Indoor location estimation using a wearable camera with application to the monitoring of persons at home

Vladislavs Dovgalecs

▶ **To cite this version:**

Vladislavs Dovgalecs. Indoor location estimation using a wearable camera with application to the monitoring of persons at home. Signal and Image Processing. Université Bordeaux 1, 2011. English. NNT : . tel-00669874

HAL Id: tel-00669874

https://theses.hal.science/tel-00669874

Submitted on 14 Feb 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N d'ordre : 4384

T H È S E

PRÉSENTÉE À

L'UNIVERSITÉ BORDEAUX 1

ÉCOLE DOCTORALE DES SCIENCES PHYSIQUES ET DE L'INGÉNIEUR

pour obtenir le titre de

DOCTEUR

SPÉCIALITÉ : Automatique et Productique, Signal et Image

Présentée et soutenue par

Vladislavs DOVGALECS

Indoor location estimation using a wearable camera with application to the monitoring of persons at home

(Localisation à partir de caméra vidéo portée)

Thèse dirigée par Rémi Mégret et Yannick Berthoumieu

préparée à l'IMS, UMR 5218 CNRS, Université de Bordeaux, Groupe Signal et Image

Devant la commission d'examen formée de :

Atilla Baskurt	Professeur INSA Lyon	
François Brémond	Directeur de recherches INRIA, Sophia-Antipolis	*Rapporteur*
Matthieu Cord	Professeur UPMC Sorbonnes Universités	*Rapporteur*
Jenny Benois-Pineau	Professeur Université Bordeaux 1	
Yannick Berthoumieu	Professeur IPB/ENSEIRB-MATMECA	
Rémi Mégret	Maître de Conférences IPB/ENSEIRB-MATMECA	

Contents

1 Introduction **11**
- 1.1 General Background . . . 12
- 1.2 Problem Statement . . . 12
- 1.3 Objectives . . . 14
- 1.4 Challenges and Problems . . . 14
- 1.5 Overview of the Related Work . . . 15
- 1.6 Contributions . . . 16
- 1.7 Thesis Outline . . . 16

2 Baseline location estimation **19**
- 2.1 Introduction . . . 20
- 2.2 Overview . . . 20
- 2.3 Visual Feature Extraction . . . 21
  - 2.3.1 From raw pixel values to higher level information . . . 21
  - 2.3.2 Local descriptors . . . 21
  - 2.3.3 Global image descriptors . . . 22
- 2.4 Feature Conditioning . . . 23
  - 2.4.1 Principal Component Analysis . . . 23
  - 2.4.2 Kernel Principal Component Analysis . . . 24
  - 2.4.3 Laplacian Eigenmaps . . . 25
- 2.5 Classification . . . 26
  - 2.5.1 Support Vector Machine classifier . . . 27
- 2.6 Experiments . . . 29
  - 2.6.1 Image database . . . 29
  - 2.6.2 Choice of visual features . . . 30
  - 2.6.3 Visualization of the data . . . 30
  - 2.6.4 Unsupervised manifold learning for classification . . . 31
- 2.7 Conclusion . . . 34

3 Multiple Information Source Fusion **39**
- 3.1 Introduction . . . 40
  - 3.1.1 Motivation . . . 40
  - 3.1.2 Outline . . . 40
  - 3.1.3 Strategies . . . 40
  - 3.1.4 Application to visual data . . . 41
- 3.2 Early Fusion at Feature Level . . . 41
  - 3.2.1 Kernel Combination Rules . . . 42

3.2.2 Feature space of the weighted kernel sum . . . 43
3.3 Late Fusion at Classification Level . . . 44
3.3.1 Brief Categorization of Fusion Architectures . . . 44
3.3.2 Notion . . . 45
3.3.3 Baseline : Majority Voting . . . 46
3.3.4 Discriminative Accumulation Scheme . . . 46
3.3.5 Support Vector Machine DAS (SVM-DAS) . . . 48
3.4 Experiments . . . 50
3.4.1 Image Database and Experimental Setup . . . 50
3.4.2 Performance comparison of Early and Late fusion approaches . . . 52
3.4.3 Conclusions . . . 55

4 Time-Aware Co-Training Framework 57
4.1 Introduction . . . 58
4.1.1 Motivation . . . 58
4.1.2 Outline . . . 58
4.2 Learning from labeled and unlabeled data . . . 58
4.2.1 Motivation and Definitions . . . 59
4.2.2 Label Propagation on Graphs . . . 59
4.2.3 Semi-Supervised Support Vector Machines . . . 63
4.2.4 Learning with self-training . . . 65
4.2.5 Learning with co-training . . . 67
4.2.6 Conclusion . . . 69
4.3 Confidence measures . . . 69
4.3.1 Literature review . . . 69
4.3.2 Derivation of confidence measures in the context of the SVM classifier . . . 69
4.4 Proposed Time-Aware Co-Training Framework . . . 71
4.4.1 Temporal Accumulation: Enforcing Temporal Video Continuity constraints . 71
4.4.2 CO-DAS : Semi-Supervised learning from multiple visual features and classifier fusion . . . 72
4.4.3 Time-Aware CO-DAS : Injection of temporal information into the learning loop 73
4.4.4 Proposition of a new Class overlap sensitive confidence measure . . . 74
4.4.5 Strategies for Visual and Temporal Information Fusion . . . 76
4.5 Experiments . . . 77
4.5.1 Data and test setup . . . 77
4.5.2 Study of confidence measures . . . 77
4.5.3 Baseline semi-supervised method results . . . 79
4.5.4 Presentation and discussion of results using time-aware co-training method . 86
4.5.5 Conclusion . . . 89

5 Invariant Visual Features for Image-Based Localization 91
5.1 Introduction . . . 92
5.2 Literature Review . . . 92
5.2.1 Translation invariant Spatial Pyramid Histograms . . . 92
5.2.2 Methods and applications incorporating invariance . . . 93
5.3 Prior information . . . 94
5.4 Invariant Support Vector Machine Formulation . . . 95
5.4.1 Incorporating invariance . . . 96

5.4.2 Linearization of the input space . . . 96
5.4.3 Algorithmic perspective . . . 97
5.5 Experimental Results . . . 97
5.6 Conclusion . . . 99

6 Application to the IMMED Context 101
6.1 Introduction . . . 102
6.2 Context . . . 102
6.3 Description of the corpus . . . 102
6.4 Experimental protocol . . . 104
6.4.1 Visual features . . . 104
6.4.2 Exploitation of annotation . . . 104
6.5 Evaluation on a test case . . . 104
6.5.1 Supervised classification . . . 105
6.5.2 Semi-supervised classification . . . 106
6.6 Computational Cost . . . 107
6.7 Baseline image-based localization results . . . 108
6.8 Conclusion . . . 108

7 Conclusion 111
7.1 Overview of the work . . . 111
7.2 Contributions . . . 112
7.2.1 Place of feature selection in location estimation . . . 112
7.2.2 Evaluating early and late fusion strategies with multiple features for localization 112
7.2.3 Exploiting unlabeled data and temporal continuity of the video . . . 112
7.2.4 Proposing a translation invariant global descriptor . . . 113
7.3 Future perspectives . . . 113

A Feature Conditioning and Classification 115
A.1 Details on Principal Component Analysis . . . 115
A.2 Details on Kernel PCA . . . 116
A.3 Details on Laplacian Eigenmaps . . . 118
A.4 Elements of Bayesian decision theory . . . 120
A.5 Linearly separably and non-separable data . . . 122
A.5.1 Problem for Linearly separable data . . . 122
A.5.2 Non-separable case of SVM . . . 123

B Multiple Feature exploitation 125
B.1 Details on Multiple Kernel Learning . . . 125

C Details on Semi-Supervised Learning 127
C.1 Semi-Supervised Laplacian SVM . . . 127

D The IDOL2 database 129

E The IMMED corpus 131
E.1 Overview . . . 131
E.2 Acquisition protocol . . . 131
E.3 Topological locations . . . 132

E.4 Evaluation protocols 132

Bibliography **133**

F List of Contributed Articles **145**
F.1 Published and accepted 145

Abstract

Visual lifelog indexing by content has emerged as a high reward application. Enabled by the recent availability of miniaturized recording devices, the demand for automatic extraction of relevant information from wearable sensors generated content has grown. Among many other applications, indoor localization is one challenging problem to be addressed.

Many standard solutions perform unreliably in indoors conditions or require significant intervention. In this thesis we address the problem of localization from the perspective of image-based approach using wearable video camera sensors. The key contribution of this work is the development and the study of a location estimation system composed of diverse modules, which perform tasks ranging from low-level visual information extraction to final topological location estimation with the aid of automatic indexing algorithms. Within this framework, important contributions have been made by efficiently leveraging information brought by multiple visual features, unlabeled image data and the temporal continuity of the video.

Early and late data fusion were considered, and shown to take advantage of the complementarities of multiple visual features describing the images. Due to the difficulty in obtaining annotated data in our context, semi-supervised approaches were investigated, to use unlabeled data as additional source of information, both for non-linear data-adaptive dimensionality reduction, and for improving classification. Herein we have developed a time-aware co-training approach that combines late data-fusion with the semi-supervised exploitation of both unlabeled data and time information. Finally, we have proposed to apply transformation invariant learning to adapt non-invariant descriptors to our localization framework.

The methods have been tested on controlled publicly available data sets to evaluate the gain of each contribution. This work has also been applied to the IMMED project, dealing with activity recognition and monitoring of the daily living using a wearable camera. In this context, the developed framework has been used to estimate localization on the real world IMMED project video corpus, which showed the potential of the approaches in such challenging conditions.

Résumé

L'indexation par le contenu de lifelogs issus de capteurs portés a émergé comme un enjeu à forte valeur ajoutée, permettant l'exploitation de ces nouveaux types de donnés. Rendu plus accessible par la récente disponibilité de dispositifs miniaturisés d'enregistrement, les besoins se sont accrus pour l'extraction automatique d'informations pertinentes à partir de contenus générés par de tels dispositifs. Entre autres applications, la localisation en environnement intérieur est l'un des verrous que nous abordons dans cette thèse.

Beaucoup des solutions existantes pour la localisation fonctionnent insuffisamment bien ou nécessitent une intervention manuelle importante. Dans cette thèse, nous abordons le problème de la localisation topologique à partir de séquences vidéo issues d'une camera portée en utilisant une approche purement visuelle. Ce travail complète d'extraction des descripteurs visuels de bas niveaux jusqu'à l'estimation finale de la localisation à l'aide d'algorithmes automatiques.

Dans ce cadre, les contributions principales de ce travail concernent l'exploitation efficace des informations apportées par des descripteurs visuels multiples, par les images non étiquetées et par la continuité temporelle de la vidéo. Ainsi, la fusion précoce et la fusion tardive des données visuelles ont été examinées et l'avantage apporté par la complémentarité des descripteurs visuels a été mis en évidence sur le problème de la localisation. En raison de difficulté à obtenir des données étiquetées en quantités suffisantes, l'ensemble des données a été exploité ; d'une part les approches de réduction de dimensionnalité non-linéaire ont été appliquées, afin d'améliorer la taille des données à traiter et la complexité associée; d'autre part des approches semi-supervisés ont été étudiées pour utiliser l'information supplémentaire apportée par les images non étiquetées lors de la classification. Ces éléments ont été analysé séparément et ont été mis en œuvre ensemble sous la forme d'une nouvelle méthode par co-apprentissage avec information temporelle. Finalement nous avons également exploré la question de l'invariance des descripteurs, en proposant l'utilisation d'un apprentissage invariant à la transformation spatiale, comme une autre réponse possible au manque de données annotées et à la variabilité visuelle.

Ces méthodes ont été évaluées sur des séquences vidéo en environnement contrôlé accessibles publiquement pour évaluer le gain spécifique de chaque contribution. Ce travail a également été appliqué dans le cadre du projet IMMED, qui concerne l'observation et l'indexation d'activités de la vie quotidienne dans un objectif d'aide au diagnostic médical, à l'aide d'une caméra vidéo portée. Nous avons ainsi pu mettre en œuvre le dispositif d'acquisition vidéo portée et montrer le potentiel de notre approche pour l'estimation de la localisation topologique sur un corpus présentant des conditions difficiles représentatives des données réelles.

Acknowledgments

First of all, I would like to say thanks to God, Creator of everything visible and invisible, for all His blessings, health and defense throughout all these years in Bordeaux, France. He gave me the capacity and possibility to study, meet new people and see new place besides my more basic daily needs. Thanks for His Word which is always actual and necessary as a loaf of bread in my daily life.

I will always remember my stay with Signal and Image Group, LAPS laboratory at IMS, University of Bordeaux. My thanks go to Y. Berthoumieu, director of thesis, and R. Megret, co-director of thesis, for having accepted as their PhD student and their hospitality. I would like particularly say thanks to Remi for his constant guidance, patience and help more that an ordinary student can expect. Without his help and constant guidance the research and thesis writing would be of much lower quality.

I am also expressing my thanks and gratitude to my office room mates as well as other students, post-docs who I know from the floor 5 and 6. Thanks to Guillaume, JB, Fred, Julien, Hector and the girls Asma and Nelly for constantly appreciating the kitchen of the chief Gérard at Haut Carré and spending our meal time there. Thanks also to Guillaume from Limoges for his unique jokes, simple Audrey, always cheerful Jean-François, Marc with his ND and pragmatic Jean-Pierre. Thanks also to administrative stuff for being helpful and kind despite great pressure and being always busy.

My most warm thanks go to my family, father and mother, all my brothers Viktors, Jurijs and Edvards as well as my both sisters Oksana and Ieva (also known as Ievns with her two cats and funny Skype conversations).

Special place in my thoughts occupies my best friend Galina Portuzenkova for her constant support, understanding and simply being родная душа! You were simply there in both joyful and hard moments when so many friends came and left. May the God pay back you ten times what you did for me! Thank you from all my heart!

Chapter 1

Introduction

Contents

1.1 General Background **12**
1.2 Problem Statement **12**
1.3 Objectives **14**
1.4 Challenges and Problems **14**
1.5 Overview of the Related Work **15**
1.6 Contributions **16**
1.7 Thesis Outline **16**

1.1 General Background

Recent growth of image, video and sound recordings resulted in the explosion of different multimedia applications. Large quantities of image and video materials are generated every day which is a wealth of information that can be utilized in different domains such as education, social, healthcare, security and many others. Indeed, image and audio data carries a substantial amount of information remaining however largely unexploited. Many problems to address concern object recognition, face detection and recognition, optical character recognition, visual scene semantic analysis, among others.

A particular place occupies image and video retrieval solutions. The challenge lies in abundant and ever increasing amounts of materials as well as in the efficient understanding and semantic description of the content. The former calls for efficiency while the latter involves a difficult problem of image and video understanding. The area is in active phase of research and promises exciting possibilities for information retrieval from large corpuses of videos. The applications are numerous among them the news video search, film archive exploration, surveillance video browsing, medical applications, sports video analysis, distant learning and video conferencing. The main challenge apart from the issue of efficient large to very large video corpus processing is the learning of semantically rich content.

With the advent of digital computers and ever increasing computing powers, special place is devoted to increase the wellfare of population. Computer aided medical assistance and treatment now became a mainstream in many hospitals and clinics. Visual lifelogs have emerged as a practical aid for the daily activity recording and monitoring of the every day life. One of the practical utilities of the visual lifelogs is that the wearer's position and activities can be effectively refreshed in person's memory from the recorded wearable video or used as a monitoring device for analysing the activities of patients in a medical context. With time, such recording archives tend to grow in size making it more and more difficult for browsing efficient navigation. It is therefore clear that computerized aid for automatic organization of visual lifelogs is crucial for the succesful exploitation of such data.

1.2 Problem Statement

Automatic video lifelog indexing is a very broad topic, which covers image analysis and characterization, video processing, scalability issues with large data corpuses, and practical hardware design details.

Knowing the localization at each moment of the lifelog is an important information with potential semantics concerning the type of activities or the structure of the lifelog. In turn this information can be used as an input for video content structuration. Once the video content has been structured, a video browsing or navigation can then be implemented as a special browsing interface.

When there is a need for localization services, one very often thinks about GPS (Global Positioning System). The GPS is a system widely used for navigation outdoors but is of very limited use indoors due to the loss of precision caused primarily by the loss or strong attenuation of the signal. Solutions such as Ultra-wide Band localization and RFID for localization indoors may require an exhaustive coverage of the areas of interest with transmitters or base stations (e.g. WiFi access points) to be sufficiently precise. Together with fixed installation videosurveillance cameras and presence sensors installed in pre-defined areas, considerable intervention and material support may be required for these solutions. In cases when minimum intervention and high portability is required, these solutions may not be well suited. A potential solution is to use vision-based approaches since a considerable progress has been achieved in recording device compactness and in the quality of the

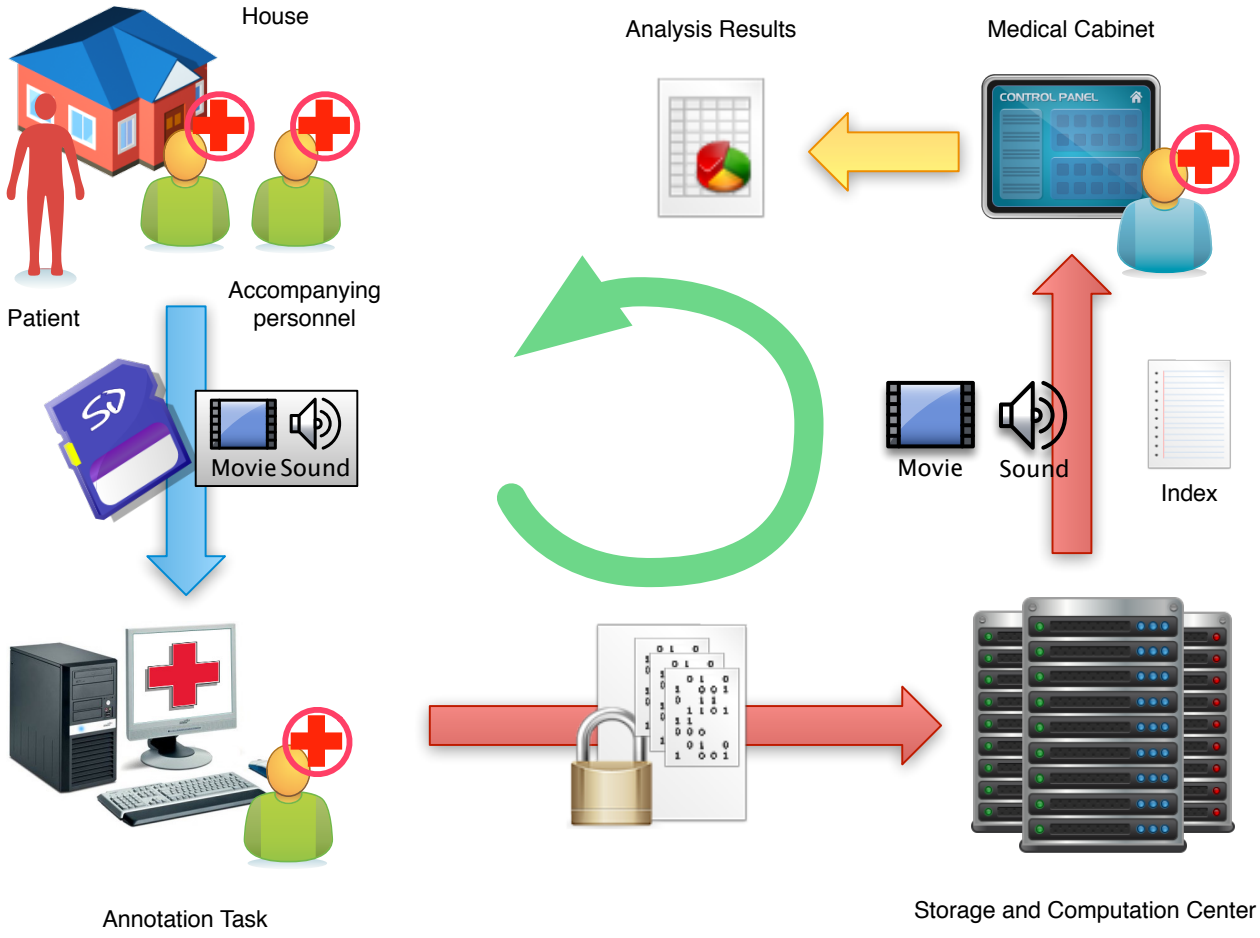


Figure 1.1: General components and workflow in a medical evaluation context



Figure 1.2: Video lifelog recording device used in the IMMED project, which is one applicative context of this work: (left) vest with autonomous video camera; (right) detachable video camera composed of camera, battery and memory card

recordings. This approach presents the advantage of exploiting directly the video data captured for the visual lifelog, thus simplifying the logistics related to the acquisition.

In this thesis we therefore propose to tackle the problem of image-based indoors localization. We are interested in topological localization of a wearable camera monitored person within a house or in its limited neighborhood, where GPS or similar techniques do not apply. Practically, such localization is related to the indexing of the lifelog stream with a spatial location such as "living room", "kitchen", "bathroom" and similar. This work will find applications in the context of the IMMED project (described in more details in Chapter 6), where patients are monitored using a wearable camera. The final goal of this project is to record patient activities in their ecological environment in order to help medical specialists better understand and diagnose the signs of early dementia, such as found in the Alzheimer disease.

1.3 Objectives

A complete processing chain representative of lifelog exploitation is depicted in Fig. 1.1 using the context of the IMMED project as an example. The whole path from video recording creation to the exploitation of the results is demonstrated. Four major phases or stages are shown in the workflow:

1. audio/video material acquisition;
2. annotation;
3. automatic video indexing;
4. indexed content navigation and analysis using a specialized interface.

The main body of our work concerns the third point: automatic video indexing. The goal of this thesis is to build algorithms for automatic indexing of visual lifelog, taking as input annotated and un-annotated videos, and compute the topological localization indexing on the latter.

There are multiple objectives in order to solve the image-based localization problem. In general, the whole process can be summarized from data acquisition and pre-processing to computation result storage. The following objectives have been established:

1. Acquire test video materials and extract visual features;
2. Study the related image-based localization solutions that apply to our context;
3. Develop image-based localization algorithms and perform validation on controlled environment visual data;
4. Evaluate the developed algorithms on audio-visual data of practical interest.

1.4 Challenges and Problems

Video content indexing by content is an extremely challenging problem. Natural scenes are very rich and very complex sources of information featuring large variability. Nowadays, indexation of events captured using video and audio recording devices in weakly constrained environments is one of the hardest tasks in the multimedia community. As part of the study and development of solutions to the stated objectives, in this thesis we outline two big challenges to which we attempt to answer:

1. How to perform automatic video indexing in low supervision conditions;

2. What additional information or knowledge is available and if it can be used.

The performance of any indexing system is first of all directly linked to the amount of available training data. In the present study we focus on learning from annotated videos in rather low supervision scenarios. The challenge lies in the efficient exploitation of all the available data and a priori knowledge.

1.5 Overview of the Related Work

Image-based approaches Still image data is a rich source of information which has been widely used in many areas. Particular attention from image and multimedia processing communities is given to content-based image retrieval (CBIR) [121, 19, 152, 50], object detection and recognition [149, 122, 18, 20, 48, 96, 159, 49, 79, 14, 63], robotics [114, 113, 112, 116], and more recently, image-based localization [142, 168, 173, 151, 123, 185, 177, 149, 117].

The CBIR systems working with large image databases are typically working in two scenarios: query by example and relevance feedback. The goals of such systems are different from those of image-based localization. CBIR systems require good precision of retrieval and use image data in their queries while for localization we need good recall and keyword based search. Such systems are usually relatively complex and should answer the challenge of efficient database exploration, multimodality of distributions in the feature space as well as relevant visual information extraction. The system proposed in [33] answers the challenge with their system RETIN, featuring adaptive quantization for image signature computation, efficient image exploration in interactive setup with relevance feedback loop and adaptive similarity measure which evolves as a user works with the system.

Robotics is a large domain where one of its tasks is environment understanding, analysis, localization and inference to make robot integration more autonomous in accomplishing practical tasks. The goals and employed methodologies can be similar to those of our tasks, though the requirements as computation power, memory needs and certain robustness properties may be relaxed.

We position this thesis in the group of image-based localization where the related techniques and methods will be discussed in the following chapters.

Video indexing Proliferation of video recording devices resulted in large and mostly unexplored archives from which even more valuable information can be extracted than from still images. Video specific applications include video retrieval [127, 63, 134] from large databases, event recognition [25, 80, 83, 62, 141, 170, 169, 45, 8] in general and in particular human activity recognition [75, 126, 137, 82] from video. We would like to mention the TRECVid [135, 119] which is a yearly competition where video indexing methods are regularly evaluated. One of its tasks is to identify the high level concepts such as “classroom”, “airplane flying”, “two people” and other. The goal is to return a ranked list of image shots from the test collection such that possibility of a high level concept is the highest.

In this thesis the primary data source are video recordings. Our goal is not to extract high level concepts as in very briefly reviewed works but rather to perform image-based localization from wearable video.

Semantic place classification Place classification can be seen as a pattern recognition problem where a scene is assigned to a class learned by a model. Two large groups can be identified for image-based localization: place recognition and place categorization. In the former, the trained

model is applied on the testing data captured in the same environment as the training data whereas in the latter the testing data may have been acquired in a new environment.

Image-based localization approaches can also be differentiated by the source such as spatial reference point, scale or abstraction of image data. Metric map [41, 167] uses a set of coordinates to define the location of a particular shot and topological localization [151, 93, 117, 142] uses an abstract metric space composed of discrete units. The approaches using both localization types are termed hybrid [148, 22, 177].

In this thesis we do not attempt to generalize the class concepts such as "living room", "corridor", neither we use precise reference coordinate systems for each image of the video.

Use of prior information in machine learning Success of any non-trivial learning is highly dependent on the a priori knowledge of the problem. In a nutshell, prior information allows to restrict the space of possible solutions. For example, large variability of visual appearance, lighting and the noise has been countered in face recognition by enforcing scale and rotation invariance [108]. It is similar for Optical Character Recognition (OCR) task [30, 58] and for object classification [110, 53] using tangent approximation approach.

In this thesis, we contribute a novel translation invariant descriptor, which improves a state-of-the-art descriptor in scene classification and recognition. Its usefulness was demonstrated in low supervision conditions.

1.6 Contributions

In this thesis we make multiple contributions to answer the challenges within image-based localization problem from wearable video. We adopt the point of view of pattern recognition to solve this problem and evaluate how to take into account prior knowledge that is specific to the problem. The work is organized around a complete processing chain comprised of multiple modules. The processing chain consists of algorithms starting from low level visual data extraction up to the final system output, which is estimation of topological location of the wearable camera wearer. Several contributions were made in the decision part of the processing chain: profiting from large quantities of un-annotated video frames, complementarity of visual descriptors and temporal continuity of the video content. Another contribution concerns the exploitation of the available knowledge where we demonstrate an improved performance of the system by constructing a new visual descriptor with this information taken into account.

1.7 Thesis Outline

The thesis is composed of the following chapters :

Chapter 1: Introduction In the current chapter the problem of image-based localization is introduced. We show how it relates to the general context of lifelog analysis and what objectives and challenges are faced.

Chapter 2: Baseline location estimation One of the first steps for visual place recognition is to extract relevant visual features. State-of-the-art visual features are often of high dimensionality. From a Statistical Learning perspective this poses a serious learning problem, especially in low supervision scenarios and from a computational point of view. We investigate the relevance of

unsupervised dimensionality reduction known also as feature selection which render numerical image representations more compact, and the use of classification approaches.

Chapter 3: Multiple Information Source Fusion Experimental results showed the efficiency and confirmed the usefulness of pre-processing unsupervised data on simple test databases. However, none of the state-of-the-art visual features provides sufficient discriminant power. To tackle this problem, early and late fusion approaches are thoroughly evaluated in the task of indexing.

Chapter 4: Time-Aware Co-Training Framework for Image-based Localization Sufficient amount of annotation is crucial to succeed with real-world video indexing. Unfortunately it is costly and therefore we investigated in this chapter how to benefit from both labeled, unlabeled and a priori information to improve the baselines. We evaluated several semi-supervised learning methods and contributed with a unifying learning framework based on the co-training paradigm, that leverages multiple visual features, unlabeled data and takes into account the temporal structure of the visual content.

Chapter 5: Invariant Visual Features for Semi-Supervised Localization Apart from temporal information inherent to every video recording, there is some additional prior information that can be exploited. We contribute with original idea and application by introducing translation invariant visual features based on discriminant non-invariant features and show the gain of this approach in the context of localization.

Chapter 6: Experimental Results on IMMED Project Data We devote this chapter to evaluate the best performing indexing algorithm on the video recordings carrying practical interest. These recordings were carried in ecological environment by multiple volunteers and potential patients to provide some valuable test data having also practical interest for the medical specialist. The data is considered as challenging and is seen as a great opportunity to validate the top performing baseline algorithms as well as proposed algorithms to point out any undisclosed difficulties.

Chapter 2

Baseline Location Estimation

Contents

2.1 Introduction **20**
2.2 Overview **20**
2.3 Visual Feature Extraction **21**
2.4 Feature Conditioning **23**
2.5 Classification **26**
2.6 Experiments **29**
2.7 Conclusion **34**

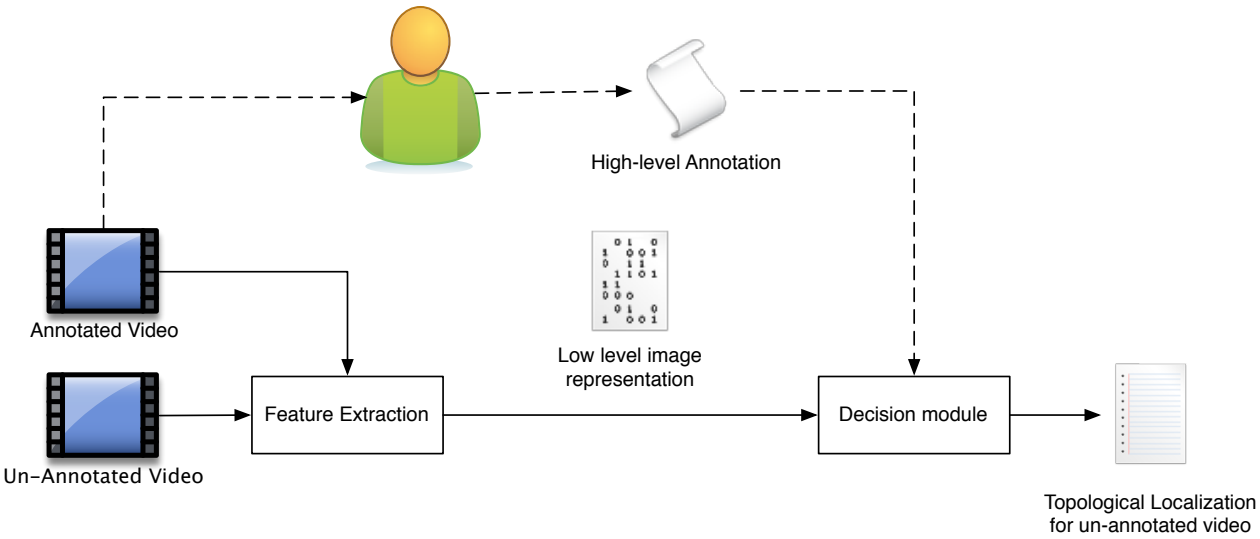


Figure 2.1: Baseline processing flow: from video to topological localization

2.1 Introduction

In this chapter we introduce all the necessary elements for image-based localization in visual lifelogs that will serve as baseline for the localization problem. These elements will be further developed and extended throughout the thesis with new methods and modules.

A whole chain of diverse algorithms should be presented such that the raw numerical video sequence data could be translated into semantically high level decision of video camera wearer's topological localization using indexing algorithms. This yields a baseline processing chain, which takes as input video data and returns a final topological localization output. The processing chain consists of visual feature extraction algorithms followed by compact image representation computation and decision making module.

2.2 Overview

To achieve the task of image-based localization, several components or elements are necessary. First, video sequence data should be prepared and provided with a small but sufficient annotation. Secondly, relevant information extraction should be conducted from the raw video data and prepared for the decision making stage. Finally, based on numerical representations of the visual content and given annotation for a part of the data, the decision module should provide the final topological localization output. The output can be seen as a human-readable tag or index for the video data such that it can enable video content browsing.

In Fig. 2.1 a common processing work flow is shown where two important aspects are shown: manual high-level annotation creation and automatic indexing algorithm. The automatic algorithm may basically contain only visual feature extraction and decision module.

In the current chapter we address briefly visual feature extraction in Section 2.3 and then proceed with discussion on feature selection in Section 2.4 and decision module in Section 2.5. The experimental evaluation for the baseline localization is presented in Section 4.5.

2.3 Visual Feature Extraction

Classically all image data is stored in a numeric format in the memory of a computer. The image is represented as an ordered collection of pixels composed usually of red, green and blue components or channels. Numerically, the components of each color are three values ranging from 0 (dark) to 255 (light) and only after displaying on a screen give familiar perception of the color. In case of gray level images, there is a single component with each pixel value ranging numerically from 0 (black) to 255 (white).

2.3.1 From raw pixel values to higher level information

Given two or more images and asked to compare the content, the straightforward approach is to compare the corresponding pixel values. Unfortunately, this approach is highly unreliable since it can detect only duplicate images and is computationally very costly. Therefore, certain assumptions should be made and only relevant numerical information should be extracted. The numerical representation is required to be compact and when compared to similar content images should return high similarity measure, and low measure for different content images.

It is important to outline the fact that the visual feature extraction process attempts to capture relevant information from images which is often high level (objects, scene, action) to be of practical utility. An attempt in transition from raw numerical pixel values to higher level information can be color, shape, region and texture description. Specific descriptors may capture edge or dominant direction information. This gives a rise to a large variety of description approaches, which are often application specific. We now review descriptors that have proven to be useful for location recognition and for image recognition in general.

2.3.2 Local descriptors

Local invariant descriptors has received particular interest from the multimedia community. The power of these descriptors in in their repeatability and robustness properties to various transformations. The well known Scale Invariant Feature Transform (SIFT) [84] descriptor has been used widely for object and scene recognition in multiple contexts where it demonstrated its robustness to scale, rotation and affine transformations to certain degree. Enhancements such as to capture the information about color were proposed using different methods in [122]. Comprehensive study comparing different descriptors with respect to different interest regions types and matching approaches was carried out in [92]. Issues of imaging conditions such as light changes were studied in [24] and comparing standard grey-color descriptors to color descriptors.

The visual information of an image is captured when a set of local descriptors is found in predefined (e.g. regular or dense sampling) or in automatically detected (e.g. feature detectors) regions. Owing to their repeatability property, a straightforward approach to image comparison would be to compare every possible pair of descriptors also known as matching. The idea is to count the number of matches linking the local descriptors in two images. Larger number of links suggests for similar visual content images. The false matches are usually removed by removing ambiguous matches and enforcing additional spatial constraints. To remove the ambiguous matches, [84] proposed to compute the ratio between the the best and the second best match and impose that the ratio of these descriptor affinities should exceed some threshold [84]. Spatial constraints were enforced in [134] to be able to detect the region or object of interest in the video.

To speedup the query process of an image with respect to the database of images, a vocabulary tree approach has been proposed in [32] for localization estimation with an extension to inverted files such that the tree provides also the votes from which image the matching descriptor takes its origins.

Object search in video was proposed in [134] where the vocabulary tree serves to detect the matches between the regions automatically and the spatial constraints helps to remove false matches.

2.3.3 Global image descriptors

Local feature matching-based approaches are simple but also computationally expensive since many local feature comparisons could be necessary. The scalability of these approaches become quickly an issue and renders it unfeasible in large-scale applications if applied directly. To address this issue, images can be also represented by global descriptors. The selection of the global descriptors include:

- Bag of Visual Words (BOVW)
- Composite Receptive Field Histograms (CRFH)
- Spatial Pyramid of Histogram (SPH)

Attempts to build global image descriptors or signatures had been actively researched as early as in [88] and [51] in the context of image retrieval. These works investigated Gabor filter features, color features to build color histograms and others.

Bag of Visual Words The Bag of Visual Words (BOVW) global descriptor [100] stems from the domain of text retrieval domain by counting occurrences of visual words found in an image. Visual words are obtained from invariant local features (SIFT or SURF using a feature detector) that are quantized into a finite set of words according to a common vocabulary. The vocabulary is commonly built from a large set of available local features using a clustering approach. Clustering effectively reduces the number of available words and provides a restricted set of available "words" that can be found in an image. Therefore, an image numerical representation is a histogram of visual word occurrences. Due to local feature stability and repeatability property, visually similar content images will result in similar histograms.

In our application, the BOVW histograms are 1111 dimensional vectors. The visual vocabulary was built in a hierarchical manner [100] with 3 levels and 10 sibling nodes to speed up the search of the tree. This allows to introduce visual words ranging from more general (higher level nodes) to more specific (leaf nodes). The effect of overly frequent visual words is addressed with the use of common normalization procedure tf-idf [100] from text classification.

Composite Receptive Field Histograms The CRFH [79] global descriptor describe a scene globally counting responses produced by a specific filter being applied on an image. Every dimension of this descriptor effectively counts the number of pixels sharing similar responses to a particular filter. The filter can be a color histogram, first or second order partial derivatives at different scales, gradient magnitudes and directions among multiple other possibilities. Attractiveness of such a descriptor is that different properties of a scene can be captured.

Due to multidimensional nature and the size of an image, such descriptor often results in a very high dimensionality vector. In our experimental evaluations we used second order derivatives filter in three directions, at two scales with 28 bins per histogram. The total size of global descriptor resulted in very sparse up to 400 million dimension vectors.

Spatial Pyramid Histograms The SPH [1, 77] global descriptor harnesses the power of the BOVW descriptor but addresses its weakness when it comes to spatial structure of the image. This is done by constructing a pyramid where each level defines coarse to fine sampling grid for histogram

extraction. Additionally, each grid histogram is obtained by constructing a BOVW histogram with local features SIFT sampled in a dense manner. The final global descriptor is composed of concatenated individual region and level histograms.

We empirically set the number of pyramid levels to 3 with the dictionary size of 200 visual words, which yielded in 4,200 dimensional vectors per image.

2.4 Feature Conditioning

Real world data is often high dimensional and there are often redundant measurements mainly due to the lack of the knowledge about the data. The simplest approach to reduce redundancy and select the most meaningful features is using a linear transformation. We introduce and discuss the suitability of Principal Component Analysis (PCA) [66, 130] for image-based localization.

Compared to linear approaches, non-linear methods are usually more powerful since the representation using latent variables may exploit non-linear relationships between observed values. Often this is true for complex data like visual information. The non-linearity in input space is addressed elegantly with the use of kernels. Prohibitively high computation demands and complexity of modeling the input space are alleviated with the use of implicit non-linear mapping. Mapped patterns in the feature space are then used for feature extraction, classification and novelty detection. We introduce Kernel Principal Component Analysis (KPCA) as well as Laplacian Eigenmaps (LapEig) and demonstrate their use for conditioning the features.

Linear transformation Let $X = (\mathbf{x}_i)_{i=1}^n \in \mathcal{R}^{d \times n}$ be a matrix with its patterns organized in columns. Given a transformation matrix $A^{k \times n}$ (with the coefficients of the linear mapping for each output dimension in columns $A = (\mathbf{a}_1, \ldots, \mathbf{a}_k)$) and $k < d$, linear transformation for a pattern $\mathbf{x}_i$ writes as

$$\mathbf{z}_i = A^T \mathbf{x}_i \tag{2.1}$$

or in matrix form

$$Z = A^T X \tag{2.2}$$

where the columns in the matrix $Z = (\mathbf{z}_i)_{i=1}^{k \times n}$ are the pattern embeddings $\mathbf{z}_i \in \mathcal{R}^k$. Therefore, a pattern $\mathbf{x}_i$ living in a d-dimensional space is now represented by a pattern $\mathbf{z}_i$ in $k < d$ dimensions. Methods using linear transformation differ in the criterion used to assess the fitness of the transformation matrix A to a particular task.

2.4.1 Principal Component Analysis

The method of Principal Component Analysis (PCA) [66, 130] is a linear and unsupervised method often used for data dimensionality reduction. PCA is one of the simplest methods and is often used when no prior knowledge on the data is known.

The goal of PCA is to recover the hidden low dimensional structure of the complex data with respect to variance. More precisely, it finds an orthogonal projection subspace such that the variance of the data is maximized

$$\arg\max_{\mathbf{a}} \mathbf{a}^T S \mathbf{a} \quad \text{s.t.} \ \|\mathbf{a}\|_2 = 1 \tag{2.3}$$

where the $\mathbf{a}$ is a mapping vector and the S is a sample covariance matrix.

PCA is a linear method where such data transformation can be seen in the framework of Eq. 2.2. In particular, highly correlated dimensions can be reduced by keeping only a linear combination of them, thus reducing redundancy of the data.

The technical details on linear dimensionality reduction using the PCA method are presented in Annex A, in Section A.1.

2.4.2 Kernel Principal Component Analysis

KPCA [136] is a non-linear extension of linear PCA. An appropriate selection of a kernel function k, that computes affinities between the patterns, allows to define a feature space $\mathcal{H}$ in which linear operations can be performed, and that is related to the original space by a non-linear mapping function Φ. Finding the largest variance directions can be defined inside linear space $\mathcal{H}$.

As for linear PCA, the goal is to obtain the pattern embeddings in less dimensions such that the data is represented the best in terms of variance.

There is a need to introduce some definitions such as the concepts of kernel and Reproducing Kernel Hilbert Space (RKHS).

The notion of kernel and its positive definiteness property Given two patterns $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^d$, a simple similarity measure can be that of dot product

$$\left\langle \mathbf{x}, \mathbf{x}' \right\rangle = \sum_{i=1}^{d} \mathbf{x}\left(i\right) \mathbf{x}'\left(i\right) \tag{2.4}$$

In general, a similarity measure (kernel) can be seen as an output of a two argument function

$$k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \tag{2.5}$$

which returns a real value for two given objects from a set $\mathcal{X}$. In the following it is assumed that it is also symmetric such that $k\left(\mathbf{x}, \mathbf{x}'\right) = k\left(\mathbf{x}', \mathbf{x}\right)$ for all $\mathbf{x} \in \mathcal{X}$.

Although there are no restrictions on the mapping functions Φ, not any kernel function is valid. Acceptable kernels satisfy Mercer conditions of positive definiteness. For any two patterns $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$. A kernel function k that is symmetric $k\left(\mathbf{x}, \mathbf{x}'\right) = k\left(\mathbf{x}', \mathbf{x}\right)$ is also positive definite iff

$$\int k\left(\mathbf{x}, \mathbf{x}'\right) f\left(\mathbf{x}\right) f\left(\mathbf{x}'\right) d\mathbf{x} d\mathbf{x}' \geq 0 \tag{2.6}$$

for all functions $f : \mathcal{X} \to \mathbb{R}$ such that

$$\int f^2\left(\mathbf{x}\right) d\mathbf{x} < \infty \tag{2.7}$$

In practice this has the following consequence: if $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X}$, then for any $\forall c_1, \ldots, c_n \in \mathbb{R}$, the following condition holds

$$\sum_i \sum_j c_i k\left(\mathbf{x}_i, \mathbf{x}_j\right) c_j \geq 0 \tag{2.8}$$

A Gram matrix $K = \left(k\left(\mathbf{x}_i, \mathbf{x}_j\right)\right)_{ij}$ computed using this kernel function k is called semi-positive definite (SPD). In practical applications only Gram matrices which are at least SPD are admissible to use with kernel learning methods.

The property of RKHS is important in kernel learning since it uniquely determines a kernel function $k(\mathbf{x}, \cdot)$ in some particular Hilbert space $\mathcal{H}$ (see [125] for more details). In practice $\mathcal{H}$ can be abstracted as the image $\Phi(\mathbf{x})$ with a non-linear, possibly infinite dimensional, mapping function with the property such

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \left\langle \Phi(\mathbf{x}), \Phi\left(\mathbf{x}'\right) \right\rangle \tag{2.9}$$

which is also known as the kernel trick. This property allows considering non-linear similarities as bi-linear dot-products in the associated RKHS and compute linear operations in the infinite dimensional $\mathcal{H}$ as corresponding operations using the kernel function k, which is computable using the available data

$$\begin{aligned} \Phi : \mathcal{X} &\rightarrow \mathcal{H} && (2.10) \\ \mathbf{x} &\mapsto \Phi(\mathbf{x}) && (2.11) \end{aligned}$$

Solving for principal components and embedding creation The non-linear PCA uses the same approach for covariance matrix diagonalization. Due to potentially high dimensionality space induced by the non-linear mapping functions, computation of the covariance matrix and its diagonalization is computationally unfeasible. Fortunately, it is possible to compute compact embeddings using the Gram kernel matrix computed from all the available data patterns.

The derivation details and functionality are described in Annex A, section A.2.

Computational efficiency Dimensionality reduction with KPCA poses several practical issues.

The first and evident issue is the computation and storage of the kernel matrix K. Its computational and storage demands grow quadratically with the number of patterns. Although the kernel matrix can be rendered sparse by some heuristics, the kernel centering procedure in Kernel PCA will render it dense.

The second issue is the eigenproblem resolving issue. It involves an eigendecomposition of a dense $n \times n$ kernel matrix K. It is known that computational complexity is of $O\left(n^3\right)$ to find all the eigenvalues and eigenvectors. Modern numerical methods use efficient iterative algorithms to find k largest eigenvalues eigenvectors. A small number of leading eigenvectors can be computed using an efficient Nystrom approximation [52] where the computation of the whole Gram kernel matrix is not necessary.

2.4.3 Laplacian Eigenmaps

Laplacian Eigenmaps (LapEig) [11] is a spectral unsupervised learning method which aims to preserve the local geometry of the data represented by a graph. A graph is a set of ordered pairs $G = (V, E)$, where V is a set of vertices (nodes) and E is a set of vertices pairs, also called edges. In the following discussions we use only uni-directed graphs.

Building a graph The graph is reflecting a prior knowledge on the data. The nodes represent the objects or entities and the links reflect the similarity between any two objects. In the case of limited knowledge about the domain, one often tries a selection of commonly used graphs as a a starting point:

1. Fully connected graph
   All graph nodes are interconnected to all others. Typically for this graph an edge has higher

weight for a pair of nodes which are more similar. Disadvantage of a complete graph is a high computational cost of construction which grows quadratically with new nodes added. It was observed empirically [183] that graph-based methods are less performing for this kind of graphs.

2. Sparse graph
   Using some selection criteria, only a subset of possible edges are used in this kind of graph. Evidently there is an advantage in lower computational and storage demands. Disadvantage is in the node selection criterion (domain knowledge) and learning of graph any necessary parameters.

3. k-Nearest Neighbor graph
   A special variant of sparse graph family where nodes v_i and v_j are connected if v_i is one of k nearest neighbors of v_j or vice versa. This is a common choice of sparse graphs as its hyper-parameter k controls the density of the graph. Attention should be exercised since a graph may get disconnected for too low values of k. Weighted graphs with small k perform relatively well which may be due to removal of many noisy links compared to a fully connected graph.

A graph with n nodes is often represented in computer memory as an adjacency or affinity matrix W of size $n \times n$. For the reasons that will be presented in the following discussions, graph edge weights $w_{ij} \in \mathcal{R}^+$ and the matrix W needs to be non-negative and symmetric.

Creation of graph embeddings The idea of Laplacian Eigenmaps is to compute a new compact embeddings $\mathbf{z}$ such that node affinities of the graph are respected. The following energy functions should be maximized

$$E(\mathbf{z}) = \frac{1}{2} \sum_{i,j} w_{ij} \|\mathbf{z}_i - \mathbf{z}_j\|^2$$

The technical details for Laplacian Eigenmaps and the notion of graph Laplacian, and its role is presented in Annex A, section A.3.

2.5 Classification

Once low level visual features have been extracted from each image and relevant features have been selected, the final task is to produce a decision concerning localization. In this section we turn to the problem of classification, which given the training data and an unlabeled image representation estimates its class. The class in our context is seen as a particular topological location in indoors environment.

Classification of patterns occupies an important part in machine learning. Suppose a binary classification algorithm is given a set of training patterns $\{\mathbf{x}_i\}_{i=1}^m \in \mathcal{R}^d$ together with their labels or targets $\{y_i\}_{i=1}^m \in \{-1, +1\}$. For an unlabeled pattern $\mathbf{x}_{\text{new}}$, a classification algorithm should assign it to one of the two classes. This task is known as a supervised learning problem.

Widely used methods of classification include but are not limited to neural networks, Gaussian Mixture models, Hidden Markov models, decision trees and Bayesian approaches like Naive Bayes (reviewed in Annex A, section A.4) to name few. In this work we will make use of support vector machines classifier.

Supervised learning methods can be divided in two large groups : generative and discriminative approaches. Generative approaches model each class density distribution and using maximum posterior, provide class estimates. Discriminative approaches are more direct, since they attempt to

find a function of parameters that models the separation between the classes. In case of complex data such as visual content, density estimation of the class-specific density distributions necessary for generative model classifiers can be a more complex task than finding separating hyperplane for two classes. For this reason, we selected discriminative family Support Vector Classifier, which we introduce in the next section and with more details given in Annex A, Section A.5.

2.5.1 Support Vector Machine classifier

The SVM classifier operates in a mapped and possibly infinite dimensionality space where a separating hyperplane can be found easier than in original space.

Historically, Support Vector Machine classifiers were defined in three phases. First it has been proposed that an optimal hyperplane should be constructed such that the training patterns are separated with the largest margin [154]. Then [21] proposed to construct a hyperplane in feature space induced by a kernel function (See Subsection A.5.1 for Hard Margin classifier). Previous two approaches are linear and work for linearly separable data only. In [34] authors propose to address this problem by allowing some patterns to violate the optimal margin constraints (See Subsection A.5.2 for Soft Margin classifier), thus allowing the control of classifier generalization capacity.

Definition of hyperplane and margin From a geometrical point of view a pattern $\mathbf{x}_i$ can be seen as a point in some d-dimensional space. For two class data, one might be able to draw a hyperplane that separates different class patterns. Considering the class of hyper-planes in some dot product space $\mathcal{H}$

$$\langle \mathbf{w}, \mathbf{x} \rangle_{\mathcal{H}} + b = 0 \tag{2.12}$$

where $\mathbf{w} \in \mathcal{H}$ is a normal of a hyperplane and $b \in \mathcal{R}$ an offset from the origin. For points lying on a hyperplane the equality holds as in Eq. 2.12.

Every hyperplane has a positive and a negative side. Such a hyperplane explicitly partitions the space $\mathcal{H}$ in two parts that divides patterns in two groups. It may serve as a criterion two classify a pattern in one or another class by constructing the following decision function

$$f(\mathbf{x}) = \operatorname{sign}\left(\langle \mathbf{w}, \mathbf{x} \rangle_{\mathcal{H}} + b\right) \tag{2.13}$$

A separating hyperplane, for a given training set $\{(\mathbf{x}_i, y_i)_{i=1}^n\}$, is one that produces correct estimates $f(\mathbf{x}_i) = y_i \in \{+1, -1\}$ for all patterns of the training set.

Lets define two parallel hyper-planes on both sides of a hyperplane in the form as in Eq. 2.12. With fixed $\mathbf{w}, \mathbf{x}, b$ by incrementing (positive side) and decrementing (negative side) the variable b by some value, two parallel hyper-planes can be defined. The distance between two parallel hyper-planes corresponds to a margin. It is maximal when one pattern of class $+1$ belongs on positive side hyperplane and a pattern of class -1 to the negative side hyperplane. If $\langle \mathbf{w}, \mathbf{x} \rangle + b = 1$ and $\langle \mathbf{w}, \mathbf{x} \rangle = -1$ are such a pair of separating hyper-planes, the margin of the central hyperplane is

$$\min\left\{\|\mathbf{x}_i - \mathbf{x}\| \,|\mathbf{x} \in \mathcal{H}, \langle \mathbf{w}, \mathbf{x} \rangle + b = 0\right\} = \frac{1}{\|\mathbf{w}\|} \tag{2.14}$$

From statistical learning point of view, the choice of the hyperplane and the decision is based on two facts:

- for linearly separable data, among all hyperplane separating the data, there exists one unique hyperplane with a maximum margin between any training pattern and the separating hyperplane;

- there is a link between the generalization performance of a hyperplane classifier and a hyperplane with maximal margin;

Thus our goal is to find a separating hyperplane with a maximal margin $\Delta_{\max}$ which is at the core of Support Vector Machine classifier.

$$\Delta_{\max} = \max_{\mathbf{w}\in\mathcal{H}, b\in\mathcal{R}} \min\left\{\|\mathbf{x}-\mathbf{x}_i\| \,|\mathbf{x}\in\mathcal{H}, \langle\mathbf{w},\mathbf{x}\rangle + b = 0, i = 1,\ldots,m\right\} \tag{2.15}$$

The technical details to estimate an optimal decision function taking into account these principles, the soft margin SVM are reviewed in Annex A, Section A.5.

Multi-class Classification Classification with a hyperplane implies only two class classification problems. Location recognition is rather a multi-class classification problem. There are three main approaches to extend a two class classifier to a multi-class setup [44] : one-vs-all, one-vs-one and multi-class SVM. Suppose that our data has M classes with at least one pattern per class.

In one-vs-all approach one trains M binary classifiers. For each class ω_i a decision function is built $f^i(\mathbf{x})$. Classification rule is following

$$j = \arg\max_i \left\{f^i(\mathbf{x})\right\} \Longrightarrow \text{assign } \mathbf{x} \text{ to class } \omega_j \tag{2.16}$$

This approach has two issues:

1. It is possible that more than one binary classifier return a positive value which defines ambiguous regions.

2. This technique is dealing with rather asymmetric problem where one class may have much more pattern than another.

One-vs-one is an alternative approach where $\frac{M(M-1)}{2}$ binary classifiers are trained to separate any combination of two classes. The final decision is based on the basis of majority vote. The disadvantage of this approach is that a relatively large number of classes may render this approach expensive due to the high number of class pairs to be considered.

The third approach attempts to solve the problem of simultaneous multi-class separation by modifying the objective function of SVM. Authors in [27] and others have noted that in general no multi-class approach outperforms the others. The choice is often determined by practical reasons like training time, number of classes etc. For more multi-class approaches or their variations, see [44, 139].

Link between SVM, RKHS and the kernel function The linear kernel SVM classifier can be extended to work directly with a data adapted kernel. This will correspond to find a hyperplane in the possibly infinite dimensional space $\mathcal{H}$ but with mapped patterns $\Phi(\mathbf{x})$ induced by a non-linear kernel. The advantage is that no unsupervised processing is needed. We briefly introduce the necessary elements to show that a solution to the classification problem with a linear kernel corresponds to an expansion using the computed kernel values.

In learning problems, the function f to be learned cannot be of any form even if it agrees with the training data. It is necessary to specify precisely what is goal is to be achieved or what criteria to be maximized. This leads to a notion of loss function which attributes some loss to an erroneous decision.

Definition : Denote by $(\mathbf{x}, y, f(\mathbf{x})) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ the triplet consisting of a pattern $\mathbf{x}$, an observation y and a prediction $f(\mathbf{x})$. The the map $c : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \in [0, +\infty)$ with the property $(\mathbf{x}, y, y) = 0$, $\forall \mathbf{x} \in \mathcal{X}$ and $\forall y \in \mathcal{Y}$ which corresponds to a case if a correct decision has been made.

It is natural to select a function which incurs the least possible loss (that is, zero loss) given some training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where $\mathbf{x}_i \in \mathcal{X}$ are the inputs and $y_i \in \mathcal{Y}$ are the expected outputs. In statistical learning the problem then is formulated as minimization of functional of regularized risk

$$R(f) = \min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} c(\mathbf{x}_i, y_i, f(\mathbf{x}_i)) + \lambda \Omega(\|f\|_{\mathcal{H}}) \tag{2.17}$$

First part of the equation reflects the goal to find a function f that commits least possible errors on the training set. Second part of the equation is called a regularization term which is weighted by a parameter $\lambda > 0$. With regularization term in place, this renders the learning problem better conditioned. That is, not all functions f that perform very well (small or zero loss) on the training set may predict equally well on new unseen patterns. This is called generalization property. Thereof, with regularization parameter $\lambda > 0$ one trades between minimization of loss and the simplicity or smoothness which is enforced by $\Omega(\|f\|_{\mathcal{H}})$. With larger parameter λ, smoother or simpler functions will be preferred that is linked to generalization property.

With these elements in place, the Representer theorem states that the minimizer of the regularized risk can take particular form given a kernel function k and its induced RKHS $\mathcal{H}$.

Definition : Denote by $\Omega : [0, \infty) \to \mathcal{R}$ a strictly monotonic increasing function, a set $\mathcal{X}$ and an arbitrary loss function $c : (\mathcal{X} \times \mathcal{Y} \times \mathcal{Y})^n \to \mathcal{R} \cup \{\infty\}$. Each minimizer $f \in \mathcal{H}$ of the regularized risk

$$c((\mathbf{x}_1, y_1, f(\mathbf{x}_1)), \ldots, (\mathbf{x}_n, y_n, f(\mathbf{x}_n))) + \Omega(\|f\|_{\mathcal{H}}) \tag{2.18}$$

admits the following representation of the function

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) \tag{2.19}$$

This result is important since it allows to express a function f, which lives in infinite dimensional space, in a form involving training patterns only. In other words, the solution lies in the span of n particular kernels that are centered on training patterns from the set $\mathcal{X}$. It has particular interest for support vector machines. For more complete treatment of the elements of statistical learning and functional analysis on the subject, please refer to [125].

2.6 Experiments

In this section we demonstrate the performance of the image-based localization system using the extracted visual descriptors, feature selection and classification module.

Given the visual descriptors, we study the effects and properties of the dimensionality reduction applied on the visual data since the goal is to find the best representation of the visual data contained within images. The experiments range from purely illustrative to the best localization results possible with the available methods.

2.6.1 Image database

For this experimental part we chose a publicly available video database KTH IDOL2 [86]. The database features the recordings from two mobile robot platforms ("dumbo" and "minnie") in well

controlled conditions indoors. We arbitrarily limit the choice video sequences taken by the robot platform called "minnie" due to its higher position of the recording camera. There are three different lighting conditions and approximately six month time span between the recordings. The trajectory made by the mobile platform was practically the same throughout different factor changes.

In addition to its public status, these properties make the choice of this database motivated for the experimental part. Refer to [86] for more exhaustive properties and complete technical specification of the database. In Annex E information about the database in the light of test setup is provided.

2.6.2 Choice of visual features

For the evaluation purposes we use the following global image descriptors (reviewed in Section 2.3):

- Bag of Visual Words (BOVW)
- Composite Receptive Field Histograms (CRFH)
- Spatial Pyramid of Histograms (SPH)

The goal of these experiments is to show the advantages and limitations of different dimensionality reduction approaches in an illustrative and comparative study.

2.6.3 Visualization of the data

If we limit to a rigid transformation of the camera in the environment, it might be possible to find as low as six dimensional representation of the data. Data points representing individual images should form a smooth manifold reflecting local similarities and evolution of the visual information in time.

One of the basic and principal assumption of dimensionality reduction is that of data lying on a low dimensional subspace. Though different methods employ different criteria, one should be able to find lower dimensionality representation. We therefore use unsupervised learning methods to project the descriptor data onto 3D space in order to visualize it.

First, we attempt to visualize a short excerpt from the video recording. The recording is several second long and shows the displacement of the robot platform in the selected location labelled "Corridor". For visual comparison, the same visual data was visualized with two different unsupervised methods - Kernel PCA and Laplacian Eigenmaps using as input the BOVW signatures. The Fig. 2.2 visualizes the data cloud for methods KPCA and LapEig respectively. The ensemble of the data points correspond to all the embeddings from the class "Corridor" while the outlined track (green path) outlines the selected video recording with temporal continuity. Two interesting aspects arise from these visualizations:

1. Locality and Compactness
   There is a compact point cloud with relatively small variability that corresponds to scenes depicting the corridor. For example, in bottom panel of Figure 2.2 the main portion of the class "Corridor" is in the compact cloud. In fact, such scenes represent the best the class "Corridor" that we want to be learned for classification task. This is true for both dimensionality reduction methods.

2. Visual Variability
   Apart from the compact point cloud we note a connected trail which evolves relatively far away from it. Corresponding scenes reveal the fact of specific or even confusing information

contained therein. For example, in top panel of Figure 2.2 images C, D and E show the visual content that may not be associated with the class "Corridor" cleanly. Rather, such data points can be considered as noisy and will likely be the cause of the class overlap during the learning and classification. Nevertheless, such data points may be needed for the discovery of the smooth data manifold which is crucial for semi-supervised learning methods in the low supervision conditions.

Secondly, our goal is data preparation for the task of classification. In Fig. 2.3 we visualize the reduced dimensionality representation of whole sequence "minnie_night2" in 3 dimensions using two dimensionality reduction methods. The first insight of the visualization is that of intra-class compactness. Compared to KPCA, LapEig tends to build smaller and more compact class clusters with relatively few distant data points. From previous visualization experience, we may confirm that these distant points form a set of visually very different scenes. Relative size and compactness of the cluster may hint about the visual variability within a class. For example, the class "Corridor" is relatively compact and shows a low degree of overlap with other classes. On the contrary, the class "Two_Person_Office" is rather dispersed and is overlapping with several classes which may be a difficulty for the classifier. This preliminary analysis of the data is limited only three dimensions and the class overlap may get less severe as more dimensions are added.

2.6.4 Unsupervised manifold learning for classification

In order to feed the classifiers with the data of manageable complexity, we now evaluate dimensionality reduction approaches to obtain reduced size embeddings to be using for classification. The methods include linear PCA and its non-linear version KPCA with different kernels as well as Laplacian Eigenmaps using the same kernels. First, we observe the effect of varying the dimensionality of the data. Second, we vary amount of training data supplied for the manifold learning. The former allows to speak about intrinsic data dimensionality while the former informs about the necessary amount of training data to discover such low dimensional manifold.

Intrinsic data dimensionality In this experiment the goal is to discover the intrinsic dimensionality of the data. The whole database was split into three parts : training, testing and validation sets, in the proportions from the total size of the database 50%, 25% and 25% respectively. In total 5 random splits were created. Parameter C for SVM was learned using validation set and the best one was applied on the testing set. Pattern embeddings were computed on the whole database in a transductive setting.

In Fig. 2.4 we depict the classification results using four visual features (BOVW, CRFH, SPH and Match) after KPCA pre-processing with χ^2 kernel and then classified using Nearest Neighbor classifier. At the relatively high supervision level (50% training) and remaining half left for validation (25%) and testing (25%), we note the effect of reduced data dimensionality and the impact of Nearest Neighbor classifier parameter k. First, there exists a certain number of dimensions that describes the data best for classification task - around 100 dimensions. Second, the choice of k for the classifier is not very influential. Basically it reflects the compactness of the classes since the performance does not deteriorate too rapidly as the neighborhood is increased.

The results for dimensionality reduction followed by SVM classifier are summarized in Fig. 2.5. At the current 50% labeling rate we are able to achieve remarkable results.

A simple classifier is the Nearest Neighbor classifier with only one parameter to tune. The performance of such classifier is a good indicator of the relevance of the features used. When applied after dimensionality reduction, it can help us assert if some relevant features were removed or enhances, thus helping to tune the number of dimensions.

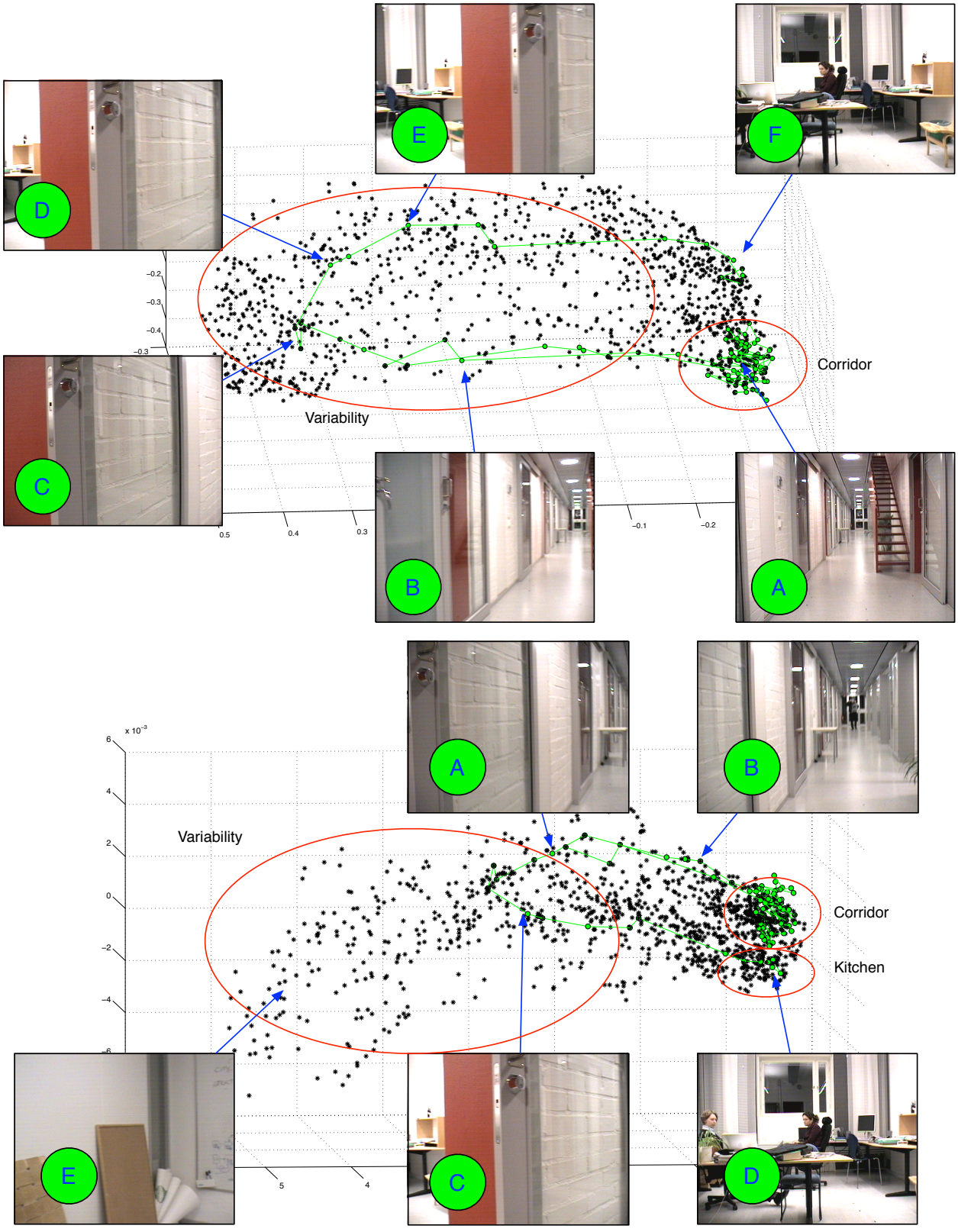


Figure 2.2: KPCA (top), LapEig (bottom) : Sequence "minnie_night2" with a temporal track of class "Corridor"

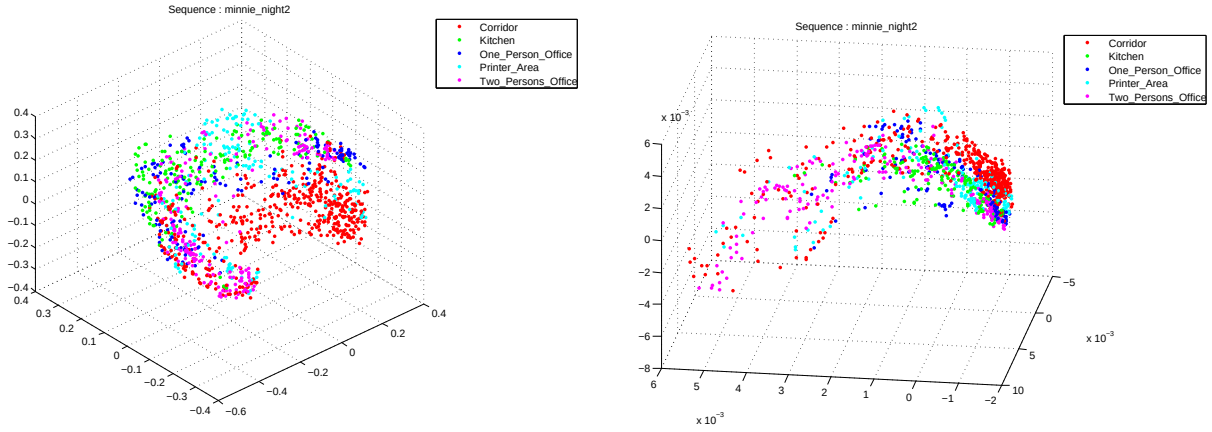


Figure 2.3: KPCA (left), LapEig (right) : Sequence "minnie_night2" with class information

First of all, we note the increase of classification performance as more dimensions are kept. From the results, the underlying manifold is of relatively high dimensionality (around 1000 dimensions) and adding more dimensions is beneficial though less effective once saturation point has been reached.

Secondly, we note the very similar performance of different kernels together with different visual features. Similar performance of three kernels may be explained by the simplicity of the data or generalization capabilities of the classifier though Chi-Square and Intersection kernels tend to perform slightly better. Performances are rather comparable for BOVW, CRFH and Match kernel features.

In the similar setup we employ SVM classifier using the same four visual features with respective performances shown in Fig. 2.5. The results show interesting similarity between the two dimensionality reduction methods and the choice of the kernel. Regardless the feature type, the intrinsic data dimensionality is higher than for Nearest Neighbor classifier result and reaches 1000 dimensions for the best performance. In all cases, a completely linked graph was used with no link pruning in place.

From the results point of view, a simpler Nearest Neighbor classifier outperforms the more complex SVM. The explanation may lie in the complexity of the data. Recall the number of intrinsic dimension for each of the classifiers - it was much higher for SVM than for Nearest Neighbor. It appears that the best separating margin was found in comparably higher dimensional space. Summarizing, the issue may lie in the selection of the kernel used in SVM.

Finally, we note an interesting similarity of performances using KPCA and LapEig dimensionality reduction methods. Recalling the criteria of the dimensionality reduction for both methods and observing the similarity of the results, we may draw a conclusion of equality of the approaches. It seems that criteria of finding the largest variance axes (KPCA) in feature space is comparable to locality preserving properties of the LapEig method in the context of a graph.

SVM : Amount of training data One may argue that in real world conditions it may be difficult to obtain that large labeled data-set. To address this issue, we use the same data and fix its dimensionality to 500 dimensions. Then we vary labeling rate from as low as 1% to 50% in the training set while keeping remaining data as testing and validation sets in equal proportions. Performances are expected to vary for low labeling rates, therefore 10-fold cross-validation procedure was used at validation stage. For the sake of brevity, we use KPCA as a dimensionality reduction

method.

The Fig. 2.6 depicts the global accuracy as the amount of training data supplied to SVM is changed. Starting with as low as 1% of labeled data, we note a relatively low performance of around 60% of correct classification. Such performance drop is characteristic for all visual feature types with low supervision levels. In our application the method should be able to deal with low labelling rates since the annotation provided is very sparse. This problem can be potentially solved by using multiple visual features or semi-supervised learning methods. Low supervision cases are the subject for further discussions in the next chapters.

Sparsification of the affinity matrix Throughout all experiments we used full affinity matrices which correspond to fully connected graphs. Supported by empirical evidence, it might not be necessary to preserve all the links unless we use a very discriminative image descriptor. Moreover, each image in the video can be similar or related to a relatively small part of the whole database. Hence, the full graph may get pruned by preserving the strongest links.

In the current experiment we used 10% of the database as training set and remaining part in two halves - validation and testing. During cross-validation phase both best data dimensionality and best complexity parameter C was used. The study consists in observing the impact of increasing sparcification of the affinity matrix. In our case, we varied the k from full affinity matrix to $k = 10$ and recorded the global accuracy measure. The sparsification procedure uses k-Nearest Neighbor pruning to retain k largest affinity links for each node of the graph then symmetries the affinities by adding missing reciprocal links.

The results are summarized in the right panel of Fig. 2.7. First we observe that the best performance can be obtained using either fully connected affinity matrix either by severe sparsification (up to $k = 10$). The drop of performance induced by removal of lower similarity links can be explained by the fact that largest similarity values does not strictly follow real image similarities. The "valley" of lower classification indicates that relevant links in the graph had been removed, which is due to the difficulty to select a priori affinities solely on their raw pairwise values (see left panel of the Fig. 2.7 illustrating overlap of link weights linking same and different class nodes). The best performance is obtained due to most extreme graph sparcification. Such observation may be supported by the fact that absolutely largest affinity values (close to 1) are representative (e.g. two temporally neighbor image signatures) and meaningful. From histogram we see that lowest affinity value ever found in the kernel matrix is around 0.5 while during sparcification we threshold lower affinities to 0. This may create dis-balance in the graph and thus impair the final performance.

2.7 Conclusion

In this chapter we established a baseline for image-based localization using state-of-the-art visual features, dimensionality reduction techniques and SVM classification. We introduced and discussed several methods such as linear and non-linear approaches and their utility for visual content representation in lower dimensionality space. A graph perspective to represent the data both in fully connected and in sparse graphs was reviewed and evaluated. The non-linear kernels were also considered for comparing descriptors such as BOVW, CRFH and SPH in order to evaluate the similarities between the images. In the following studies we consider dimensionality reduction step as a preprocessing aimed to render high-dimensional descriptions more compact and taking into account the non-linear nature of the data itself by the use of appropriate kernel.

The image-based localization results presented in this chapter can be considered as baseline and shall be compared to when introducing and discussing more advanced methods.

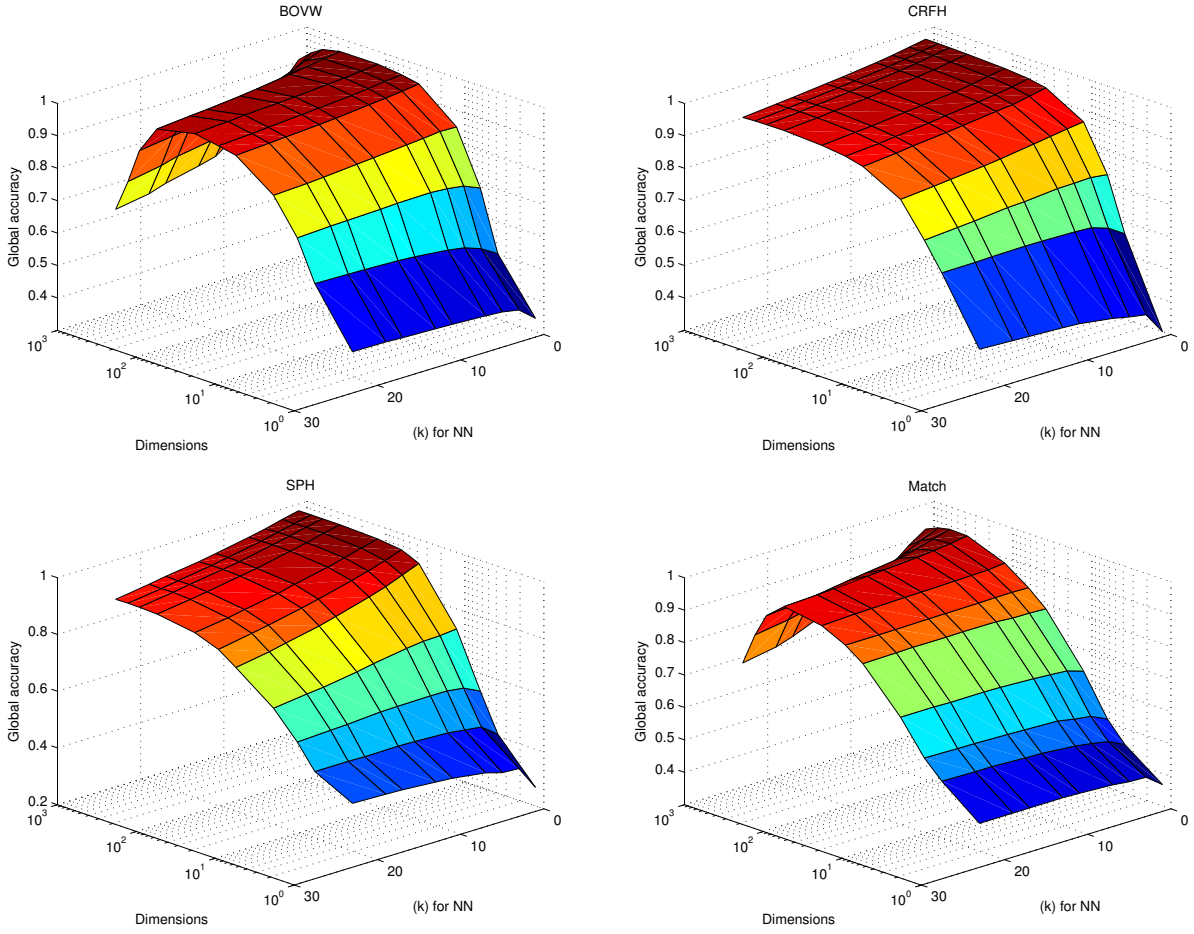


Figure 2.4: KPCA + NN : Changing the number of dimensions and number of neighbors for k-NN classifier (top: BOVW and CRFH; bottom: Pyramid Histograms and Match kernel)

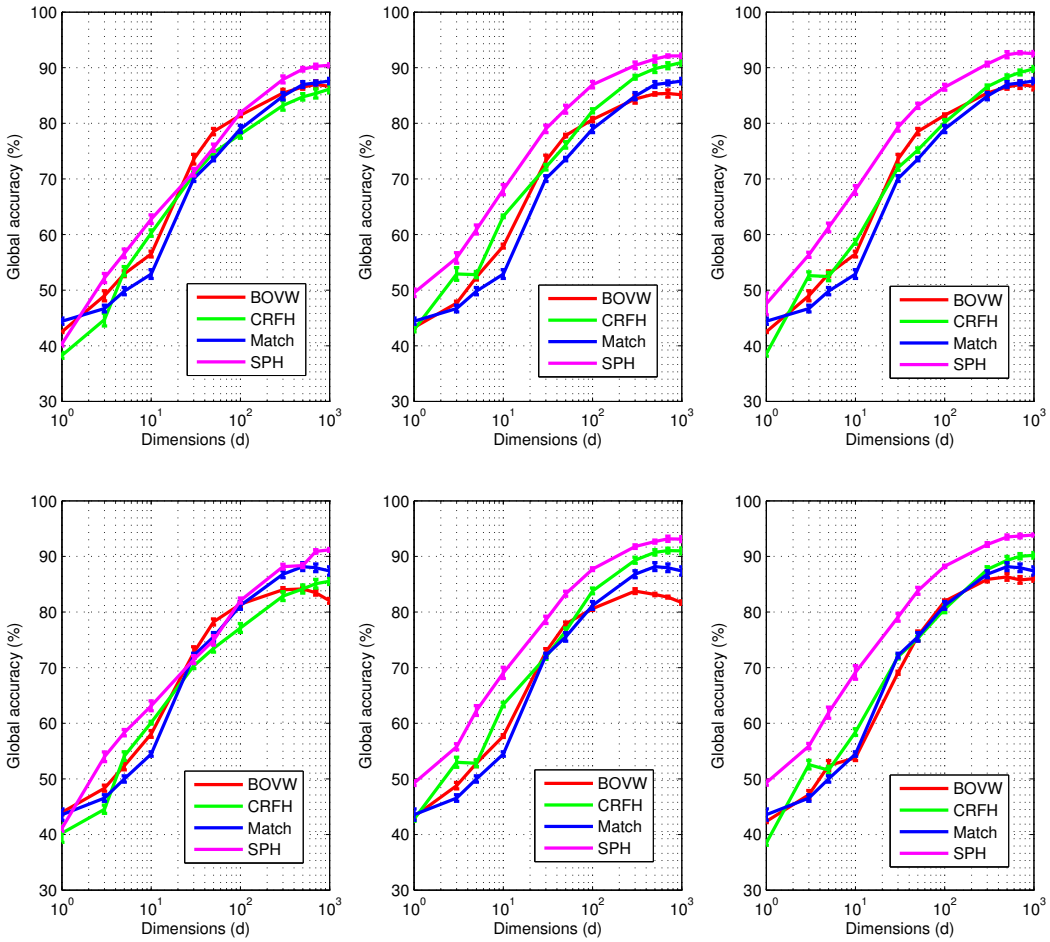


Figure 2.5: KPCA + SVM (top) and LapEig + SVM (bottom) : Effect of the number of dimensions on the performance (50% labeled data-set) for (left) linear, (middle) χ^2 and (right) intersection kernels respectively. Best viewed in color.

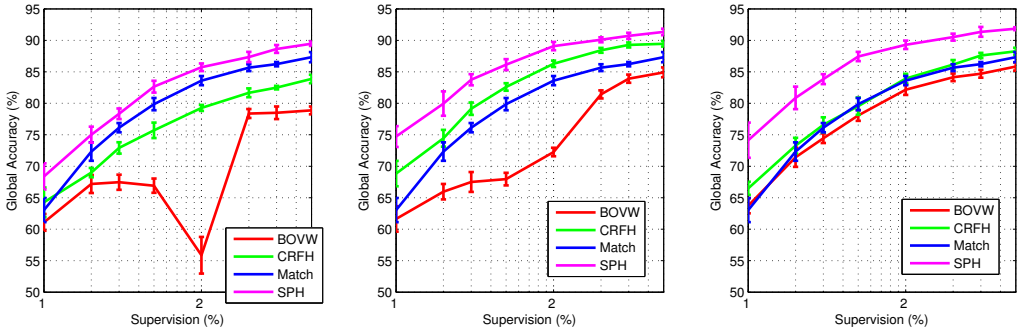


Figure 2.6: KPCA + SVM : Effect of the amount of training data on the performance for (left) linear, (middle) χ^2 and (right) intersection kernels respectively. Best viewed in color.

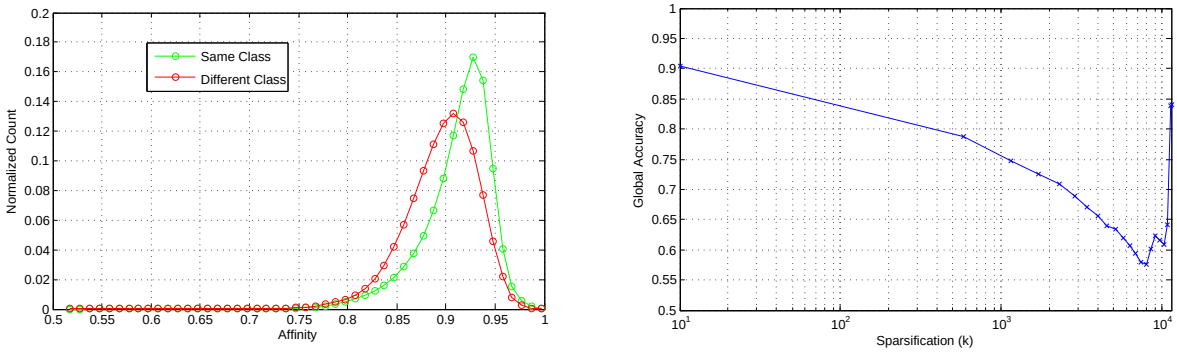


Figure 2.7: LapEig + SVM: (left) Distributions of the pairwise affinities; (right) Effect of the sparsification parameter k for k-NN pruning on the performance. The CRFH visual feature embeddings using χ^2 kernel have been used.

Chapter 3

Multiple Information Source Fusion

Contents

3.1 Introduction **40**
3.2 Early Fusion at Feature Level **41**
3.3 Late Fusion at Classification Level **44**
3.4 Experiments **50**

3.1 Introduction

3.1.1 Motivation

In this chapter we investigate utility of multiple visual cues for the task of image-based localization.

Due to multiple reasons, in real world applications a single feature is often insufficient for the task of classification. Different visual features capture different aspects of the scene and their choice is not always optimal for the task to solve. It is often unclear what visual features are most useful to extract the meaningful information for classification. As pointed out in a study [168] comparing several visual features, there is no optimal choice for all situations. Even humans perform poorly if using only one information source of perception [23]. To this end, instead of designing a specific and adapted descriptor for the task at hand, several existing state-of-the-art visual descriptors could be combined thus yielding increased discrimination power. For instance, one descriptor captures local salient features on a gray image, another may capture color histograms for each hue channel. A scene recognition algorithm would perform more likely better when using information captured by both such descriptors.

Kernel methods for supervised and semi-supervised classification are powerful method when dealing with real-world data. As confirmed by numerous practical applications in bioinformatics, computer vision and others, kernel methods work well if an appropriate kernel is used. Basically, the choice of kernel is data dependent. In practice, it is rarely known which kernel function should be used as well as how to choose the kernel parameters (if any). Together with the aforementioned weakness of single feature approach, the need for more robust learning method, while staying in the same framework, is evident. Therefore, we are left with a problem how to learn a kernel which is well suited for the task to solve.

3.1.2 Outline

This chapter is organized in four major sections. The first section 3.1 introduces to the subject and gives the motivation and brief review of the existing work. In section 3.2 and 3.3 two major strategies and their techniques for information fusion are presented. The former discusses the possibility of fusion at the feature level whereas the latter postpones the fusion to the late stage of classification. The chapter concludes with the section 3.4 devoted to the experimental part.

3.1.3 Strategies

There exist two major strategies on how to learn from multiple cue data: early and late fusion strategies.

Early Fusion strategies focus on the combination of input features before using them in a classifier. In the case of kernelized classifiers, the features can be seen as defining a new kernel that takes into account several features at once, thus defining a new RKHS fused space. Such methods receive the generic name Multiple Kernel Learning (MKL) where the objective is to estimate the optimal parameters for kernel combination. Earliest works of [35] and [36] on MKL focused on optimization of some loss function like kernel target alignment.

Classification problem was addressed in influential work of [74] showing that a new kernel can be learned using a linear combination of the base kernels. The problem was formulated as a Semi-Definite Program (SDP) or Quadratic Program with Quadratic Constraints (QCQP) if kernel weights are non-negative. Due to complexity, such approach does not scale well for larger data sets. Issue of scalability was addressed in the work of [7] by the new formulation of the problem in which an

efficient Sequential Minimal Optimization (SMO) algorithm [105] was used allowing to work with large-scale problems.

More efficiency was achieved in [138] by showing that the Semi-Infinite Linear Program (SILP) can be efficiently solved using off-shelf standard SVM solvers thus allowing to train on tens of thousand examples and 20 kernels in reasonable time. Gradient descent optimization was used in [118] with SimpleMKL and in a study of descriptor discriminability power and invariance in [158].

Utility of the norm inducing sparsity directly in the feature space was studied in [6]. Latest advances show more generality learning from large kernel spaces by using non-sparse regularization [157]. Finally, authors in [160] showed that it is possible to reuse standard SMO optimization algorithm for efficient optimization of ℓ^p MKL formulation.

Sparse ℓ^1 and non-sparse ℓ^p multiple kernel learning methods were compared and studied in the context of object classification [96]. An interesting result shows that an optimal p is dependent on the data and yields the best performance only if correctly tuned. The large values of p work best in the case of kernels bringing similar amounts of independent information. In contrast, smaller values or even $p \to 1$ works best for redundant or similarly informative kernels.

Late fusion strategy consists in training one or several base classifiers and feed their outputs to a second and decision maker layer.

In literature the approaches where the outputs of classifiers are fed into the next layer of classifiers, are called stacking methods. Refer to comprehensive and in-depth discussion on Multiple Classifier Systems in [2, 69, 73, 150, 97].

Such strategy can employ SVM as a base classifier. Following the work of [98], it was shown that SVM outputs, in the form of decision values, can be combined linearly using Discriminative Accumulation Scheme (DAS) [112] with the use of confidence measure. The following work evolved by relaxing the constraint of linearity of combination using a kernel function on the outputs of individual single feature outputs giving rise to Generalized DAS [116]. Other examples follow a similar path by respectively using combination rules (max, product etc) [85] and a comprehensive comparison of different fusion methods in [54] in the context of object classification.

3.1.4 Application to visual data

MKL was used for object detection within the challenge PASCAL VOC 2009 in the work of [159]. Very good performance was obtained on Caltech 101 database with the use of group sensitive multiple kernel learning [171] for object recognition. A computationally attractive for feature fusion [85] leverages individually trained SVM classifier outputs and applies four different strategies for result fusion. Besides of comparison of different visual descriptors (SIFT, CENTRIST, CT, SSIM, PATCH and OG) within Bag-of-Features framework and two popular kernels (intersection and χ^2) for image classification, authors show comparable and superior performances on databases Scene-{8, 13, 15} [77] and Caltech-{6, 101} [48].

Late fusion strategy for robot localization was used in [113, 112, 116].

3.2 Early Fusion at Feature Level

In this section we review the multiple kernel learning method [118] as a representative tool for multiple visual cue exploitation. The review of the method and experimental part show the place and fitness of multiple kernel learning for visual place recognition task.

3.2.1 Kernel Combination Rules

In Chapter 2 we reviewed the necessary elements of kernel learning methods for dimensionality reduction and classification. The central part in kernel methods is devoted to the proper selection of a kernel function $k(\mathbf{x}, \mathbf{x}')$ which induces the corresponding feature space $\mathcal{H}$ and is crucial for successful learning of the problem.

Suppose that a training set is given $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ composed of d-dimensional patterns $\mathbf{x}_i$ and the corresponding label y_i. Then by an explicit usage of proper user selected kernel function $k(\mathbf{x}, \mathbf{x}')$, a positive definite Gram matrix $(K)_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ is constructed. Recall the result of the Representer Theorem that a decision function $f \in \mathcal{H}$ can expressed in form

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \tag{3.1}$$

where the vector $\boldsymbol{\alpha} \in \mathcal{R}^n$ and the bias term $b \in \mathcal{R}$ are learned by an SVM solver. The task is to build the Gram matrix K and learn the parameters $\boldsymbol{\alpha}, b$.

Suppose a sum or averaging kernel. This kernel combination is one of most widely used in MKL due to its natural interpretation and conformity of the combined kernel with Mercer conditions. Intuition is to assign each kernel Gram matrix K_k a weight $\boldsymbol{\beta}_k > 0$ which is larger if the respective kernel is useful for supervised classification task. Learning the SVM model parameters $\boldsymbol{\alpha}, b$ together with kernel weights $\boldsymbol{\beta}_i$ is known as Multiple Kernel Learning.

A new Gram matrix K can be constructed from m multiple base Gram matrices K_k using a summing rule

$$(K)_{ij} = \sum_{k=1}^{m} \boldsymbol{\beta}_k (K_k)_{ij} \tag{3.2}$$

where $\sum_{i=1}^m \boldsymbol{\beta}_i = 1$ and $\boldsymbol{\beta}_i \geq 0$. It can be shown [125] that a weighted sum of Mercer kernels is also a valid Mercer kernel.

An useful insight can be gained if we examine the underlying functional framework. Let m be a number of weighted positive definite kernels $k_1(\mathbf{x}, \mathbf{x}'), \ldots, k_m(\mathbf{x}, \mathbf{x}')$ with their respective weight $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_m$. Each kernel function $k_k(\mathbf{x}, \mathbf{x}')$ induces an associated Hilbert space $\mathcal{H}_k$ whereas the weighted kernel function $\boldsymbol{\beta}_k k_k(\mathbf{x}, \mathbf{x}')$ respectively a Hilbert space $\tilde{\mathcal{H}}_k \subset \mathcal{H}_k$. Let $f \in \tilde{\mathcal{H}}_k$, the respective RKHS can be simply written using linearity property of dot product

$$f(\mathbf{x}) = \frac{1}{\boldsymbol{\beta}_k} \langle f(\cdot), \boldsymbol{\beta}_k k_k(\mathbf{x}, \cdot) \rangle_{\mathcal{H}_k} = \langle f(\cdot), \boldsymbol{\beta}_k k_k(\mathbf{x}, \cdot) \rangle_{\tilde{\mathcal{H}}_k} \tag{3.3}$$

such that the Hilbert space $\tilde{\mathcal{H}}_k$ contains the functions

$$\tilde{\mathcal{H}}_k = \left\{ f | f \in \mathcal{H}_k : \frac{\|f\|_{\mathcal{H}_k}}{\boldsymbol{\beta}_k} < \infty \right\} \tag{3.4}$$

endowed with a dot product

$$\langle f, g \rangle_{\tilde{\mathcal{H}}_k} = \frac{\langle f, g \rangle_{\mathcal{H}_k}}{\boldsymbol{\beta}_k} \tag{3.5}$$

Hence we see that the power of the MKL lies in its adaptability of the dot product metric for the classification task. A small positive kernel weight will render the functions $f, g \in \tilde{\mathcal{H}}_k$ dissimilar which is due to this modification of the scalar product.

Finally, the individual Hilbert spaces $\tilde{\mathcal{H}}_k$ can be combined into a Hilbert space $\mathcal{H}$ using orthogonal or direct sum [47] operation

$$\mathcal{H} = \oplus_{k=1}^{m} \tilde{\mathcal{H}}_k \tag{3.6}$$

where standard result on RKHS states that there exists a function $k(\mathbf{x}, \cdot) \in \mathcal{H}$ such that the corresponding Gram matrix of Eq. 3.2.

Finding the the weights to improve the discrimination between the classes can be done using supervised MKL method, which is reviewed in detail in Annex B.

3.2.2 Feature space of the weighted kernel sum

Weighted linear sum of kernels has an interesting interpretation in functional framework. It can be shown that a weighted linear sum of Mercer kernels corresponds to an augmented feature space where individual mapping vectors are concatenated.

Sum kernel for MKL Suppose that $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$ and m kernel functions $k_1(\cdot,\cdot), \ldots, k_m(\cdot,\cdot)$ are given. Each kernel function k_k induces a corresponding RKHS space $\mathcal{H}_k$ such that

$$k_k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi_k(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle_{\mathcal{H}_k} \tag{3.7}$$

Given kernel weights vector $\boldsymbol{\beta}$, a new kernel function can created

$$k_{\mathrm{MKL}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{m} \boldsymbol{\beta}_k k_k(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{m} \boldsymbol{\beta}_k \langle \Phi_k(\mathbf{x}_i), \Phi_k(\mathbf{x}_j) \rangle \tag{3.8}$$

where $\boldsymbol{\beta}_k \geq 0$.

Functional framework for sum kernel We can show that a mapping function Φ_{MKL} composed of m weighted mapping functions $\Phi_i, i = 1, \ldots, m$

$$\Phi_{\mathrm{MKL}}(\mathbf{x}) = \begin{pmatrix} \sqrt{\boldsymbol{\beta}_1}\Phi_1(\mathbf{x}) \\ \sqrt{\boldsymbol{\beta}_2}\Phi_2(\mathbf{x}) \\ \vdots \\ \sqrt{\boldsymbol{\beta}_m}\Phi_m(\mathbf{x}) \end{pmatrix} \tag{3.9}$$

corresponds to a weighted sum of kernel functions $k_1(\cdot,\cdot), \ldots, k_m(\cdot,\cdot)$. The new kernel function can be written such that

$$\begin{aligned}
k_{\mathrm{MKL}}(\mathbf{x}_i, \mathbf{x}_j) &= \langle \Phi_{\mathrm{MKL}}(\mathbf{x}_i), \Phi_{\mathrm{MKL}}(\mathbf{x}_j) \rangle = & (3.10)\\
&= \left\langle \begin{pmatrix} \sqrt{\boldsymbol{\beta}_1}\Phi_1(\mathbf{x}_i) \\ \sqrt{\boldsymbol{\beta}_2}\Phi_2(\mathbf{x}_i) \\ \vdots \\ \sqrt{\boldsymbol{\beta}_m}\Phi_m(\mathbf{x}_i) \end{pmatrix}, \begin{pmatrix} \sqrt{\boldsymbol{\beta}_1}\Phi_1(\mathbf{x}_j) \\ \sqrt{\boldsymbol{\beta}_2}\Phi_2(\mathbf{x}_j) \\ \vdots \\ \sqrt{\boldsymbol{\beta}_m}\Phi_m(\mathbf{x}_j) \end{pmatrix} \right\rangle = & (3.11)\\
&= \sum_{k=1}^{m} \left\langle \sqrt{\boldsymbol{\beta}_k}\Phi_k(\mathbf{x}_i), \sqrt{\boldsymbol{\beta}_k}\Phi_k(\mathbf{x}_j) \right\rangle_{\mathcal{H}_k} = & (3.12)\\
&= \sum_{k=1}^{m} \boldsymbol{\beta}_k k_k(\mathbf{x}_i, \mathbf{x}_j) & (3.13)
\end{aligned}$$

where we used Eq. 3.9 and linearity property of dot product. We see that a weight $\boldsymbol{\beta}_k$ for a kernel function k_k has an effect of scaling by a factor of $\sqrt{\boldsymbol{\beta}_k}$ of the kernel mapping function $\Phi_k(\mathbf{x})$.

Practical considerations The previous property has a practical utility. Suppose that multiple Gram kernel matrices are given and the goal is to compute low dimensional representations of the corresponding graph. Instead of computing a new weighted kernel for every new $\boldsymbol{\beta}$ and launching Kernel PCA on it, following steps can be run :

1. Compute low-dimensional embeddings Z_k for each Gram matrix K_k separately;
2. Weight each embeddings vector $\mathbf{z}_i^k \in Z_k$ with $\sqrt{\boldsymbol{\beta}_k}$

$$\tilde{\mathbf{z}}_i^k = \sqrt{\boldsymbol{\beta}_k}\mathbf{z}_i^k \tag{3.14}$$

3. Construct final embeddings vector as a concatenation

$$\mathbf{z}_i^{\mathrm{MKL}} = \begin{pmatrix} \tilde{\mathbf{z}}_i^1 \\ \tilde{\mathbf{z}}_i^2 \\ \vdots \\ \tilde{\mathbf{z}}_i^m \end{pmatrix} \tag{3.15}$$

The procedure allows efficient computation of low-dimensional graph embeddings in the case of Kernel PCA acting on graph represented by a kernel matrix. Although, there may be a risk of overfitting because of concatenation of embeddings.

3.3 Late Fusion at Classification Level

Similarly to the previous statement that there is no single best performing feature for all recognition task, there is also no reason to believe that one particular classification method outperforms the others. Indeed, as the familiarly called "No Free Launch Theorem" [46] states, "if there is no prior knowledge about the nature of the classification problem, the best classification method will be dependent on the data".

Leaving the vast subject of algorithm-independent machine learning [46] to the interested reader, in this section we are interested in high level classifier combination. According to [69], there are two major scenarios for classifier combination: multiple classifiers trained on the same data representation and multiple (not necessarily the same methods) classifiers trained on different representations of the same object or phenomena. As in [69], we shall discuss the methods and present the experimental results using the second scenario if not stated otherwise.

3.3.1 Brief Categorization of Fusion Architectures

According to [73], there are four general fusion architectures or topologies : parallel, serial, hierarchical and hybrid. In pattern classification literature [73] these are called classifier ensembles which include type and number of base classifier in the ensemble. Refer to the comprehensive study [97] for more details on so-called topic Multiple Classifier Systems.

In a parallel architecture multiple base classifiers operate independently. Outputs of the base classifiers are merged with the help of a fuser function providing the final result. Base classifiers should not necessarily be of the same type. This architecture was used for SVM output fusion into a single result using an accumulation scheme in [98, 116, 113] for robot localization task indoors.

A fuser function can be further categorized in two categories:

1. Integration
   Each base classifier contributes for a final decision. This implies competition of base classifiers.

2. Selection
   The final decision for each pattern depends on one or a selection of base classifiers. This implies complementarity of base classifiers.

Serial architectures In a serial architecture a succession of base classifiers are used with each classifier providing a reduced set of possible classes. Often used for data with many classes such as character recognition [68] and in biometrics [150] where parallel architecture proved to be too costly to employ. In [94] authors used sequential architecture for robot topological place recognition by building probability like decision histograms from the outputs of base classifiers.

Hierarchical architectures In a hierarchical architecture two or more layers of base classifiers feed their outputs to the next layer of classifiers. This architecture is often application specific and has been used for hyperspectral data classification [72] and high dimensional data classification [72, 2].

Hybrid architectures Hybrid architectures represent base classifier combinations that do not belong fully to any of the mentioned categories. Hybrid architectures are investigated and used less often than serial and hierarchical architectures.

The choice of the architecture In our context, the number of places is limited (less than 10) and we favorize a simple architecture for easy inclusion of multiple visual features.

3.3.2 Notion

Suppose that we are given a set of n patterns with their respective labels $\left\{\left(\mathbf{x}_i^t, y_i\right)\right\}_{i=1}^n$ and with descriptor types $t = 1, \ldots, T$. Therefore, each image $\{I_i\}_{i=1}^n$ belongs to one of disjoint classes $\{\omega_j\}_{j=1}^m$ and is represented simultaneously by T different descriptor vectors. In Fig. 3.1 we depict multi-class SVM scores computed for a set of images.

Consider also a trained classifier $h^t\left(\mathbf{x}^t, \boldsymbol{\theta}^t\right)$ which accepts type t patterns $\mathbf{x}^t \in \mathcal{R}^{d_t}$, a parameter vector $\boldsymbol{\theta}^t$ and returns a vector of real-valued outputs $\boldsymbol{s}^t = \left[s_1^t, \ldots, s_p^t\right]$. Finally, output of some rule

$$\hat{y}_i^t = P\left(\boldsymbol{s}_i^t\right) \tag{3.16}$$

is used to obtain class estimation $\hat{y}_i^t$ for an image I_i.

In case of an SVM classifier, the parameter vector $\boldsymbol{\theta}$ is called a model and an output vector $\boldsymbol{s}$ contains the scores. For one-vs-all setup the dimensionality of scores vector $\boldsymbol{s}$ is $p = m$ while for one-vs-one it equals to $p = \frac{m(m-1)}{2}$. The standard one-vs-all rule works as follows

$$\hat{y}_i^t = \arg\max_j \boldsymbol{s}^t \tag{3.17}$$

where index j runs through all scores vector elements s_j^t.

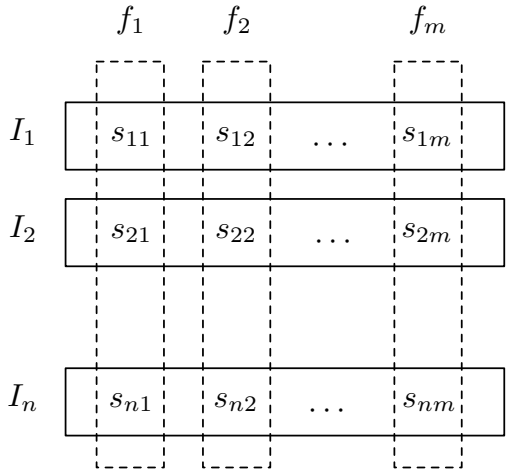


Figure 3.1: SVM scores returned for a set of images $(I_1, \ldots, I_n)$. There are m topological locations and in one-vs-all setup m real-valued scores are obtained for each image.

3.3.3 Baseline : Majority Voting

Majority voting [46, 73] is one of simplest rules for multiple classifier output fusion. It assumes that each base classifiers provide their estimations and that the majority is right.

Suppose that an test image I_{new} gets computed descriptor vectors $\left\{\mathbf{x}_{\text{new}}^t\right\}_{t=1}^T$. Then using some training set $\left\{X^t\right\}_{t=1}^T$, a number of base classifiers $\left\{h^t\left(\cdot, \boldsymbol{\theta}^t\right)\right\}_{t=1}^T$ was constructed. Each of these base classifiers provides an output vector

$$\boldsymbol{s}_{\text{new}}^t = h^t\left(\mathbf{x}_{\text{new}}^t, \boldsymbol{\theta}^t\right) \tag{3.18}$$

where application of rule P would produce class estimations

$$\hat{y}_{\text{new}}^t = P\left(\boldsymbol{s}_{\text{new}}^t\right) \tag{3.19}$$

Majority voting rule simply counts the number of votes for each class $\omega_j, j = 1, \ldots, m$ and returns the final estimation for which there were most votes. Ties are usually broken at random.

In practice, majority voting performs well if each classifier is an expert on some part of input space. However, this method usually performs poorly when some classifiers are either very good or very bad at classification [97].

3.3.4 Discriminative Accumulation Scheme

A discriminative accumulation scheme (DAS) was initially proposed in [98] and then generalized to confidence measure exploitation [112] to resolve ambiguous cases for application of robot localization indoors. This scheme belongs to an architecture of parallel base classifier combination followed by a fuser. Advantage of this scheme is two folds [115]:

1. Less sensitivity to misleading individual base classifier estimates
   If one of the base classifiers in majority voting scheme provide incorrect estimates, the final class estimation is at risk.

2. Lessened risk of curse of dimensionality
   One of the straightforward approaches for early multiple cue fusion is to concatenate respective

descriptor vectors which is reminiscent of what MKL is doing in the RKHS. Unfortunately, this implies a risk of overfitting. The effect is more pronounced for the small training sets with large dimensionality of description space.

Authors of DAS in [98] used the SVMs as base classifiers. Linearly weighted sum of corresponding scores produced a final set of scores from which class estimation took place and which we review here.

Linear model for SVM score combination Recall that a binary SVM classifier provides real-valued outputs (See Chapter 2 for details)

$$f\left(\mathbf{x}_{\text{new}}\right)=\sum_{i=1}^{n}\alpha_i y_i k\left(\mathbf{x}_i,\mathbf{x}_{\text{new}}\right)+b \tag{3.20}$$

For data with classes $\omega_1,\ldots,\omega_m$ and One-vs-All setup, m decision functions f_j will be trained. For a pattern $\mathbf{x}_{\text{new}}$ evaluation of j decision functions produces an output that can be summarized in a vector

$$\boldsymbol{s}_{\text{new}}=\left[f_1\left(\mathbf{x}_{\text{new}}\right),\ldots,f_m\left(\mathbf{x}_{\text{new}}\right)\right] \tag{3.21}$$

The standard one-vs-all class estimation scheme reduces to find the largest decision value

$$\hat{j}=\arg\max_j\left\{f_j\left(\mathbf{x}\right)\right\}$$

Suppose now that T SVM classifiers were trained and produced T m-dimensional vectors in output for the test pattern

$$\boldsymbol{s}_{\text{new}}^t=\left[f_1^t\left(\mathbf{x}_{\text{new}}\right),\ldots,f_m^t\left(\mathbf{x}_{\text{new}}\right)\right] \tag{3.22}$$

At this point cue estimations are independent and there is a need to produce a merged result. DAS creates a linear weighted sum of binary decision functions across multiple cues. That is, for a fixed decision function f_j across the cues $\left\{f_j^t\left(\cdot\right)\right\}_{t=1}^{T}$, a combined output writes as a weighted linear sum

$$f_j^{\text{DAS}}\left(\mathbf{x}_{\text{new}}\right)=\sum_{t=1}^{T}\boldsymbol{\beta}_t f_j^t\left(\mathbf{x}_{\text{new}}\right) \tag{3.23}$$

Repeating the same weighted summation procedure for all binary classifiers, the DAS output is of same format as a single cue SVM output

$$\boldsymbol{s}_{\text{new}}^{\text{DAS}}=\left[f_1^{\text{DAS}}\left(\mathbf{x}_{\text{new}}\right),\ldots,f_m^{\text{DAS}}\left(\mathbf{x}_{\text{new}}\right)\right] \tag{3.24}$$

from which the estimated class can be computed

$$\hat{j}^{\text{DAS}}=\arg\max_j\left\{f_j^{\text{DAS}}\left(\mathbf{x}_{\text{new}}\right)\right\} \tag{3.25}$$

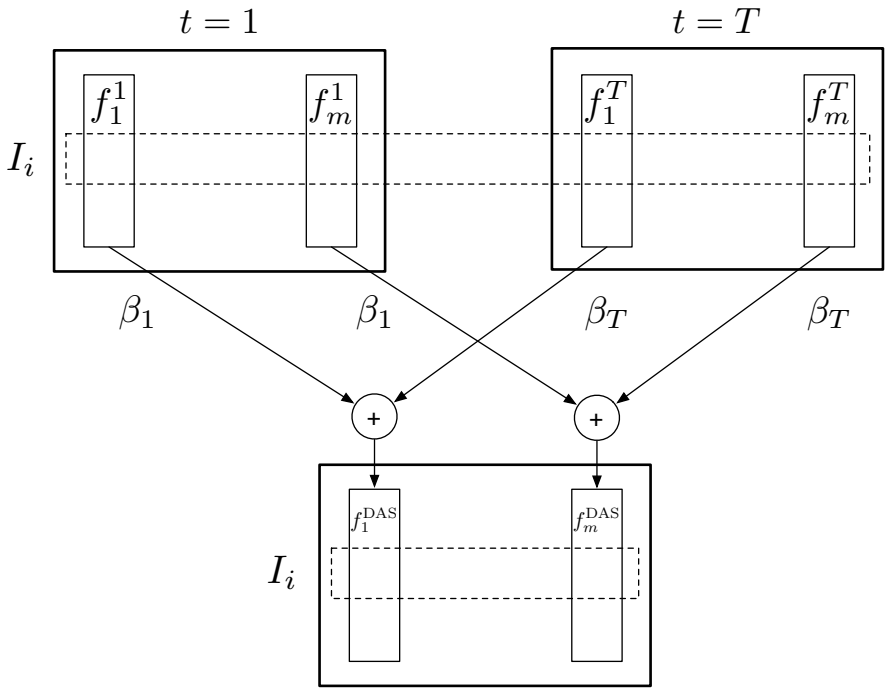


Figure 3.2: DAS scores combined from base classifiers over T visual cues

Graphical view of linear SVM score combination For illustration purposes we can graphically represent in Fig. 3.2 the DAS method for cue fusion as a two layer architecture.

Indeed, the first layer is composed of classifiers $\left\{h^t\left(\cdot, \boldsymbol{\theta}^t\right)\right\}_{t=1}^T$ which provide the outputs in form of scores $\boldsymbol{s}_{\text{new}}^t = h^t\left(\mathbf{x}_{\text{new}}^t, \boldsymbol{\theta}^t\right)$ for every test pattern. The second layer consists of a single fuser function which linearly combines all the scores in the final output. It is clear from the diagram that functions $f_i^t, i = 1, \ldots, m$ outputs for each cue type t are combined independently with respect to other functions. This architecture can resemble a simplified neural network [46] with two layers using independent and linear feature combination.

We will compare this architecture with a more general scheme which we review in the next subsection.

3.3.5 Support Vector Machine DAS (SVM-DAS)

DAS procedure effectively merges the individual base classifier outputs and experimentally showed [98, 112] to give the results of at least the best base classifier performance. However, this is true only in the case of correct selection of weights for each base classifier. While providing improvement, the method has two weaknesses:

1. Linear combination
   Potentially limiting factor is the model for base classifier fusion using a linear combination. While simple and intuitive, this may not be optimal rule for result fusion for real-world problems.

2. Optimizing the weights
   The weights used in linear combination should be optimized in order to get the best results. In all references [98, 112], the weights were found using a cross-validation procedure. While

being increasing costly for more cues, it also requires to have a separate validation set that may be wasteful of valuable training data.

Generalized model for SVM score combination SVM-DAS [116, 115] addressed these issues by using a non-linear kernel on individual base classifier outputs within a SVM framework. Suppose that the T base classifiers produced outputs as in Eq. 3.22. For a particular type of feature t, the m-dimensional vector $\boldsymbol{s}_{\text{new}}^t$ can be regarded as a part of a new pattern $\boldsymbol{z}_{\text{new}}$ of dimension $T \times m$

$$\boldsymbol{z}_{\text{new}} = \left[\boldsymbol{s}_{\text{new}}^1, \boldsymbol{s}_{\text{new}}^2, \ldots, \boldsymbol{s}_{\text{new}}^T\right] \tag{3.26}$$

which is obtained by the concatenation of scores. At this point, a new classifier can be built for each binary classification problem

$$f_j^{\text{SVMDAS}}\left(\boldsymbol{z}_{\text{new}}\right) = \sum_{i=1}^{n} \alpha_{ij} y_i k\left(\boldsymbol{z}_i, \boldsymbol{z}_{\text{new}}\right) + b_j \tag{3.27}$$

and the class estimations can be computed from the new vector of scores

$$\boldsymbol{s}_{\text{new}}^{\text{SVMDAS}} = \left[f_1^{\text{SVMDAS}}\left(\boldsymbol{z}_{\text{new}}\right), \ldots, f_m^{\text{SVMDAS}}\left(\boldsymbol{z}_{\text{new}}\right)\right] \tag{3.28}$$

using the rule of largest score

$$\hat{j}^{\text{SVMDAS}} = \arg\max_j \left\{ f_j^{\text{SVMDAS}}\left(\boldsymbol{z}_{\text{new}}\right)\right\} \tag{3.29}$$

One should note that an appropriate kernel function should be selected.

Graphical view of generalized SVM score combination At closer inspection, the SVM-DAS scheme can be seen as generalization of DAS if a linear kernel is used. Recall that a SVM-DAS pattern $\boldsymbol{z}$ is constructed from concatenation of multiple SVM classifier outputs $\boldsymbol{s}^1, \ldots, \boldsymbol{s}^T$. It is straightforward to show that with linear kernel the dot product

$$k_{\text{SVMDAS}}\left(\boldsymbol{z}_i, \boldsymbol{z}_j\right) = \left\langle \boldsymbol{z}_i, \boldsymbol{z}_j \right\rangle = \sum_{t=1}^{T} \left\langle \boldsymbol{s}_i^t, \boldsymbol{s}_j^t \right\rangle \tag{3.30}$$

Substituting Eq. 3.30 into the expression of decision function Eq. 3.27 and exchanging the sums we get

$$\begin{aligned} f_j^{\text{SVMDAS}}\left(\boldsymbol{z}_{\text{new}}\right) &= \sum_{i=1}^{n} \alpha_{ij} y_i k_{\text{SVMDAS}}\left(\boldsymbol{z}_{\text{new}}, \boldsymbol{z}\right) + b & (3.31)\\ &= \sum_{i=1}^{n} \alpha_{ij} y_i \sum_{t=1}^{T} \left\langle \boldsymbol{s}_i^t, \boldsymbol{s}_j^t \right\rangle + b & (3.32)\\ &= \sum_{t=1}^{T} \left\langle \boldsymbol{s}_i^t, \boldsymbol{s}_j^t \right\rangle \sum_{i=1}^{n} \alpha_{ij} y_i + b & (3.33) \end{aligned}$$

Denote

$$\boldsymbol{w}_j^t = \sum_{i=1}^{n} \alpha_{ij} y_i \boldsymbol{s}^t \tag{3.34}$$

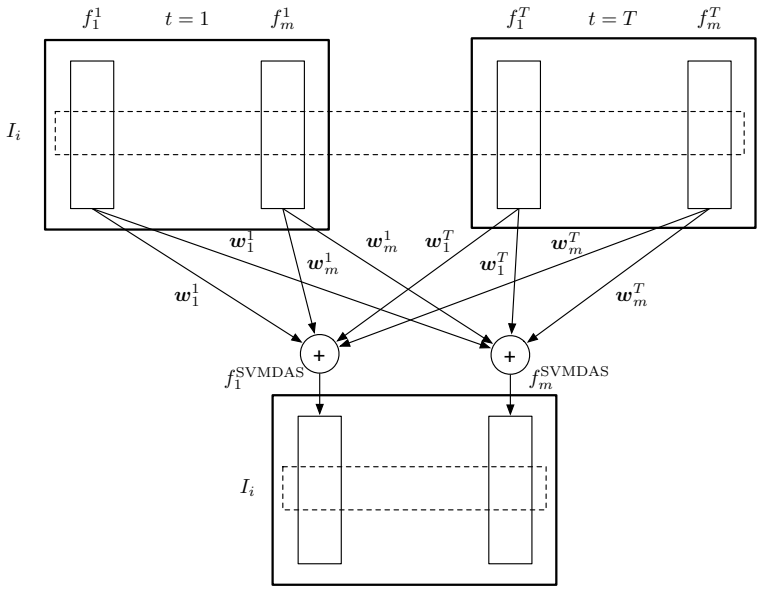


Figure 3.3: SVM-DAS scores combined from base classifiers over T visual cues and compared using linear kernel

where $\boldsymbol{w}_{jk}^t$ is kth element of the vector $\boldsymbol{w}_j^t$. Then the SVM-DAS decision function takes the form

$$f_j^{\text{SVMDAS}}\left(\boldsymbol{z}_{\text{new}}\right) = \sum_{t=1}^{T}\sum_{k=1}^{n} \boldsymbol{w}_{jk}^t f^t\left(\mathbf{x}_{\text{new}}\right) \tag{3.35}$$

where we have used the fact that $\boldsymbol{s}_{\text{new}}^t = f^t\left(\mathbf{x}_{\text{new}}\right)$ is simply a vector of SVM scores for a pattern $\mathbf{x}_{\text{new}}$. From the result in Eq. 3.35 is evident that all multiple cues as well as all other pattern scores are contributing for every new decision function value. Note the summing over all the cues and the summing over all the patterns. Graphically the effect of using a linear kernel on concatenated SVM scores over multiple cues is depicted in Fig. 3.3.

Comparing Fig. 3.2 for DAS and Fig. 3.3 for SVM-DAS we can notice the similarities in terms of score accumulation (summation rule) and difference in the links used to combine them. The DAS scheme can be easily seen as special case of SVM-DAS using the linear kernel. The different choices of kernel function may lead to various and probably non-linear relationships between the cues and binary classifiers.

3.4 Experiments

In this section we present the results of multiple cue fusion using the two presented strategies : early and late fusion. The former is the MKL method which learns a new kernel that is more adapted for a specific classification problem depending on the data. The latter is the SVM-DAS method that builds a final decision based on SVM decision on base classifier scores.

First we compare both strategies with respect to the baseline - standard single feature approaches. Secondly we compare the two strategies and point out differences and advantages.

3.4.1 Image Database and Experimental Setup

For the experimental part of this chapter we selected the IDOL2 image database [86].

Set / Split	Training (%)	Validation (%)	Testing (%)
1	1	49.5	49.5
2	2	49	49
3	3	48.5	48.5
4	5	47.5	47.5
5	10	45	45
6	20	40	40
7	30	35	35
8	50	25	25

Table 3.1: IDOL2 supervision levels

Visual Feature	Original Dim.	Kernel Fnc	Reduced Dim.	Ref.
Bag of Visual Words (BOVW)	1111	χ^2	2000	[100]
Comp. Rec. Field Hist. (CRFH)	439 191 718 (sparse)	χ^2	2000	[79]
Lowe Match (Match)	-	Lowe match	2000	[32]
SPH Level 3	4200	χ^2	2000	[77]

Table 3.2: Low level visual features

Test setup The corpus containing 12 video sequences for the "minnie" part of the database is considered as a global set of images discarding the information about sequences, lighting conditions and time span. This allows us to focus on the intrinsic properties of the features and their best combination.

To simulate different supervision levels, we sampled randomly 8 sets comprised of training, validation and testing as described in Table 3.1. To assess the stability of classification performance, 10 different folds were created for each set.

In all experiments we used the training set solely for the training of the model and the validation set for the estimation of the model parameters. The performance was assessed only on the testing set. The final classification performance was averaged over 10 folds within each set of supervision.

Choice of Visual Features Image contents were described using 4 visual features - Bag of Visual Words, Composite Receptive Field Histograms, Matching information and Spatial Pyramid Histograms at level 3.

Dimensionality of each feature was reduced to 2000 dimension using the Kernel PCA method, thus generalizing embedding considered to belong to an Euclidean space on which the linear kernel is applied. A summary of the feature properties, used kernels and other information is given in Table 3.2.

Note on the use of MKL SimpleMKL method is a supervised classification method that learns a new weighted kernel from k base kernels. Due to 8 supervision levels and 10 folds within each of them, computation of the best kernel weights and then computation of embeddings for the new weighted kernels was considered too high. To alleviate this issue, we launched the SimpleMKL method to find the best kernel weights on the validation set and created the embedding vectors for the tests by concatenation of weighted base kernel embeddings as described in subsection 3.2.2. Notice that concatenation yielded in increasingly higher dimensional embeddings.

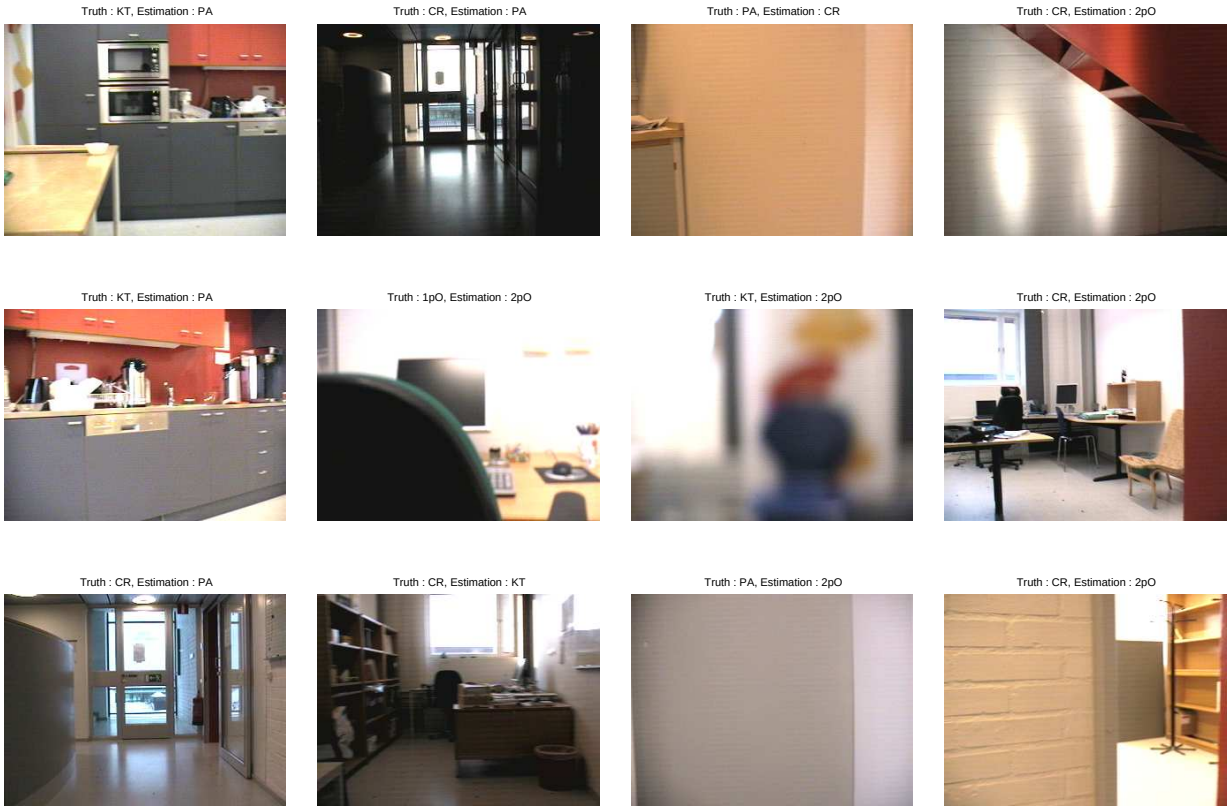


Figure 3.4: BOVW (top row), SimpleMKL (middle row) and SVM-DAS (bottom row) misclassification samples : (a) misclassification, (b) poor light, (c) non-informative, (d) label noise

The choice of the classifier In all experiments in the current chapter we used a soft margin SVM classifier (See Chapter 2) with linear kernel. The only free parameter C controlling the complexity of the learned model was estimated using the validation set.

3.4.2 Performance comparison of Early and Late fusion approaches

Single feature performance Using the established test setup with 8 supervision levels, we present an averaged global accuracy for each of the visual features in Fig. 3.5.

The results show a steady increase of classification performance for all the single visual features as more labeled samples are provided for training. Confusion matrices are depicted for the low supervision set in top line of Fig. 3.7.

We are interested in the region of low supervision rates, for example, that is, 1% and 10% of the current data set. Comparison of the performances provided by diverse methods in this region is interesting because real-world application is expected to work on sparsely annotated data. Note that different visual features exhibit different performance.

Early feature fusion using MKL In this experiment we are interested to assess the performance of the early feature fusion using SimpleMKL, where the kernel weights are learned by the algorithm, and the EvenMKL using same kernel weights.

We compare the two MKL approaches with respect to single feature baselines. Note an overall

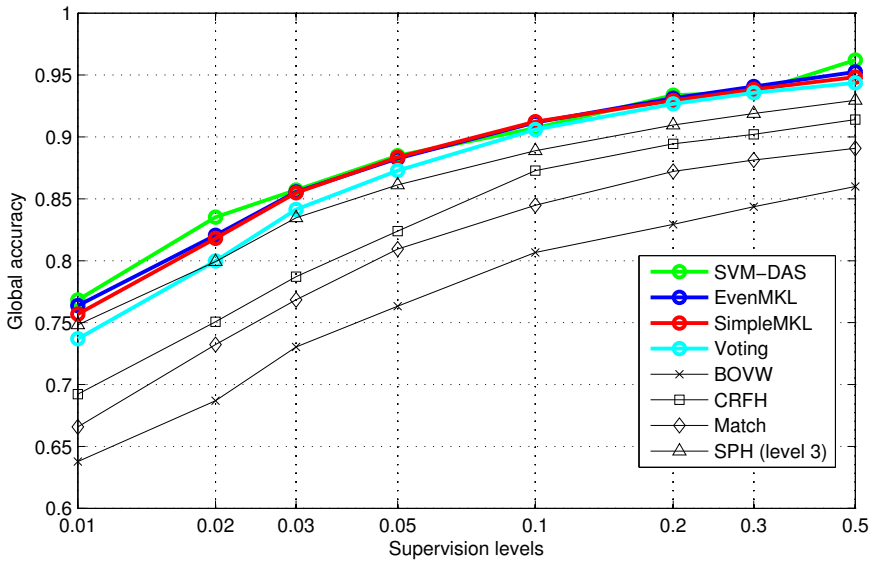


Figure 3.5: Average performance of single feature approaches and multiple information source fusion for varying supervision levels.

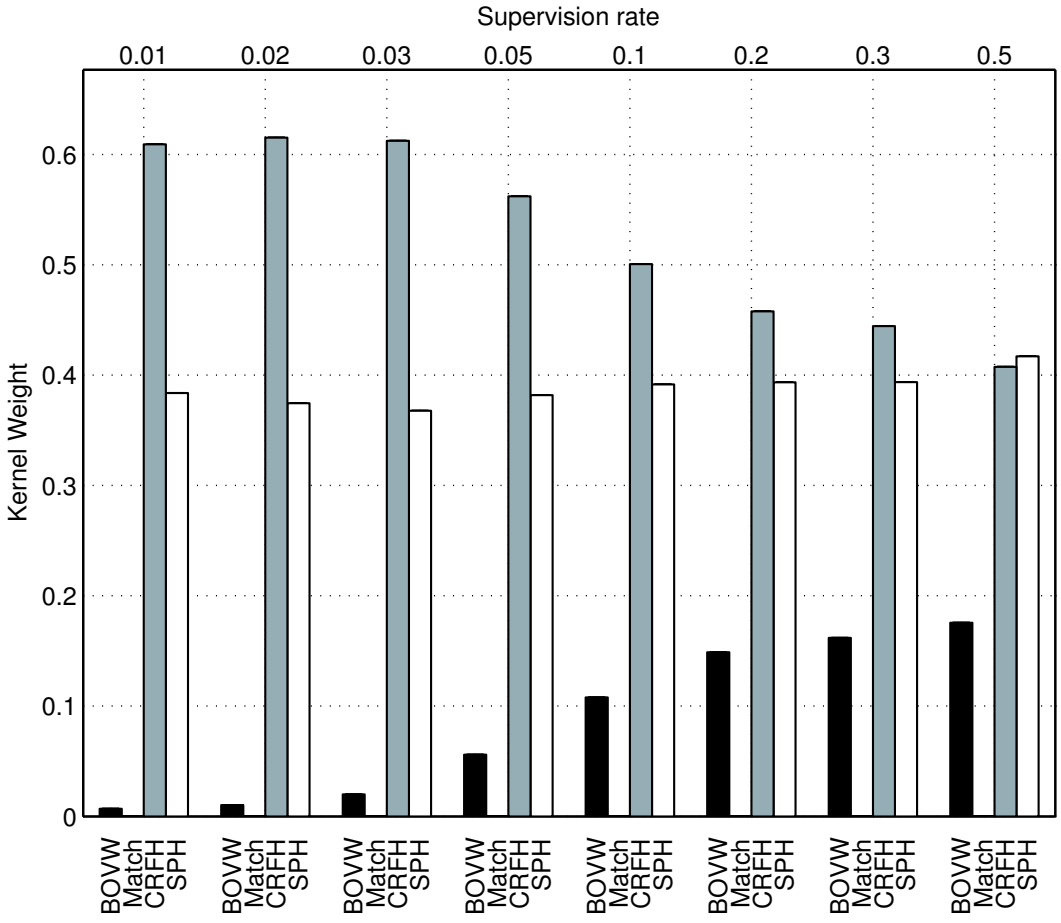


Figure 3.6: Multiple Kernel Learning : Kernel weights learned using SimpleMKL

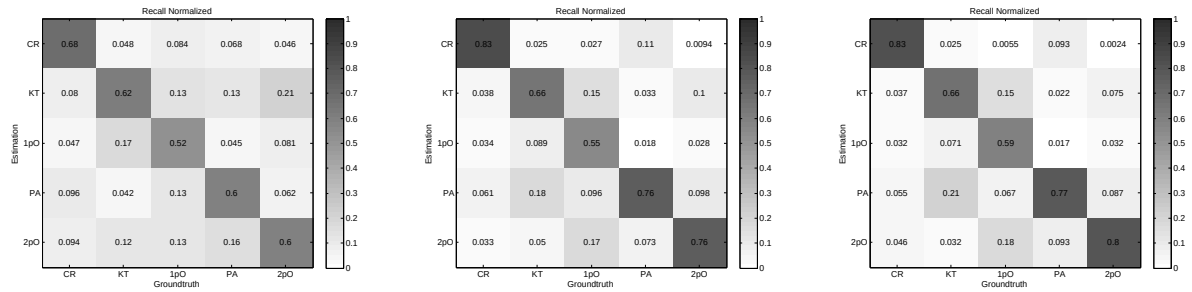


Figure 3.7: Confusion matrices (1% of supervision): (top) BOVW, (middle) SimpleMKL, (bottom) SVM-DAS

superiority of both MKL methods over all baselines. This is true at all levels of supervision. Precision and recall normalized confusion matrices, selected at lowest supervision rate, Fig. 3.7 middle line, shows that the classes like "Printer_Area", "Kitchen" and "One_Person_Office" are hardest to classify.

It is interesting to visualize the kernel selection weights $\boldsymbol{\beta}_k$. Since the MKL method is a supervised learning method and endorses sparse selection of kernels, it may be expected that most relevant kernels are selected for the classification task. In Fig. 3.6 kernel weights are displayed as they were found at each supervision level independently. We note that the Match kernel was never selected despite its relatively higher performance with respect features such as BOVW.

From the same figure we can observe different kernel weights found using the supervised approach at varying supervision levels. This may indicate the lack of available training data which in turn may hinder to learn a new combined kernel with improved discrimination power.

If we validate the results of MKL, we note that even weighted kernel combination performs as good as the more sophisticated method. Therefore, a simple feature concatenation proves to be beneficial for the classification task for this data.

Late feature fusion using SVM-DAS In the second part of experimentation we perform the comparison between late fusion approach and baselines. The results are presented in Figure 3.5.

Recall that late fusion approach named SVM-DAS effectively constructs new patterns from baseline classifier outputs as described in Subsection 3.3.5.

We note that classifier result fusion using SVM-DAS method is superior to all of the baselines. Its performance is comparable to that of SimpleMKL for this data set. Its confusion matrix for 1% supervision is depicted in Fig. 3.7.

In present experiments no sequence information nor time information was exploited. Due to random data sampling for testing purposes, these results can be seen as optimistic estimates when all visual information is used and the training data is sampled uniformly from all possible weather and time conditions.

Misclassification patterns Global classification accuracy is one of indicators of the classification performance. It may be interesting to analyze the individual image label estimates to understand the reasons of misclassification.

From Fig. 3.5 we note a good performance of all methods at 50% of labeling rate. It is expected that few errors will be made by any of the methods since the training and testing samples are interleaved. The reason to select this supervision rate (50% of training) and the test setup is to observe the principal difficulties that may arise in real life conditions. The problem of low supervision

is not covered here.

We selected three methods to analyze the estimates:

1. Baseline BOVW
2. Early fusion SimpleMKL
3. Late fusion SVM-DAS

The analysis of the classification estimates revealed several types of misclassification regardless the method used. We identified four groups:

1. Generic misclassification
   This group of misclassification appeared rarely and typically hints for insufficient supervision, class overlap or imperfections of the visual features used. Images in this group are expected to be classified correctly because of good image quality and characteristic contents for a particular place and class. This type of error can be fixed.

2. Poor light conditions
   Images in this group feature over or underexposed scenes, such that visual feature detectors fail to capture relevant discriminant information. For example, local visual features like SIFT [84] and SURF [10] exhibit poor performance when light conditions are bad. This type of error can be fixed by using features that are more robust to poor light conditions or take advantage of temporal consistency where inference from good quality temporal neighbors can be made.

3. Non-informative image contents
   This group of misclassifications concerns the images of generally good lighting but contain no characteristic information that would help to attribute it to any of classes. Such image examples are: blurred contents, close-up of some scene or object. This type of error could be fixed using temporal context or by detecting low confidence estimations for later exclusion or post-processing.

4. Label noise
   Due to arbitrary class label assignment for images that occur when the mobile robot platform changed the room, the effect of label noise shall be always present. The robot camera might have captured the visual information depicting the class "One Person Office" while the ground truth information indicates the class "Corridor". This type of error cannot be avoided altogether but its effect can be diminished by a class boundary detection procedure for example.

Some typical misclassification examples are depicted in Fig. 3.4 for BOVW, SimpleMKL and SVM-DAS methods. Empirically we observed that poor light condition and label noise type errors of misclassification dominated for this data set.

3.4.3 Conclusions

In this chapter we studied two major information source fusion approaches - early and late fusion. Early fusion aimed to combine multiple visual cues at image similarity level whereas the late fusion counterpart attempted to provide decision at higher level from multiple independent base classifier outputs.

The experimental evaluation of the baseline methods has shown that both fusion strategies exceed all single feature baselines. The ranking of the methods with respect to performance is

the same at different supervision levels. However, for the database at hand, no cue fusion scheme clearly outperforms the others. We therefore will consider in the next chapter the gain of additional complementary information stemming from unlabeled data, temporal structure of the video and the notion of confidence of classification.

Chapter 4

Time-Aware Co-Training Framework for Image-Based Localization

Contents

4.1	**Introduction**	**58**
4.2	**Learning from labeled and unlabeled data**	**58**
4.3	**Confidence measures**	**69**
4.4	**Proposed Time-Aware Co-Training Framework**	**71**
4.5	**Experiments**	**77**

4.1 Introduction

In this chapter we introduce the framework that combines multiple information sources in a semi-supervised setup including unlabeled data and temporal continuity of the video.

4.1.1 Motivation

In Chapter 2 we reviewed the methods of dimensionality reduction and their practical utility for the classification task. In Chapter 3 we showed the usefulness of the fusion of several visual features. The former can be seen as data preparation step where high-dimensional data is represented in a compact form while taking into account non-linearities in the original feature space. The latter provides more discriminative power due to the complementarity of various visual features.

In this chapter we turn to the problem of semi-supervised learning from weakly annotated videos. We stress the point that information brought by unlabeled data has been implicitly taken into account only in unsupervised manner - through dimensionality reduction. In this chapter we are interested to study and compare several semi-supervised learning methods where unlabeled data is leveraged during the learning stage. Going one step further, we also study the question of using multiple information sources in the same framework.

4.1.2 Outline

The outline of the current chapter is following:

1. Introduction of the State-of-the-Art from semi-supervised method family
2. Contributions
   (a) semi-supervised learning method fusing multiple visual features (CO-DAS);
   (b) enhanced semi-supervised learning method including temporal information (CO-DAS + TA);
   (c) proposed confidence measure with comparison to state-of-the-art;
3. Experimental part confirming the correctness of the selected semi-supervised learning approach

4.2 Learning from labeled and unlabeled data

In this section we attempt to provide a brief review of semi-supervised methods that exploits both labeled and unlabeled data during the learning process. Our goal is not to make an exhaustive review but rather show the place of the co-training method in the field of semi-supervised learning.

Throughout the chapter we will use the notion introduced by [181] for semi-supervised learning.

Suppose that we are given a training set $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and an unlabeled set of patterns $U = \{\mathbf{x}_j\}_{j=l+1}^{l+u}$ where $\mathbf{x} \in \mathcal{X}$ and the problem of classification is binary: $y \in \{-1, +1\}$.

Our visual data may have p multiple cues describing the same image I_i. Suppose that p cues has been extracted from an image I_i

$$\mathbf{x}_i \rightarrow \left(\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)}, \ldots, \mathbf{x}_i^{(p)}\right) \tag{4.1}$$

where each cue $\mathbf{x}_i^{(j)}$ belongs to an associated descriptor space $\mathcal{X}^{(j)}$.

Denote also p decision functions $f^{(1)}, f^{(2)}, \ldots, f^{(p)}$, where $f^{(j)} \in \mathcal{F}^{(j)}$, that are trained on the respective visual cues and are providing estimation $\hat{y}_k^{(j)}$ on the pattern $\mathbf{x}_k^{(j)}$.

Additionally, we are interested in obtaining a confidence measure $z_i \in \mathcal{R}^+$ of the estimation.

4.2.1 Motivation and Definitions

Semi-supervised learning can be seen as an answer to a problem how to learn from large data sets where only a small part of it is labeled by an expert. The scenario with scarce labeling is common in real-world applications where additional labeling is costly or requires significant human labour. The models learned using classic supervised method may suffer from over-fitting or incapability to generalize on the unlabeled data which is direct consequence of the lack of training data. Unsupervised methods do not use label information. They may detect a structure of the data, however, a prior knowledge and correct assumptions about the data is necessary to be able to characterize a structure that is relevant for the task.

Therefore, one needs to employ the methods that are able to learn from labeled and unlabeled parts of the data sets. In the literature there are three major groups of methods:

1. Transductive learning
   Given a labeled set L and an unlabeled set U, the goal is to provide direct predictions for the latter. The usual hypothesis is that the two sets are sampled i.i.d. according to the same joint distribution $p(\mathbf{x}, y)$ to render learning possible. There are no intentions to provide estimations on the data outside the sets L and U. Refer to [27] for more details and references.

2. Semi-Supervised learning
   In similar setup as transductive learning, semi-supervised learning methods are called inductive methods since a function $f : \mathcal{X} \mapsto \mathcal{Y}$ is also learned. Function $f \in \mathcal{F}$, where $\mathcal{F}$ is a hypothesis space, is learned such that predictions can be made on the entire space $\mathcal{X}$, not only on $U \subset \mathcal{X}$. The success of learning depends on an important assumption such that the labeled and unlabeled data comes from the same distribution $p(\mathbf{x})$. Refer to a survey [181] on semi-supervised method and a book [27] devoted to the subject. Short but concise introduction to a subject with brief description and discussion of different methods can be found in [183].

3. Active learning
   Active learning includes the models that build a strong learner interactively from annotations provided by human experts. Usually the methods try to minimize the number of such queries for human annotation while learning the strong learner. Refer to a survey [128] on active learning. Active learning methods are out of the scope of the current study, as in our applicative context annotation is done before automatic processing on the servers such that interactive queries are not possible. It could nevertheless provide interesting avenues to our work in a different annotation context.

In the following subsection we introduce briefly the major approaches for transductive and semi-supervised learning: for transductive learning we present Label Propagation on graphs in Subsection 4.2.2, for semi-supervised learning, semi-supervised SVM in Subsection 4.2.3, self-training in Subsection 4.2.4 and the related co-training method that also combines multiple features in Subsection 4.2.5.

4.2.2 Label Propagation on Graphs

Graph-based semi-supervised learning methods have enjoyed increased popularity. The idea relies on the notion of graph $G = (V, E)$ whose nodes in V are the data points and similarities among

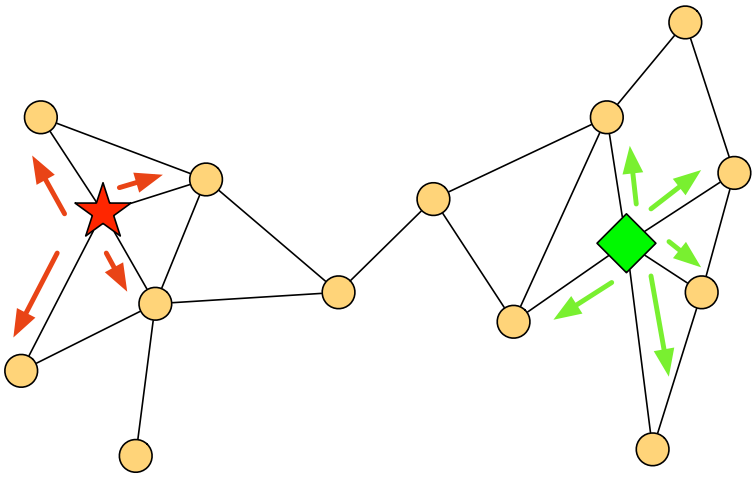


Figure 4.1: Illustration of Label Propagation

them are expressed as links in E. Some of the nodes can be labeled explicitly by a label whereas the labelling is unknown for the remaining unlabeled nodes.

Assuming that the graph is connected or at least one node per connected component is present, the idea is to propagate the labels from labeled to unlabeled nodes. The methods exploiting graph structure to label the remaining part of the graph are classic examples of transductive learning since no direct extension to future patterns is possible. However, several extensions exist [15][146] that allow to provide estimates on additional patterns without rebuilding the whole graph.

Assumptions The key to success using graph-based learning lies in the conformity to one of these two general assumptions:

1. Neighbor nodes carry the same label
   This assumption reflect local properties of a graph since it requires to have local label consistency in a neighborhood of each node.

2. Points on the same structure are likely to have the same label
   This assumption reflects a more global property of a graph. If there are cluster formations or subsets representing a low-dimensional manifold in a graph, then the node labels within them should be consistent.

These two assumptions require a certain structure for the graph. It is clear that not all graphs will conform to these requirements when given a partially labeled data. Therefore, graph construction step is an important factor.

Label Propagation algorithm The idea of label propagation is to spread the labels from the labeled nodes to the unlabeled ones until global convergence is achieved. See Fig. 4.1 for a simple illustration.

Suppose that a labeled set $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and unlabeled set $U = \{\mathbf{x}_i\}_{i=l+1}^{l+u}$ is given from which a graph $G = (V, E)$ is constructed. Graph structure can be completely defined by so-called affinity matrix W where its particular element $W_{ij} = s(\mathbf{x}_i, \mathbf{x}_j)$ is a similarity measure between two graph nodes. The node affinities are often required to be non-negative and $W_{ii} = 0$ to avoid self-reinforcement.

Following [178], denote f a set of $(l+u) \times c$ matrices where c is a number of classes. Class estimation for a pattern $\mathbf{x}_i$ corresponds to

$$\hat{y}_i = \arg\max_{k \leq c} f_{ik} \tag{4.2}$$

where $f_{i\cdot} = (f_{i1}, \ldots, f_{ic})$ can be seen as a scores vector in analogy to SVM scores. Similarly, the label information is given in form of a matrix $Y \in \mathcal{F}$ such that

$$Y_{ik} = \begin{cases} 1 & \text{if } \mathbf{x}_i \text{ is labeled as } y_i = k \\ 0 & \text{else} \end{cases} \tag{4.3}$$

and for unlabeled data in U all the corresponding entries are initialized $Y_{ik} \leftarrow 0$. Finally, denote Y^0 an initial labeling of the data as given by the set L.

With all necessary definitions at hand, label propagation can be written as an iterative procedure:

1. Build an affinity matrix W from both sets L and U using some similarity function between the nodes $s(\cdot, \cdot) \in \mathcal{R}^+$

2. Build a normalized affinity matrix $W_{\text{symm}} = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where $D_{ii} = \sum_j W_{ij}$

3. Set a parameter $\alpha \in [0, 1]$, initial estimation $f^0 = Y^0$ and iterate until convergence $\hat{Y}$

   (a) Compute

   $$f^{t+1} = \alpha W_{\text{symm}} f^t + (\alpha - 1) Y^0 \tag{4.4}$$

   (b) Current label estimates can be computed for each pattern $\mathbf{x}_i$ as

   $$\hat{y}_i^{t+1} = \arg\max_{j \leq c} f^{t+1} \tag{4.5}$$

4. Use the convergence result $\hat{Y}$ to compute the final estimates

Usually the labeled part of current estimate at iteration $t+1$ is clamped. However, authors in [27] revealed that relaxing this constraint helps to work with data sets featuring class overlap.

Convergence properties of label propagation methods are proved and discussed in [178, 27]. An important result shows that the final iteration result can be found in a closed form using the initial annotation Y^0 and the graph W_{symm}

$$\hat{Y} = (I - \alpha W_{\text{symm}})^{-1} Y^0 \tag{4.6}$$

Note that a properly constructed graph is needed as well as minimum annotation to bootstrap the label propagation. The balancing parameter α controls the importance of the initial labeling (second term in Eq. 4.4) with respect to the information provided by the global graph structure (first term in Eq. 4.4). This parameter should be set manually or using some automatic scheme such as cross-validation, for example.

Regularization on Graphs Label propagation method presented above uses an update Eq. 4.4 which may not be immediately clear. Recall that for successful semi-supervised learning two assumptions should hold, that is, local and global properties of the graph. In semi-supervised learning literature a cost function c is used. The function $f \in \mathcal{F}$ is selected such that the cost is minimized

$$\hat{f} = \arg\min_{f \in \mathcal{F}} c\left(\mathbf{x}, y, f\right) \tag{4.7}$$

Cost function that corresponds to the update step in Eq. 4.4 which takes into account the both constraints takes the following form [178]

$$c\left(\mathbf{x}, y, f\right) = \frac{1}{2}\left(\lambda \sum_{i=1}^{l} \left\|f_i - Y_i\right\|^2 + \sum_{i,j=1}^{l+u} W_{ij} \left\| \frac{f_i}{\sqrt{D_{ii}}} - \frac{f_j}{\sqrt{D_{jj}}} \right\|^2\right) \tag{4.8}$$

The cost function could take into account several elements. One can require the function $\hat{f}$ to be coherent with the training data, known as fitting constraint. Additionally, one can require at the same time the fitness of the function with respect to the whole graph. This constraint is often called in literature as a smoothness or manifold constraint. That is, labels on the whole graph are not expected to vary greatly within certain neighborhood.

Introduction of regularization parameter λ changes the closed form solution as given in 4.6 to

$$\hat{Y} = \frac{\lambda}{1+\lambda}\left(I - \frac{1}{1+\lambda}L\right)^{-1} Y^0 \tag{4.9}$$

with derivation details given in [178] and more in depth discussion in [27] on regularization framework and links to a straight forward label propagation [182].

The regularization done in Eq. 4.8 uses a quadratic cost penalty term. A different regularization which is based on Laplacian as an operator on the functions f was proposed in [12] together with corresponding label propagation algorithm. The idea is to use a linear combination of k smoothest eigenfunction of the Laplacian operator such that it fits the labeled data the best.

Building a Robust Graph For a label propagation method to succeed, a correctly built graph is mandatory. Clearly, misleading links and their weights will direct the label propagation process in the wrong direction and the classification results will suffer. In Chapter 2 we briefly mentioned and evaluated two graph building strategies, when reviewing the Laplacian Eigenmap approach in Chapter 1: full and k-Nearest Neighbor sparsened graphs. We review one additional method for robust graph construction which will be evaluated in the experimental part of the chapter.

In [163] authors propose to build a sparse graph from pasted locally reconstructed patches. That is, for every node $\mathbf{x}_i$ in the graph G a small neighborhood $N\left(\mathbf{x}_i\right)$ is found. Then the node $\mathbf{x}_i$ is approximated from its neighbors linearly

$$\mathbf{x}_i \approx \sum_{j:\mathbf{x}_j \in N(\mathbf{x}_i)} w_{ij}\mathbf{x}_j \tag{4.10}$$

where we constrain $\sum_j w_{ij} = 1$ and $w_{ij} \geq 0$. A weight closer to 1 will indicate a close node while a distant node will receive a weight which is close to zero. Therefore, the reconstruction weights can serve as affinity values in the complete affinity matrix W representing the graph G.

For a selected neighborhood selection rule, the goal is to deduce the reconstruction weights for each node of the graph out from its neighbors. Intuitively, the reconstruction error can be minimized

$$e = \sum_{i=1}^{n} \left(\mathbf{x}_i - \sum_{j:\mathbf{x}_j \in N(\mathbf{x}_i)} w_{ij}\mathbf{x}_j \right)^2 \tag{4.11}$$

which after some transformations can be rewritten as

$$e = \sum_{j,k:\mathbf{x}_j,\mathbf{x}_k \in N(\mathbf{x}_i)} w_{ij} G^i_{jk} w_{ik} \tag{4.12}$$

where $G^i_{jk} = (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_k)$ represents a patch centered on the node $\mathbf{x}_i$.

We note that the error function is quadratic in reconstruction weights w with its constraints. This leads to a Quadratic Program [102] with linear constraints. Therefore, for every node $\mathbf{x}_i$ following problem should be solved

$$\min_{w_{ij}} \quad \textstyle\sum_{j,k:\mathbf{x}_j,\mathbf{x}_k \in N(\mathbf{x}_i)} w_{ij} G^i_{jk} w_{ik} \tag{4.13}$$

$$\text{s.t.} \qquad \textstyle\sum_j w_{ij} = 1, w_{ij} \geq 0 \tag{4.14}$$

The final sparse affinity matrix W is built by pasting together the corresponding reconstruction weights.

4.2.3 Semi-Supervised Support Vector Machines

We reviewed the fully supervised SVM in Chapter 2. In low supervision scenarios a decision boundary found using only the labeled part of the data set may not be the optimal solution and inclusion of unlabeled data may be beneficial. For example, the assumption for a decision boundary to pass in low density regions can help to reveal the underlying manifold with class-specific clusters. A toy example illustrating this situation is shown in Fig. 4.2. This idea motivated the development of semi-supervised SVM, see [91] and the references therein.

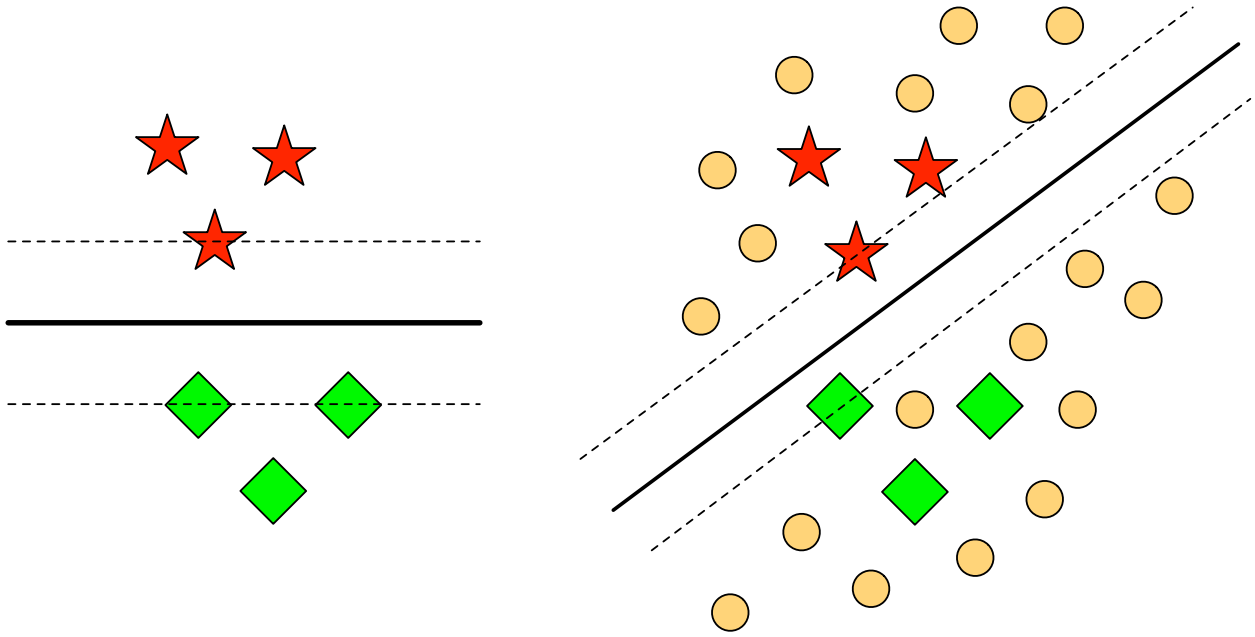
Literature review In semi-supervised learning literature, the cluster [29] and manifold [13] assumptions are the most often employed when staying within the SVM paradigm.

The cluster assumption states any two points will more likely have the same label if they both lie in a dense region through which these points can be connected by a path. Therefore decision boundaries should push in lower density regions in the feature space. Some references for Transductive SVM formulation with the cluster assumption include [65, 140] and optimization methods for semi-supervised methods can be consulted in [28].

The manifold assumption leverages the knowledge that the data may actually lie on low-dimensional manifold and to discover it, a smooth function respecting labeled and unlabeled data should be used. The methods in this family are typically graph-based [12, 13, 91, 95] and exploit the graph structure to find a regularized solution.

Including graph geometry into SVM In Chapter 2 we review the method of Laplacian Eigenmaps for dimensionality reduction which is intrinsically an unsupervised procedure. The idea of the method is to preserve the geometry captured by a graph when computing the lower dimensional representations.

Information about the manifold computed from the labeled data $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and unlabeled data $U = \{\mathbf{x}_i\}_{i=l+1}^{l+u}$ can be expressed in a form of affinity matrix $W_{ij} = s(\mathbf{x}_i, \mathbf{x}_j)$. The Laplacian



(a) Supervised SVM **(b) Semi-Supervised SVM**

Figure 4.2: Toy example with decision boundary found in (a) fully supervised setup on a limited training set and (b) with the help of unlabeled data using Semi-Supervised SVM

SVM (LapSVM) [91] method uses a regularizer based on the Laplace-Beltrami operator that can be effectively approximated by a graph Laplacian L. A decision function that conforms both to labeled data and unlabeled data writes as

$$\hat{f} = \arg\min_{f \in \mathcal{H}_K} \frac{1}{l} \sum_{i=1}^{l} c\left(\mathbf{x}_i, y_i, f\right) + \gamma_A \left\|f\right\|_K^2 + \frac{\gamma_I}{(l+u)^2} \sum_{i,j=1}^{l+u} W_{ij} \left(f\left(\mathbf{x}_i\right) - f\left(\mathbf{x}_j\right)\right)^2 \tag{4.15}$$

where three parts can be recognized :

- First term accounts for standard empirical loss in fully supervised setup. The loss function can be, for example, the squared error, hinge, Huber loss yielding different SVM formulations;
- Second term is an ambient regularization term that controls the complexity of the learned function. Together with the first term a standard supervised SVM formulation can be obtained;
- The third term introduces the information provided by the unlabeled data. One can recognize up to multiplicative factor the objective function to be minimized for Laplacian Eigenmaps.

Note the similarity of the optimization problem in Eq. 4.15 with that of Label Propagation in Eq. 4.8. The framework is the same while the cost functions are different, squared loss for LP and soft margin loss for LapSVM. Also in both cases, the regularization terms share a Laplacian type regularizer with the difference of normalization. More discussion about the choice of loss function can be found in [59] and the details about different types of Laplacian normalization in [162].

Impact of the regularization Eq. 4.15 states that functions $\hat{f}$ that are consistent with labeled data and geometrical structure of the graph G may be accepted as a solution. It is interesting to note that the solution to Eq. 4.15 can be still written using the Representer Theorem [27]

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{l+u} \alpha_i y_i \tilde{k}(\mathbf{x}_i, \mathbf{x}) + b \tag{4.16}$$

with a difference that the expansion now includes unlabeled patterns as well.

Note the kernel function $\tilde{k}$ in the expansion Eq. 4.16. Recall that for each kernel function $k(\cdot,\cdot)$ there exists a corresponding RKHS with functions $f, g \in \mathcal{H}_K$ endowed with dot product $\langle f, g\rangle_{\mathcal{H}_K}$. Inclusion of the two additional regularization terms in Eq. 4.15 reflects on the functions f, g living in the RKHS $\mathcal{H}_K$. The new space corresponding to the kernel $\tilde{k}$ now has a modified dot product for $f, g \in \tilde{\mathcal{H}}_K$

$$\langle f, g\rangle_{\tilde{\mathcal{H}}_K} = \langle f, g\rangle_{\mathcal{H}_K} + \frac{\gamma_I}{\gamma_A} \boldsymbol{f}^T L \boldsymbol{g} \tag{4.17}$$

where $\boldsymbol{f}, \boldsymbol{g}$ are function evaluations, for example, $\boldsymbol{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_{l+u}))^T$ and $L = D - W$ is an unnormalized graph Laplacian. Finally, the new kernel corresponding to the modified dot product can be computed using the

$$\tilde{K}(\mathbf{x}_i, \mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j) - \boldsymbol{k}_{\mathbf{x}_i}^T \left(I + \frac{\gamma_I}{\gamma_A} LK \right)^{-1} L \boldsymbol{k}_{\mathbf{x}_j} \tag{4.18}$$

where $\boldsymbol{k}_{\mathbf{x}_i} = (k(\mathbf{x}_1, \mathbf{x}_i), \ldots, k(\mathbf{x}_{l+u}, \mathbf{x}_i))^T$ and similarly for vector $\boldsymbol{k}_{\mathbf{x}_j}$.

Thus, the kernel $\tilde{K}$ can be used in a standard supervised SVM but the solution will include also the information brought by unlabeled data. This result allows to see what is the effect of introducing a certain regularization in the framework. For more discussion on the subject refer to [27] and the references therein.

The formulation of the Laplacian Semi-Supervised SVM problem and its solution is detailed in Annex C.

Issues Semi-supervised learning may pose certain risks and even degrade the performance [133], for example when learning from high-dimensional data spaces. It has been recently found that using Laplacian regularizer can actually lead to degenerate solutions as unlabeled data provided for learning approaches infinity [95]. Solution to the problem was proposed in [179] by using regularization based on an iterated Laplacian.

Direct implementation of LapSVM may suffer from scalability issues. Working with dense kernel matrices scales as $O(n^2)$ for storage and for computation needs as high as $O(n^3)$ [27]. Clearly, practical application of the described method is limited currently from small to moderatel size data sets. However, if resorting to linear kernel machines or the high-dimensional data is highly sparse, then optimization problems can be solved more efficiently in primal [91]. An efficient linear kernel transductive method was proposed in [132] solving the problem in primal using a finite Newton method for optimization.

4.2.4 Learning with self-training

Idea Previous approaches are based on the optimization of a global criterion, that extends and modifies unsupervised and purely supervised classifiers. In contrast, the Self-training [172, 60, 61] is a wrapper method that itself is based on existing classifiers. The idea is to iteratively increase the

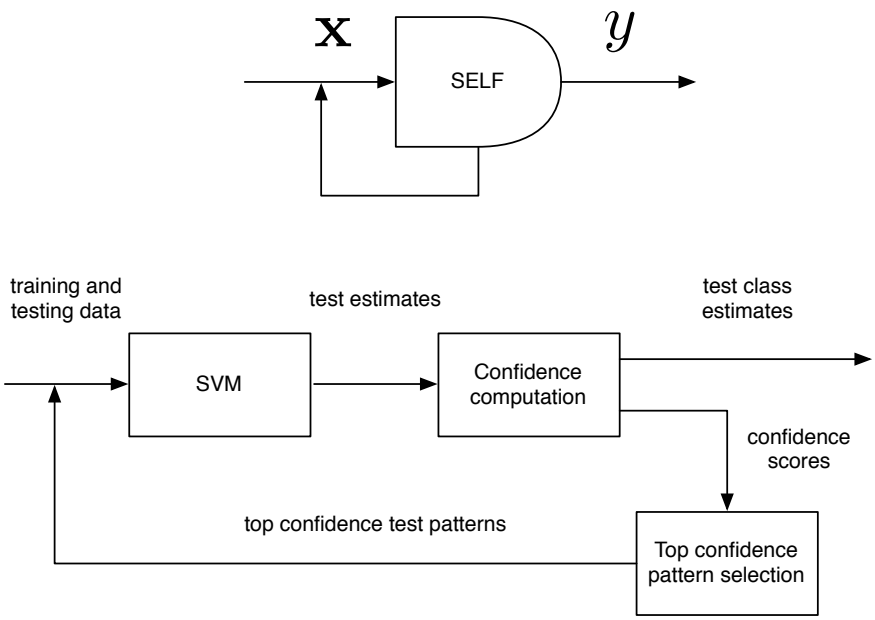


Figure 4.3: Self-training module

training set by including at each iteration the most confident estimations from the previous output of the base classifier.

Literature review The self-training approach was proposed as Yarowsky algorithm in [172, 60, 61] for text processing and for credit score inference using SVM in [89] where one needs to assess the risk of providing a loan. A generalized iterative self-training framework with a study of convergence properties was studied in detail in [38] and extended different supervised learners like kernel smoothers, generalized additive models and classification methods into a semi-supervised learner.

Algorithm Given a labeled pattern set L and an unlabeled pattern set U, where $|L| = l$ and $|U| = u$, the algorithm consists of the following steps:

1. Define a training set $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$
2. Define the training rate parameter $0 < k < l + u$
3. Repeat until the unlabeled pattern set U is empty
   (a) Train a classifier $f(\cdot)$ using the current training set L;
   (b) Classify the unlabeled patterns in the set U and obtain estimations and confidence measure $\{(\hat{y}_i, z_i)\}_{i=1}^{|U|}$;
   (c) Add the k top confidence patterns together with estimation into the set L;
   (d) Remove the k top confidence patterns from the set U. Go to step 3.

Graphically the method is depicted in Fig. 4.3.

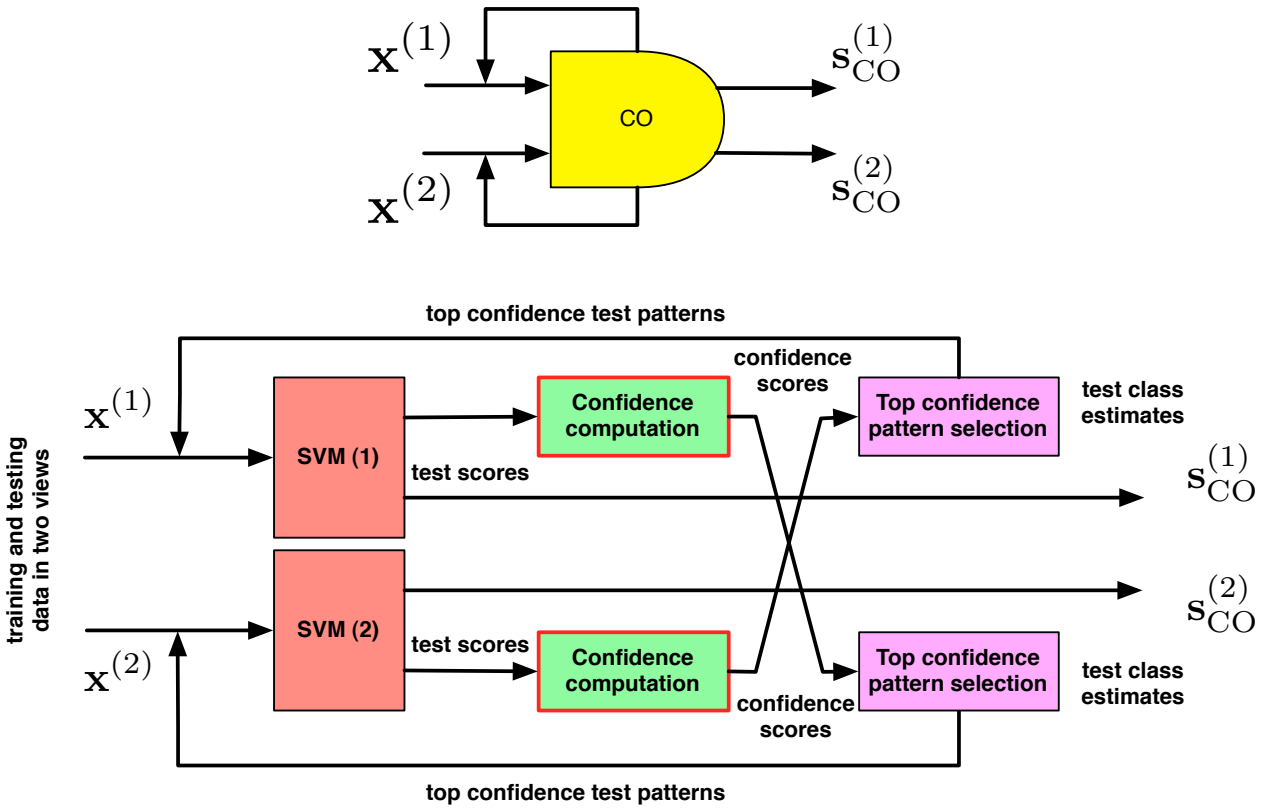


Figure 4.4: Co-Training module

Assumptions The performance of self-training algorithm depends on two implicit assumptions:

1. The training set is large and representative enough to train a good initial classifier;
2. The top confidence estimations on the unlabeled patterns are correct;

The first assumption is simple and basically requires a good start up estimations. The second assumption is crucial since the procedure is iterative and including erroneous patterns will only degrade the final performance of the classifier. There is a risk of error reinforcement in the early training iterations, especially for small sized training sets. The method typically works well if the data forms well defined class clusters.

Experiments We present and discuss the experimental results using the self-training in the section 3.4.

4.2.5 Learning with co-training

Idea Co-training [17] is another wrapper method that leverages two cues where each such set is sufficient to train a classifier. In this setup one classifier "teaches" another classifier by supplying it with high confidence estimations on the unlabeled samples.

Literature review The method of co-training was proposed in [17] as a solution to classify Web pages using both links between the pages and both the words present in the web pages. The same method was applied to the problem of Web image annotation [175, 147] and automatic video

annotation [164]. Generalization capacity of co-training on different initial labeled training sets was studied in [153]. More analysis on theoretical properties of co-training method can be found in [165] for rough estimates of maximal number of iterations. A review on different variants of the co-training algorithm is given [42] together with their comparative analysis.

It interesting to note the link of both self-training and co-training methods to label propagation in a graph as discussed in [3, 172]. Similarly, the co-training method was presented in [166] as a label propagation method on a combined graph built from individual views. Sufficient and necessary conditions for the co-training to succeed were discussed in the same works in details as well.

Algorithm Suppose that a pattern consists of two parts where each part corresponds to one view

$$\mathbf{x} = \left(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}\right) \tag{4.19}$$

Given a labeled pattern set L and an unlabeled pattern set U, where $|L| = l$ and $|U| = u$, the algorithm consists of the following steps:

1. Define two sets : $L_1 = \left\{\left(\mathbf{x}_i^{(1)}, y_i\right)\right\}_{i=1}^{l}$ and $L_2 = \left\{\left(\mathbf{x}_i^{(2)}, y_i\right)\right\}_{i=1}^{l}$
2. Define the training rate parameter $0 < k < l + u$
3. Repeat until the unlabeled pattern set U is empty
   (a) Train the classifier $f^{(1)}(\cdot)$ using the set L_1;
   (b) Train the classifier $f^{(2)}(\cdot)$ using the set L_2;
   (c) Classify the patterns in the set U using the classifiers $f^{(1)}(\cdot)$ and $f^{(1)}(\cdot)$ independently;
   (d) Add the k top confidence estimations from the classifier $f^{(1)}(\cdot)$ into the set L_2;
   (e) Add the k top confidence estimations from the classifier $f^{(2)}(\cdot)$ into the set L_1;
   (f) Remove the k top confidence patterns from the set U. Go to the step 3

Graphically the method is depicted in Fig. 4.4.

Assumptions Clearly, the performance of such approach is dependent on the quality and the amount of available training data, usefulness of unlabeled data at hand and the measure of confidence. Indeed, authors in [17] state two assumptions which are necessary for the method to work:

1. The classifiers $f^{(1)}$ and $f^{(2)}$ trained on initial training sets L_1 and L_2 should provide good enough initial estimations;
2. The two cues should be conditionally independent

$$P\left(\mathbf{x}^{(1)}|y, \mathbf{x}^{(2)}\right) = P\left(\mathbf{x}^{(1)}|y\right) \tag{4.20}$$

$$P\left(\mathbf{x}^{(2)}|y, \mathbf{x}^{(1)}\right) = P\left(\mathbf{x}^{(2)}|y, \right) \tag{4.21}$$

The first assumption is necessary to have the algorithm bootstrapped. Provided a reliable confidence measure, few but correct estimations can be sufficient for the algorithm to label the rest of the unlabeled set iteratively.

The second assumption is important due to complementary nature of the two cues. If this assumption does not hold, then, for example, a classifier $f^{(1)}$ may provide new training patterns that are highly similar and thus non-informative for the classifier $f^{(2)}$.

Experiments The results using co-training method are presented and discussed in section 3.4.

4.2.6 Conclusion

In this section we reviewed three approaches for semi-supervised learning: semi-supervised SVM, Label Propagation in a Graph and Co-Training. These methods will serve as a baseline for the experimental part of this chapter.

4.3 Confidence measures

The primary interest in classification methods is to obtain the class estimates. Some applications also require or can benefit from also estimating a confidence measure that expresses a belief in the correctness of the prediction. Therefore, classification outputs can be treated differently, for example, divided into accepted and rejected groups.

4.3.1 Literature review

The performance of a refined classifier after applying self-training or co-training round will strongly depend on the quality of confidence measure used to augment the training set with new labeled patterns. Without claim to be comprehensive and complete, we can group the methods using or providing the notion of confidence:

1. Using classic Bayesian decision theory
   The idea is to model the posterior probabilities in a Bayesian framework such that they can be used as confidence measure. In [17] used Naive Bayes approach for classification [46] to derive posterior outputs as a confidence measure. For more classical decision theory refer to broad literature [46, 59, 156, 155].

2. Derivation from SVM scores
   A classical and empirically well performing result for turning SVM scores into probabilities was proposed in [104] with implementation details in [78]. Very competing performance was demonstrated using a theoretically motivated non-parametric method in [120] which uses a simple linear relationship for a region between margins and a simple counting outside. A binning approach was used in [43] where score values are discretized into pre-defined bins and the conditional probabilities are computed on per bin basis.
   A body of work was done in [4, 5, 90, 9] for speech recognition application with SVM decision values converted into posterior probabilities or confidence measure by a simple Neural Network. We separately note the definition of confidence measures in [112]. A modified One-vs-All setup was proposed such that a test pattern is not compared to the margins in the feature space but rather to class means found from the training set.

3. Cross-Validation
   A principally different approach uses cross-validation to infer confidence measure in [55, 180]. Recently, the labeling confidence was measured in [176] using data editing techniques that improves the training set by removing probably incorrectly labeled training patterns.

4.3.2 Derivation of confidence measures in the context of the SVM classifier

An SVM-based classifier does not provide the confidence measure of classification out-of-the-box. Recall that an one-vs-all SVM classifier provides the scores based on the values of the decision

function assigned to each class

$$f^k(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b, k = 1, \ldots, c \tag{4.22}$$

where c is the number of classes, l is the number of training patterns and $f^k(\mathbf{x}) > 0$ if the pattern $\mathbf{x}$ belongs to the k-th class. The estimated class is usually estimated from the largest positive score for one-vs-all setup

$$\hat{y}_i = \arg \max_{k=1,\ldots,c} f^k(\mathbf{x}_i) \tag{4.23}$$

A majority voting is used instead for one-vs-one setups. The scores $\left(f^1(\mathbf{x}), \ldots, f^c(\mathbf{x})\right)^T$ are not normalized and cannot directly be used as confidence estimates. The goal is to provide probability $P(y|\mathbf{x})$ or a confidence measure $z \in \mathcal{R}^+$ classifying a pattern $\mathbf{x}$ in a class y.

We review several methods computing a confidence measure out of the SVM outputs.

Logistic model ("Logistic") Following [59], class probabilities can be computed using the logistic model that generalizes naturally to multi-class classification problem. Suppose that in one-vs-all setup with c classes, the scores $\left\{f^k(\mathbf{x})\right\}_{k=1}^{c}$ are given. Then probability or classification confidence is computed as

$$P(y = k|\mathbf{x}) = \frac{\exp\left(f^k(\mathbf{x})\right)}{\sum_{i=1}^{c} \exp\left(f^i(\mathbf{x})\right)} \tag{4.24}$$

which ensures that probability is larger for larger positive score values and sum to 1 over all scores. In this model, uncertainty can be detected when several classes obtain the scores of similar values. Unbalanced scores, even all negatives, can yield a high probability for the best class.

Modeling posterior class probabilities ("Ruping") In [120] a parameter-less method was proposed which assigns score value

$$z = \begin{cases} p_+ & f(\mathbf{x}) > 1 \\ \frac{1+f(\mathbf{x})}{2} & -1 \leq f(\mathbf{x}) \leq 1 \\ p_- & f(\mathbf{x}) < 1 \end{cases} \tag{4.25}$$

where p_+ and p_- are the fractions of positive and negative score values respectively.

Score difference ("Tommasi") A method that does not require additional pre-processing for confidence estimation was proposed in [145]. The idea is to use the contrast between the two top uncalibrated score values. Suppose that in a multi-class classification problem with c classes, the maximal uncalibrated distance for a pattern $\mathbf{x}$ was found

$$k^* = \arg \max_{k=1,\ldots,c} f^k(\mathbf{x}) \tag{4.26}$$

The maximum score estimation should be confident, meaning that other score values are relatively smaller. This leads to a confidence measure using the contrast between the two maximum scores

$$z = f^{k^*}(\mathbf{x}) - \max_{k=1,\ldots,c,k\neq k^*} f^k(\mathbf{x}) \tag{4.27}$$

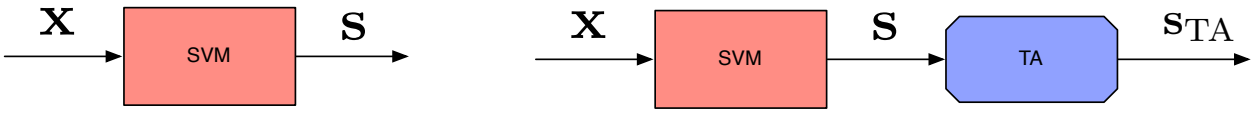


Figure 4.5: Baseline SVM architecture (left) and Temporal Accumulation (TA) architecture (right) for single feature classification

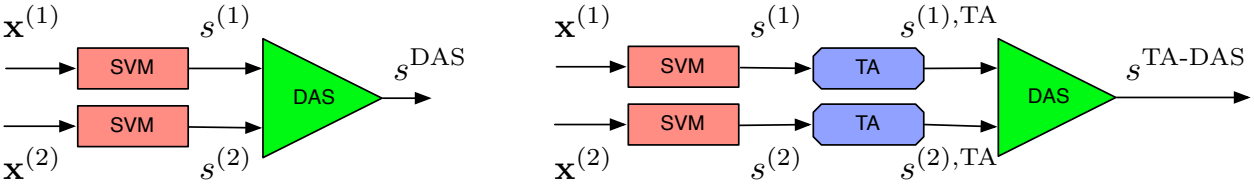


Figure 4.6: Baseline DAS multiple feature fusion (left) and TA-DAS fusion (right) methods

Since only the contrast is used, a high confidence could be assigned to a class with a high score, even if another class has also a positive score. The same could happen to a class with negative score if other classes receive much lower scores.

In [145] this measure was thresholded to obtain a decision corresponding to "no action", "reject" or "don't know" situation for medical image annotation.

Conclusion These three confidence measures are deduced from discriminative nature SVM classifier. We noticed that class overlap and reject situations are not actually taken into account. From this analysis, we will propose a class overlap sensitive confidence measure in Subsection 4.4.4.

4.4 Proposed Time-Aware Co-Training Framework

Introduction We successfully confirmed the utility of multiple feature fusion in Chapter 3. Nevertheless, two aspects were omitted: temporal continuity of the video and relatively small amounts of annotation.

We now propose an unified framework to combine multiple visual features while leveraging the available unlabeled data and integrate the temporal constraints implicitly provided by the video.

4.4.1 Temporal Accumulation: Enforcing Temporal Video Continuity constraints

Idea Video content has a temporal nature such that the visual content does not change much for a short period of time. In the case of topological localization indoors this constraint may be useful as localization changes are encountered relatively rarely with respect to the frame rate of the video.

We therefore propose to modify the classifier output such that rapid class changes are discouraged in a relatively short period of time. This leads to lower proliferation of occasional temporally localized misclassifications.

Principle of Temporal Accumulation (TA) Let $s_i^t = f^{(t)}(\mathbf{x}_i)$ be the output of a binary classifier for visual cue t and h a temporal window of size $2\tau + 1$. Then temporal accumulation can be written as

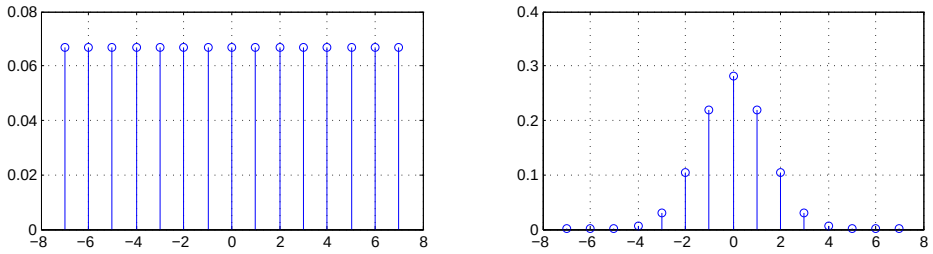


Figure 4.7: Sample windows of size 15: (left) averaging, (right) Gaussian $\left(\sigma = \sqrt{2}\right)$

$$s^t_{i,TA} = \sum_{k=-\tau}^{\tau} h\left(k\right) s^t_{i+k} \tag{4.28}$$

and can be easily generalized to multiple feature classification by applying it separately to the output of the classifiers associated to each feature $\boldsymbol{s}^t$, where $t = 1, \ldots, p$ is the feature type. An open question is how to select the window h and its size. Natural choices may be:

1. Averaging filter

$$h\left(k\right) = \frac{1}{2\tau + 1}, k = -\tau, \ldots, \tau \tag{4.29}$$

   An example window with $k = 15$ is shown in Fig. 4.7 in the left panel.

2. Gaussian filter

$$h\left(k\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{k^2}{2\sigma^2}\right), \text{ with } \tau \geq 3\sigma \tag{4.30}$$

   where σ is the standard deviation of the Gaussian. Example window with $k = 15$ and $\sigma = \sqrt{2}$ is shown in Fig. 4.7 in the right panel.

Parameters The Temporal Accumulation (TA) module requires setting the size τ for the averaging filter or the bandwidth σ for the Gaussian filter (σ). By adding the TA module to the baseline SVM, we define the TA method as shown in right panel in Fig. 4.5. By adding the TA module before the DAS fusion scheme (see Section 3.3.4), we define the TA-DAS module as depicted in right panel in Fig. 4.6.

4.4.2 CO-DAS : Semi-Supervised learning from multiple visual features and classifier fusion

Idea The standard Co-Training method introduced in Section 4.2.5 allows to benefit from the information in the unlabeled part of the corpus by using it in a feedback loop to augment the training set. Often it is not known beforehand which classifier performs the best and if the complementarity properties between the two has been leveraged to its maximum. The proposed method addresses this issue by providing a single output using late classifier fusion.

In principle the method consists of two modules: Co-Training and DAS. The two visual cues are efficiently exploited in the semi-supervised setting of the co-training module while the DAS module ensures the fusion of the two decision functions. The co-training module can be iterated for multiple rounds prior the final fusion. Graphically the method is portrayed in Fig. 4.4.

The power and assumptions of the method The power of the method lies in its capability of learning from small training sets and grow eventually its discriminative properties on the large unlabeled data set as more confident estimations are added into the training set. The following assumptions are made:

1. the two distinct visual cues bring complementary information;
2. the initially labeled set is sufficient to bootstrap the iterative learning process;
3. the confident estimations on unlabeled data are helpful to predict the labels of the remaining unlabeled data;

An attractive property of the co-training method is that the confidence is estimated using the discriminative classifier and not directly from the raw features. Graph-based methods are indeed limited by the low-level type of the links that are taken into account. The co-training method uses a discriminative model that can be adaptive to the data, thus achieving a higher level of discrimination.

Parameters The method is inherently of high level since any confidence-rated classifier $\{s_i, z_i\} = h(\mathbf{x}_i, \boldsymbol{\theta})$ can be used where we denote

- $\mathbf{x}_i$ - a test pattern;
- $\boldsymbol{\theta}$ - a vector of parameters for the classifier;
- s_i - classifier output;
- z_i - a confidence score associated to the classification result;

In the case of the SVM classifier, parameter vector $\boldsymbol{\theta}$ represents the learned SVM model: $\boldsymbol{\alpha} \in \mathcal{R}^n$ and $b \in \mathcal{R}$. To learn a model, a Gram kernel matrix K is needed which is computed from the training set L and regularization parameter C for soft-margin SVM.

The co-training method requires a number of iterations N or a stopping criterion. Stopping criteria may be a rule that stops the iterative learning process when there are no confident estimations to add or there have been relatively small difference between iterations $t-1$ and t. The parameter-less version of co-training works till the complete exhaustion of the pool of unlabeled samples.

The method is compared to the baseline semi-supervised learning methods in section 3.4 devoted to the experiments.

4.4.3 Time-Aware CO-DAS : Injection of temporal information into the learning loop

Idea The proposed CO-DAS method does not take into account the temporal structure of the video. We show that the temporal information can be efficiently leveraged while learning a new appearance model.

The method consists of three modules: co-training, temporal accumulation and DAS. Similarly to the CO-DAS, the method exploits two visual cues in semi-supervised setup and final decision fusion. The difference lies in exploitation of the temporal information. Temporal information can be injected in different parts of the learning chain:

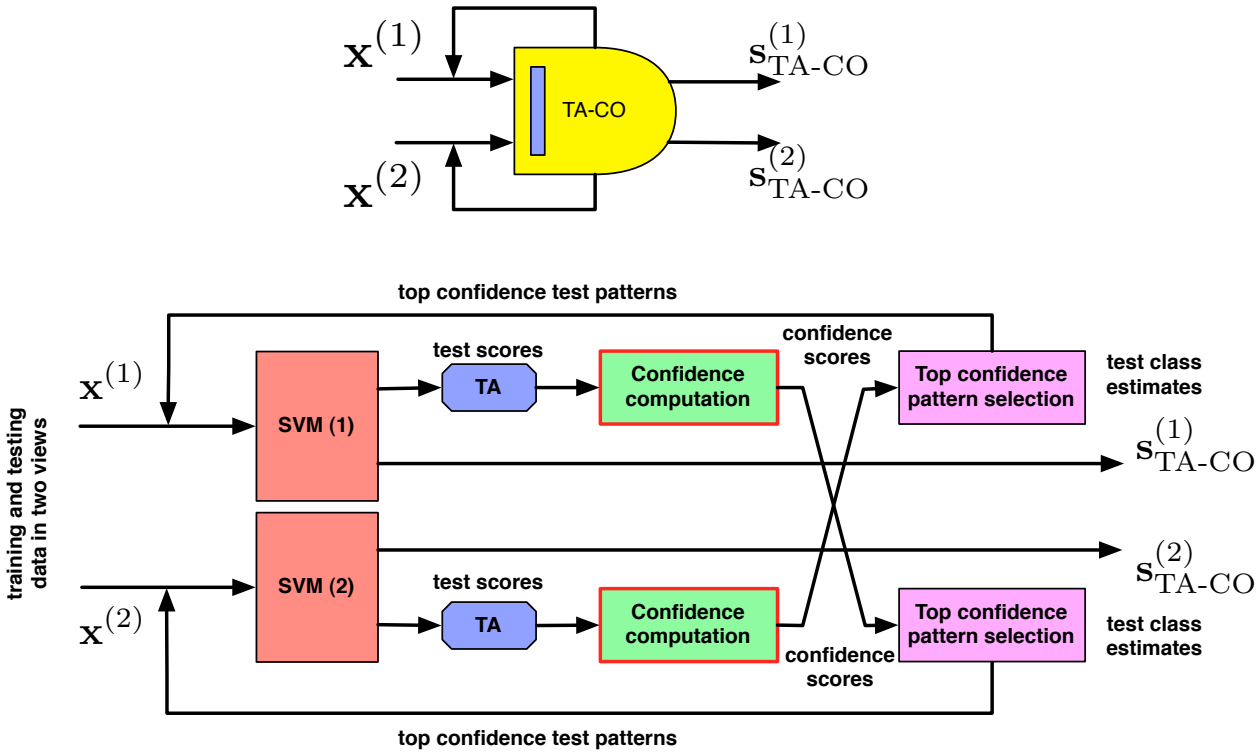


Figure 4.8: Time-Aware CO-DAS module

1. inside the learning loop of co-training, denoted as the TA-CO-DAS method
   We consider the TA module inserted inside the CO module between SVM score computation and confidence computation, as depicted in Fig. 4.8.

2. after co-training and before the DAS module, denoted as the CO-TA-DAS method

3. in both places, denoted as the TA-CO-TA-DAS method

All three proposed methods are represented schematically in Fig. 4.9.

The power and assumptions of the method These methods can be seen as an evolution of CO-DAS, therefore the same assumptions and considerations apply. The improvement stems from intelligent temporal information utilization which is not limited to a mere result post-processing. When a small portion of test patterns with top confidence estimates are added to the training set, temporal accumulation applied at this point injects also their temporal neighbor information.

The method is compared to the CO-DAS method as well as baseline semi-supervised learning methods in section 3.4.

4.4.4 Proposition of a new Class overlap sensitive confidence measure

The one-vs-all setup for multiple class classification is prone to yield ambiguous decisions. That is, it is possible to obtain several positive scores or even all positive or all negative scores.

Following the analysis from Section 4.3, we propose a confidence measure that penalizes class overlap (ambiguous decisions) at several degrees and also treats both degenerate cases. By convention, confidence should be higher if a sample is classified with less class overlap (fewer positive score

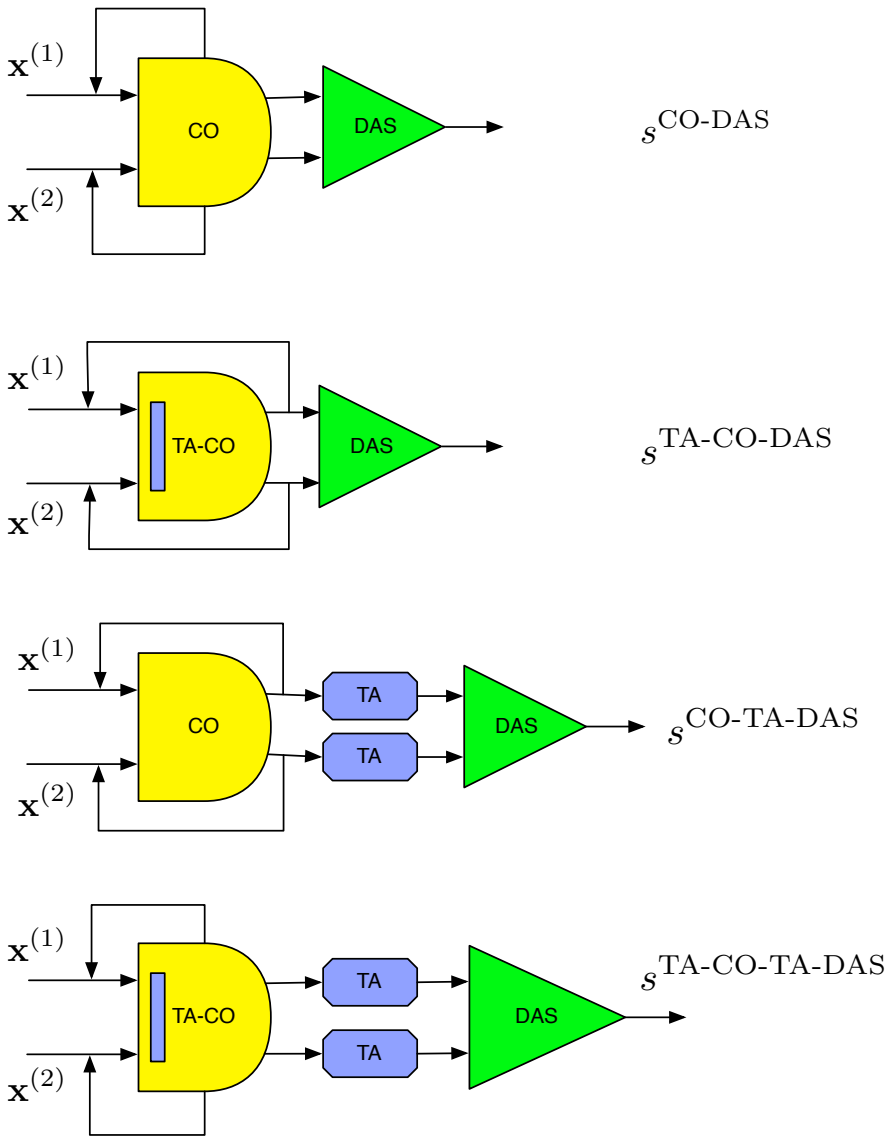


Figure 4.9: Proposed fusion method (a) CO-DAS; Time-aware learning methods (b) TA-DAS, CO-TA-DAS (c) TA-CO-DAS and (d) TA-CO-TA-DAS

Method / Module	Score fusion (DAS)	TA in the loop	TA before DAS	Co-Training module
DAS baseline	X			
TA-DAS	X		X	
CO-DAS	X			X
CO-TA-DAS	X		X	X
TA-CO-DAS	X	X		X
TA-CO-TA-DAS	X	X	X	X

Table 4.1: Compositions of various time-aware co-training methods

values) and further from the margin (larger positive value of a score). Cases with all positive or negative scores may be considered as degenerate $z_i \leftarrow 0$.

The computation is divided in two steps. First we define the contrast between the best score and the rest

$$z_i^0 = f^{j^*}(\mathbf{x}_i) - \max_{i=1,\ldots,c, i \neq j^*} f^i(\mathbf{x})$$

Then the measure z_i^0 is modified to account for class overlap

$$z_i = z_i^0 \max\left(0, 1 - \frac{p_i - 1}{C}\right)$$

where $p_i = \text{Card}\left(\left\{k = 1, \ldots, c | f^k(\mathbf{x}_i) > 0\right\}\right)$ represents the number of classes for which $\mathbf{x}_i$ has positive scores (class overlap). In case of $\forall k$, $f^k(\mathbf{x}_i) > 0$ or $f^k(\mathbf{x}_i) < 0$, we set $z_i \leftarrow 0$.

Compared to the Tommasi measure, the proposed measure specifically penalizes class overlap in order to avoid assigning a high confidence to a sample that would have several positive score values. Conversely, compared to the logistic measure, samples with no positive scores yield zero confidence, which allows to exclude them and not assign doubtful probability values.

4.4.5 Strategies for Visual and Temporal Information Fusion

With necessary methods defined, we can therefore list three big groups of methods:

1. Standard base classifier fusion
   The framework simplifies to baseline DAS [116] (See Fig. 4.6 in left panel) and its variant with temporal score accumulation : TA-DAS (See Fig. 4.6 in right panel).

2. Co-Training enabled
   The framework also includes the co-training module as shown in Fig. 4.9 in (a) and (c) panels. The respective methods are CO-DAS and CO-TA-DAS.

3. Temporal information injection into the learning loop
   In addition to Co-Training and DAS fusion, we inject temporal information in the feedback loop. This gives a rise to the TA-CO-DAS and TA-CO-TA-DAS methods (See Fig. 4.9 in (b) and (d) panels).

Note that all inputs and outputs of the modules operate with SVM outputs (scores), though it may not be the case using a different base classifier. See Table 4.1 for a summary of the named methods and which modules were used to build each of them.

4.5 Experiments

In this section we conduct the experiments related to the methods introduced in this chapter. Initially, the goal is to show the baseline performances obtained using label propagation in the graph, semi-supervised SVM and co-training. Then these results are compared to the contribution framework featuring multiple modality data fusion in semi-supervised setup enhanced with temporal constraints.

This section is organized as follows:

1. Data and test setup;
2. Study of confidence measures;
3. Baseline semi-supervised method results;
4. Preliminary results with Temporal Accumulation scheme;
5. Presentation and discussion of results using time-aware co-training method;
6. Conclusions and final remarks

4.5.1 Data and test setup

To show the potential of different semi-supervised learning methods, we selected the IDOL2 database [86].

From our previous experience in Chapter 3, the Spatial Pyramid Histogram (level 3) features are the single best performing visual features; this visual feature type is selected wherever we use single feature classification.

To give more insight about the methods, we follow the same testing setup as in Chapter 3 with 8 supervision levels with training, validation and testing patterns selected at random. In certain experiments we will explicitly mention and use three supervision levels:

1. Low supervision: 1% of labeling
2. Medium supervision: 10% of labeling
3. High supervision: 50% of labeling

Data pre-processing The data pre-processing step is used to yield compact image representations from high to very high low level visual descriptors. As described and motivated more in detail in Chapter 2, high dimensional representations are usually redundant and also pose a risk of over-fitting from statistical learning point of view. We will use the Kernel PCA [136] for dimensionality reduction with the use of χ^2 kernel [52]. The reduced dimensionality of the embeddings is set to 2'000 dimensions.

4.5.2 Study of confidence measures

Good quality confidence measure is essential for self-training and co-training methods to work. The quality of the confidence measures might be assessed in different ways. In the present study the following questions are answered:

1. Which confidence measure performs the best?

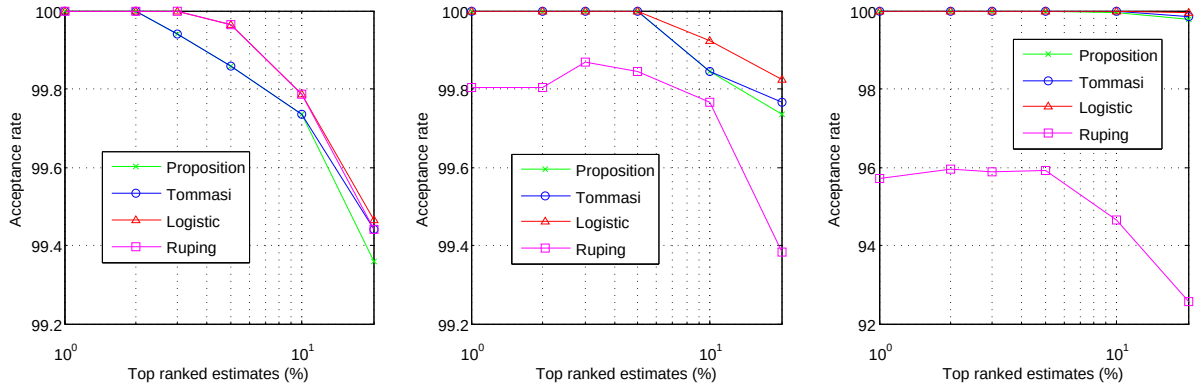


Figure 4.10: Precision of classification for the top confidence estimates: (left) 1%, (middle) 10% and (right) 50% of supervision

2. What is the impact of supervision level?

3. What kind of errors are the cause for erroneous top confidence estimations?

Two properties of the confidence measures derived from the SVM classifier: acceptance and rejection capabilities [3] are studied as well.

Acceptance: Top confidence estimates Intuitively, it is expected that the highest confidence estimates are correct with more errors as this measure gets lower. Therefore, one can compare different measures by evaluating the correctness of the p highest confidence estimates. In Fig. 4.10 the four confidence measures are compared at three supervision levels. The results show that for this data the proportion of the top confidence estimates increases as the amount of labeled patterns is raising from 1% to 50% of the total size of the database. The confidence measures perform very similarly with exception of Ruping measure which seem to degrade at higher supervision levels.

Erroneous estimations may be caused by different factors: poor discriminability of the descriptors, large class overlap, small training set and due to label noise. Each depicted frame in Fig. 4.11 show the latter type of misclassification: label noise or the class boundary problem. The problem cannot be resolved by a confidence measure alone, since visual content may not correspond to the labeling. This kind of error should be expected in all practical indexing setups.

This experiment allows to conclude that a certain amount of training data should be provided for the training of the visual appearance model for the score-based confidence measures to be reliable.

Rejection: Leave-one-out class A different aspect of a classifier is its rejection capabilities. Such situation may arise if the training set is of not sufficiently large or even missing some classes altogether. In practical applications it may be useful to put aside these patterns, for example leaving them for a post-processing stage or requiring explicit labeling by a human expert. This behavior requires abstention or no decision from a classifier [3].

Recall that we use c-class SVM classifier in one-vs-rest setup such that every unlabeled pattern $\mathbf{x}_i$ from class ω_k is expected to receive a score

$$s = \begin{cases} s_j^i > 0 & j = k \\ s_j^i \leq 0 & j \neq k \end{cases} \tag{4.31}$$



Figure 4.11: High confidence misclassification examples : (left) low, (middle) medium and (right) high supervision levels

If every class data has been presented for the training phase, in the ideal case only one binary classifier should return positive score. However, this is not the case if some class is too small or even missing during the training phase.

Suppose that the class ω_k is excluded, leaving the training set composed of data from $(c-1)$ classes. One cannot expect that the $(c-1)$ trained binary classifiers should all return negative scores for a pattern $\mathbf{x}_i \in \omega_k$ since the information about the class ω_k is missing in the model. The only available information are the scores returned by the $(c-1)$ binary classifiers.

In Fig. 4.12 an attempt to infer the rejection is made through the use of confidence measures. We compare the confidence measures returned for all samples of one class in two configurations: when the class was present or when it was missing during the training. In the experiment, every class has been excluded in turn and four confidence measures computed on the excluded class testing patterns. The final result is shown as a histogram of all confidence estimates over the 5 classes. Each row of the figure corresponds to a lowest supervision level to highest.

The results show that score-based confidence measures in general are not always a reliable estimate for the rejection. Simple use of the maximum score is not informative as for Logistic measure. Situation is different for the "Tommasi" and the proposed measures where relatively more rejected class patterns receive close to zero confidence. This can be explained by the usage of score contrast in the computation of the measure. If a pattern belongs to the excluded class, its scores tend to be more uniform over the binary decision functions and hence results in a small contrast and perhaps in random class overlaps. Enforcing the constraint of non overlap of the classes with the use of the proposed confidence measure, even more patterns from excluded class receive a close to zero confidence.

Comparison of the histogram reveals the problem of reliable rejection for small-sized training sets. As shown in top panel in Fig. 4.12, a high confidence measure ($z > 1$) can indicate reliable estimate. Rejection can be inferred for $z < 0.5$ using the "Tommasi" and the proposed measures.

4.5.3 Baseline semi-supervised method results

In this subsection we build the basis of the results obtained using four approaches for semi-supervised learning. Namely, label propagation, self-training, co-training and semi-supervised SVM (Laplacian regularized SVM). Our goal is two-folds: compare their best performances and point out their strong and weak sides. The comparison is always done with respect to baseline SVM. The parameters are estimated from the validation set as discussed earlier.

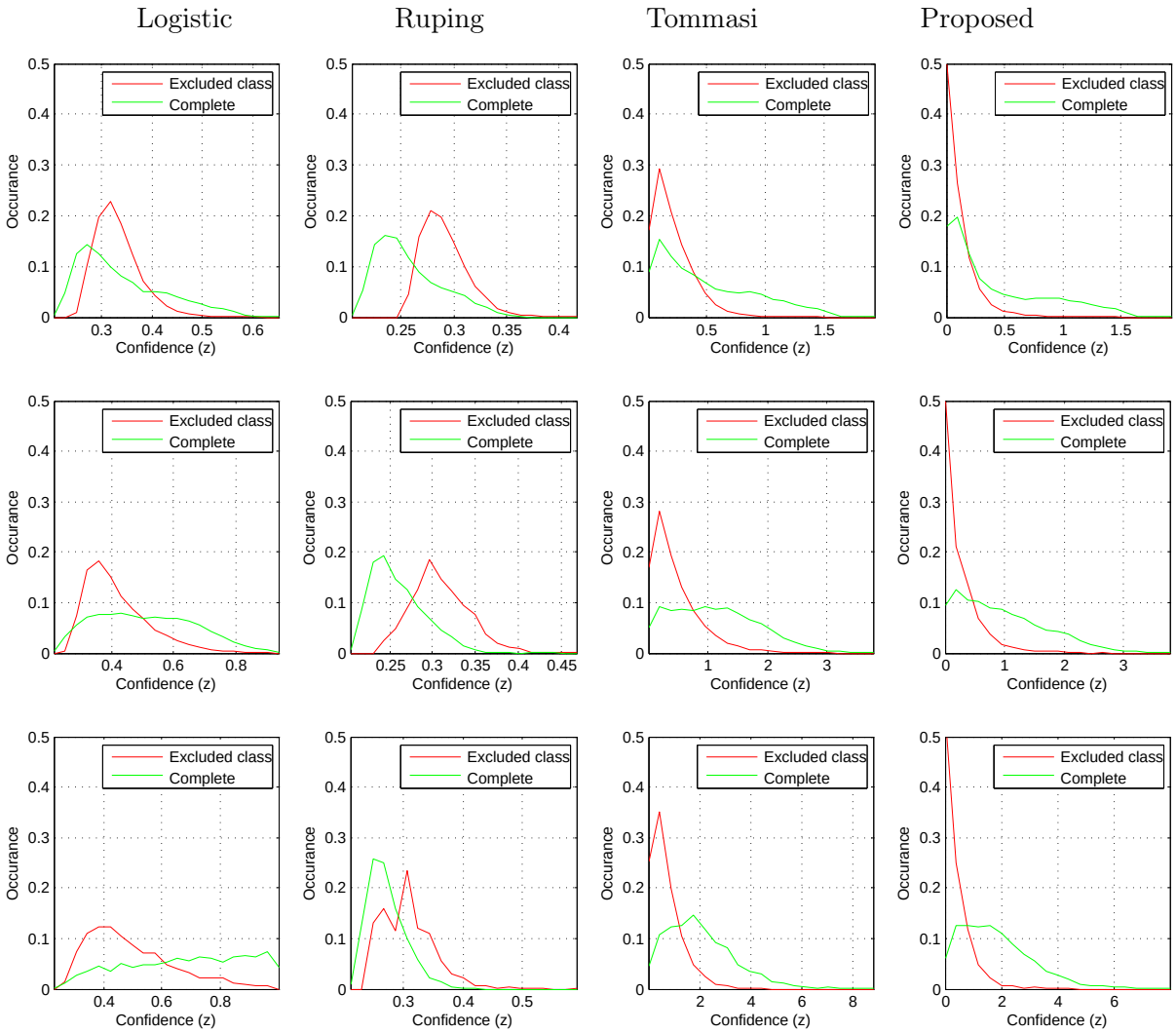


Figure 4.12: Confidence measure for rejection : rows (low, medium and high supervision), columns (Logistic, Ruping, Tommasi and the proposed class-overlap sensitive confidence measures)

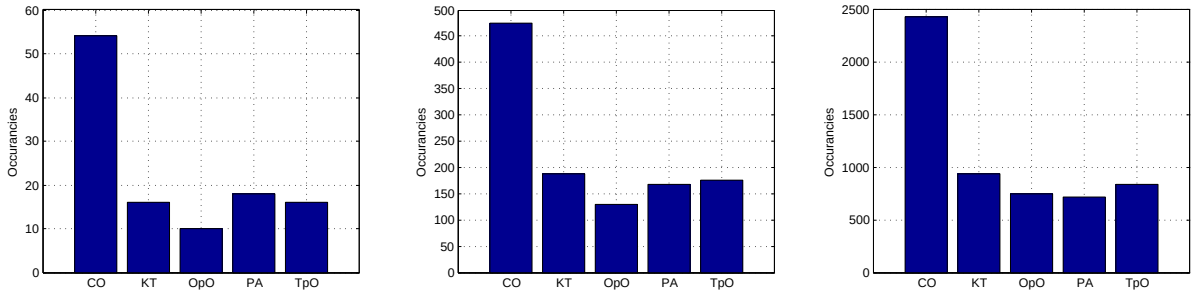


Figure 4.13: Class imbalance issue for Label Propagation method

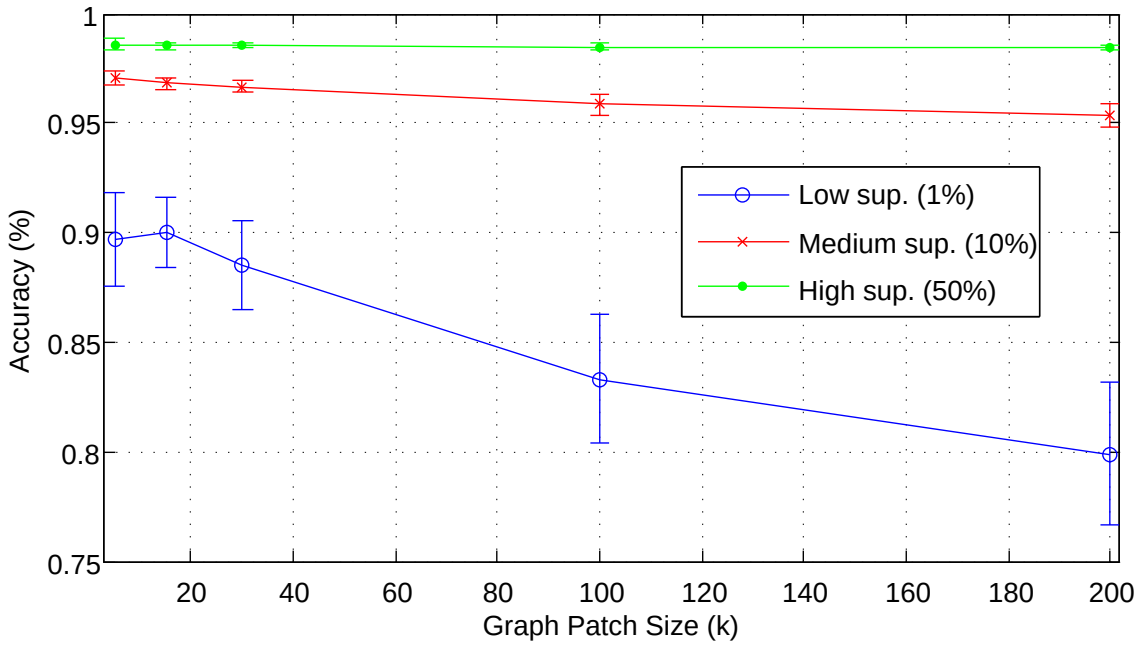


Figure 4.14: Label Propagation in a sparse patch graph

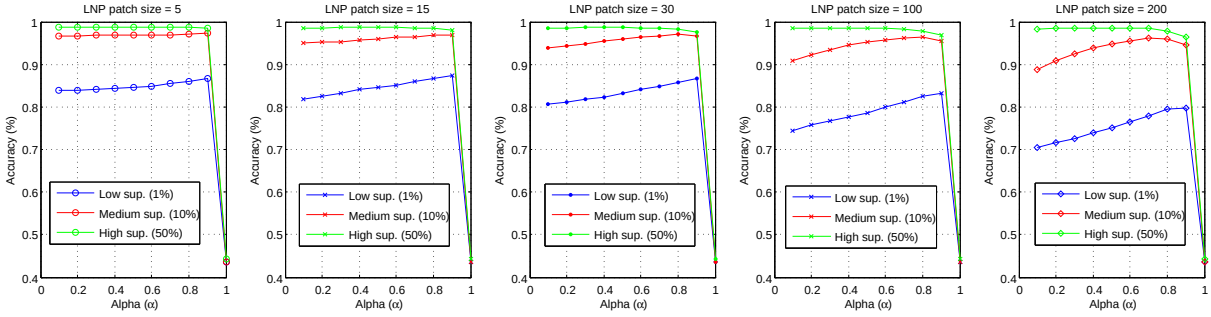


Figure 4.15: Label Propagation : influence of labeled nodes in label propagation process

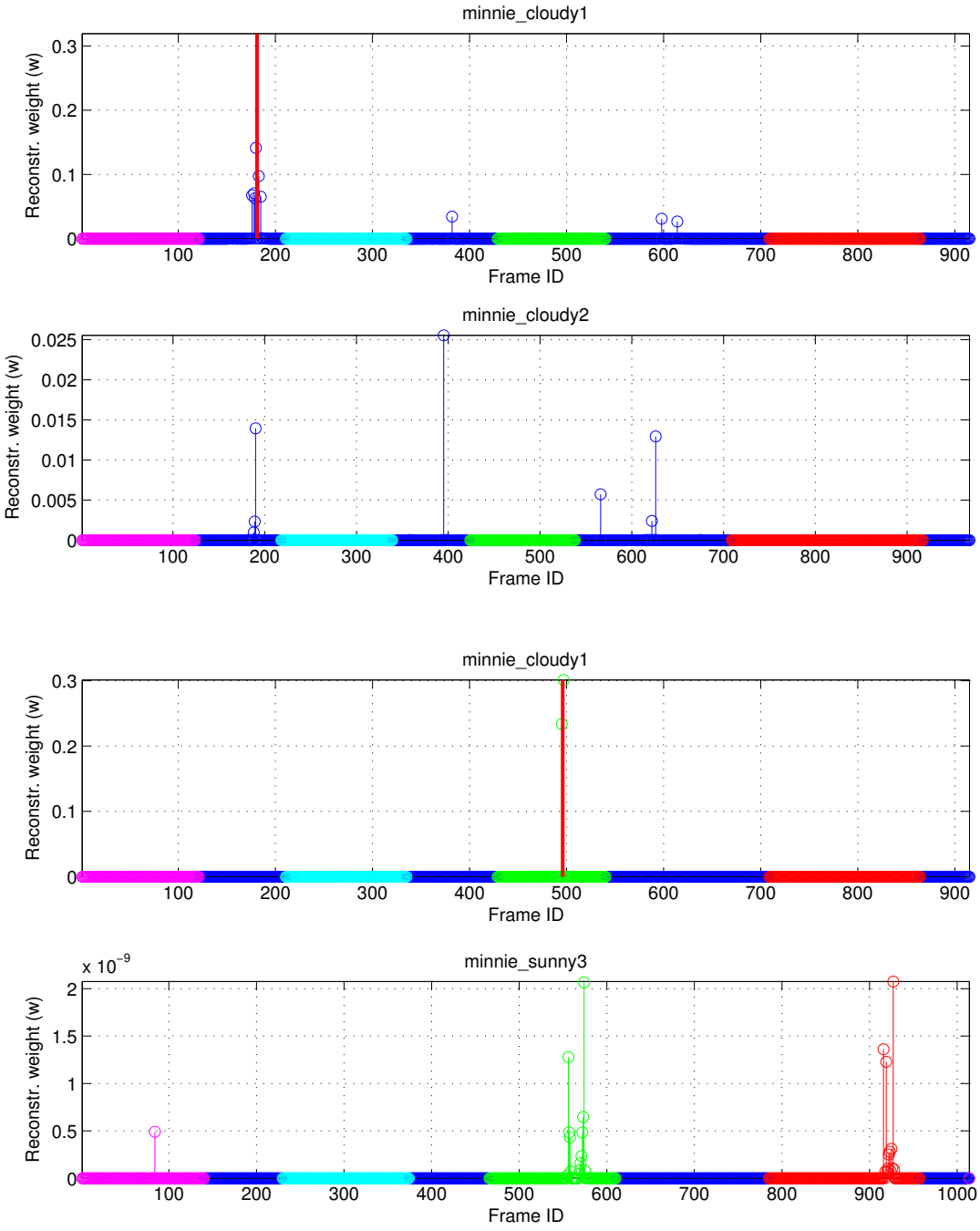


Figure 4.16: Label Propagation - Graph links found using the LNP algorithm (top) EASY: from cloudy1 to cloud2 (bottom) HARD: from cloudy1 to sunny3; Best viewed in color.

Label Propagation Label Propagation is an intuitive method in semi-supervised method family. We introduced the method in subsection 4.2.2 and apply it to our test database.

We show that the method can yield very competitive performances compared to other semi-supervised learning but is also very sensitive to the way how the graph is built and the choice of parameters.

Recall that in all graph-based methods (e.g. dimensionality reduction) we used a full graph capturing all possible affinities. The empirical results using such full graphs with Label Propagation showed a serious problem of class imbalance. Indeed, consider a fully connected graph and an image which is similar only to a subset of the database. It is clear, that contributions of irrelevant links, though expected to be weaker individually than for similar image links, will be overwhelming. In Fig. 4.13 we show relative labeled image proportions in three supervision scenarios. The empirical results with Label Propagation using this realistic scenario and a full graph confirmed the problem: all unlabeled images receive class "CO" label which is clearly incorrect. That is, the global accuracy of such classification resulted in probability of picking the largest class from the training set.

These results show that a choice of graph is critical compared to other graph-based learning methods. Motivated by temporal continuity of the video and assumptions of locality, we built a sparse graph (reviewed in subsection 4.2.2) using the locally linear patches method [163]. The graph construction is governed by a parameter k which is the size of neighborhood for each node. The classification results for different patch sizes and three supervision levels are shown in Fig. 4.14.

In attempt to visualize the graph connectivity properties for two pairs of videos, we show in Fig. 4.16 the discovered reconstruction weights in two typical cases : EASY (the links from "minnie_cloudy1" to "minnie_cloudy2") and HARD (the links from "minnie_cloudy1" to "minnie_sunny3") respectively. The colors on abscissa axis represent some class frames and larger than zero stem elements as positive matches between the two video sequences. We observed empirically that for two same light and close time video sequences (top pair), the links are highly consistent and almost never link different class images of the two videos. Moreover, occasionally more potentially useful links in the same class segments are found that is clearly advantageous for label propagation algorithm. Different situation is revealed for different light and distant video sequences. Often there are almost no links from video 1 to video 2 and occasionally erroneous links are found that is clearly harmful for label propagation algorithms. These considerations can explain relatively high classification rate for EASY cases and considerably lower performance for HARD cases.

Clearly, a properly built graph yields in very good classification performance. Keeping in mind that video labeling has been done in a sparse manner and that graph approximation with small local patches results in a sparse graph, the results are not very surprising. Even for as low as 1% of sparse video annotation resulted in 90% of correct class estimations on the rest of video. We outline the importance of the size of the local patch which seems to be more important for low supervision levels. For supervision of 10% and up to 50%, the classification results appears to be insensitive to the size of the local patch.

Finally, we study the impact of Label Propagation method parameter α that controls amount of label information that a particular node receives from its neighbors. In Fig. 4.15 we show the global classification accuracy on validation data with respect to the parameter. The impact of the patch size is also shown. We see that the parameter has little influence for small patch sizes and gets more important as its size gets larger (e.g. $k = 200$) and supervision level is lower. Remarkably, in most cases the best selection of the parameter is in the region $\alpha \in [0.8; 0.9]$. Recall that for

1. α is close to 1, label propagation is mostly governed by the propagation process by taking into account the graph globally; the influence of labeled nodes is small;
   Therefore, if the graph is correctly built, accent put with a large α will ensure correct label

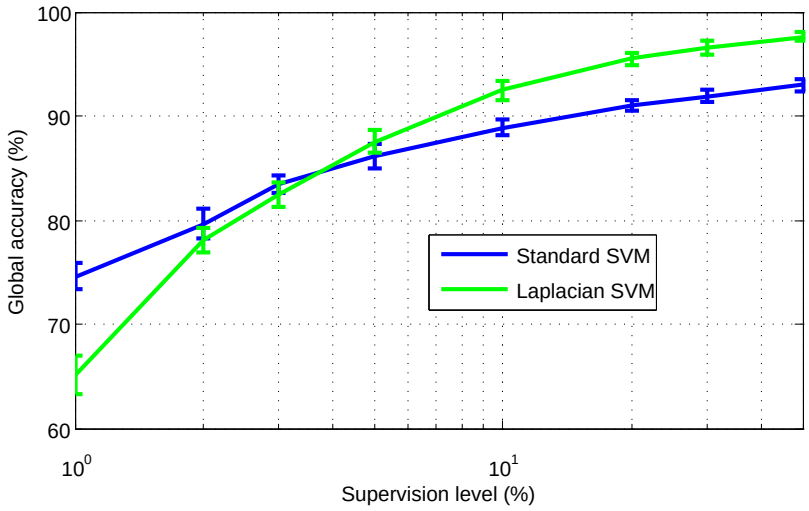


Figure 4.17: Semi-Supervised learning Laplacian SVM

propagation even with few labeled nodes.

2. α is close to 0, large weight is given to the initially labeled nodes and the global graph structure; In this opposite case, more accent is put on the labeled nodes of the graph. The labels are aggressively propagated to their neighbors which may pose a risk in the case of class imbalance and noisy labels.

For the data at hand, we note that a large value for α performs the best (selecting $\alpha = 1$ leads to a degenerate situation where no labeled information is taken into account). Thus, it indicates that the graph is modelling well the data and that relatively few labels are needed for mostly correct classification. We stress the point that for different labeling scenarios (for example, one labeled and one unlabeled video sequence), the selected graph construction method may not be optimal which may be due to problematic inter-video linking issues as depicted in Fig. 4.16.

Semi-Supervised SVM (Laplacian SVM) Semi-supervised SVM is an extension of standard supervised SVM and is reviewed in subsection 4.2.3. The idea is to leverage the unlabeled part of the data set in order to learn a better discriminative large margin classifier. In this experiment we compare the baseline soft-margin SVM and Semi-Supervised Laplacian SVM [91] classifier performances on image data. The goal is two-folds since we attempt to answer two questions:

1. Can we learn a better large margin classifier taking into account the unlabeled information?

2. Can we obtain performance increase at low supervision levels?

We summarize the classification results in Fig. 4.17 comparing standard supervised SVM and semi-supervised SVM. Cross-validation was used at each supervision level separately to obtain the best parameters.

The comparison reveals that at very low supervision levels (less than 3-5%) the semi-supervised SVM performs worse than standard SVM while substantial gain is achieved for higher than 5%-10% of supervision. The performance decrease can be explained by tedious parameter selection and the selected kernel function. Due to very costly cross-validation procedure for Laplacian SVM, we fixed

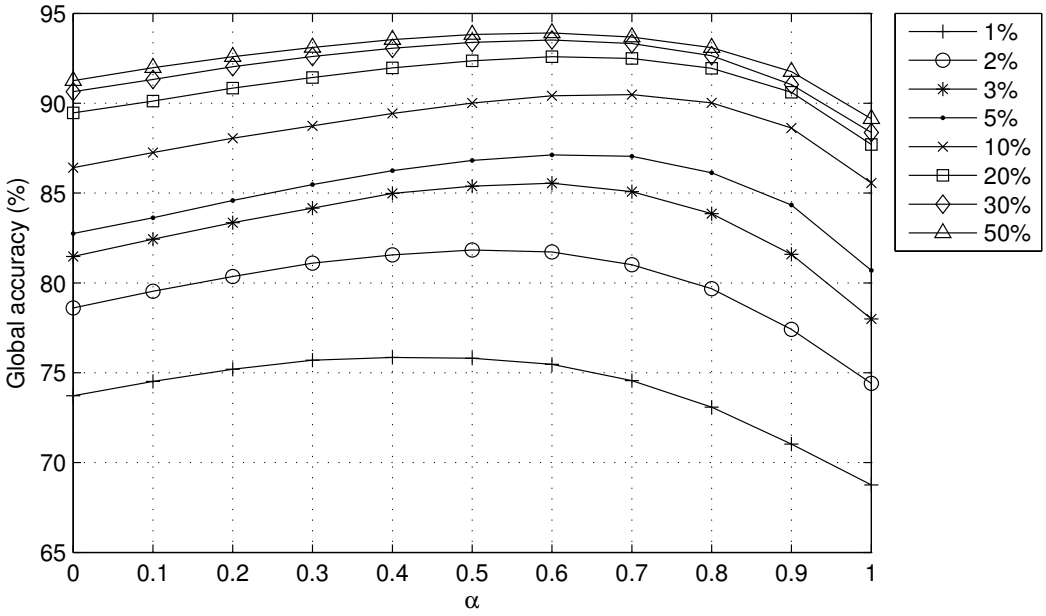


Figure 4.18: Effect of selecting fusion parameter α for DAS fusion

the ambient regularization parameter $\gamma_A = 0.01$, used very restricted list of intrinsic regularization parameters γ_I and finally used RBF kernel with its parameter σ set using Gehler's heuristic [54].

We conclude that for image classification task the usage of semi-supervised SVM can be useful to learn a better discriminative separating margin while practical difficulties of setting multiple parameters may be encountered, especially for larger data sets. Our practical results hint that parameter setting is more sensitive, and thus expensive, for low and very low supervision scenarios.

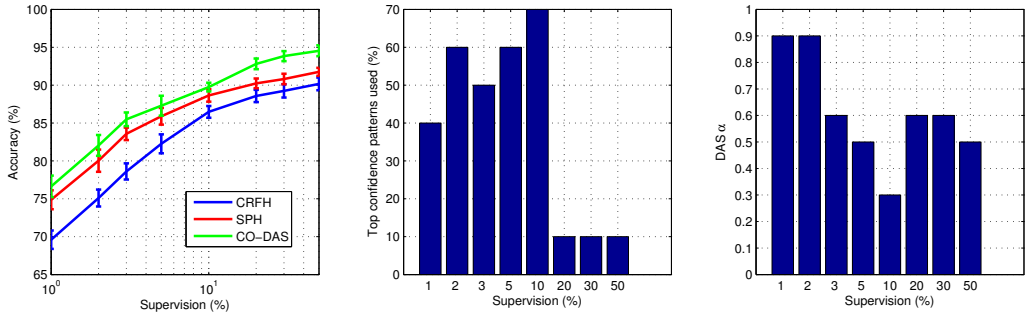


Figure 4.19: CO-DAS: (left) comparing testing performance for baseline CRFH and SPH with CO-DAS; (middle) cross-validation procedure selected amount of top confidence patterns to be added; (right) cross-validation procedure selected DAS mixing parameter α

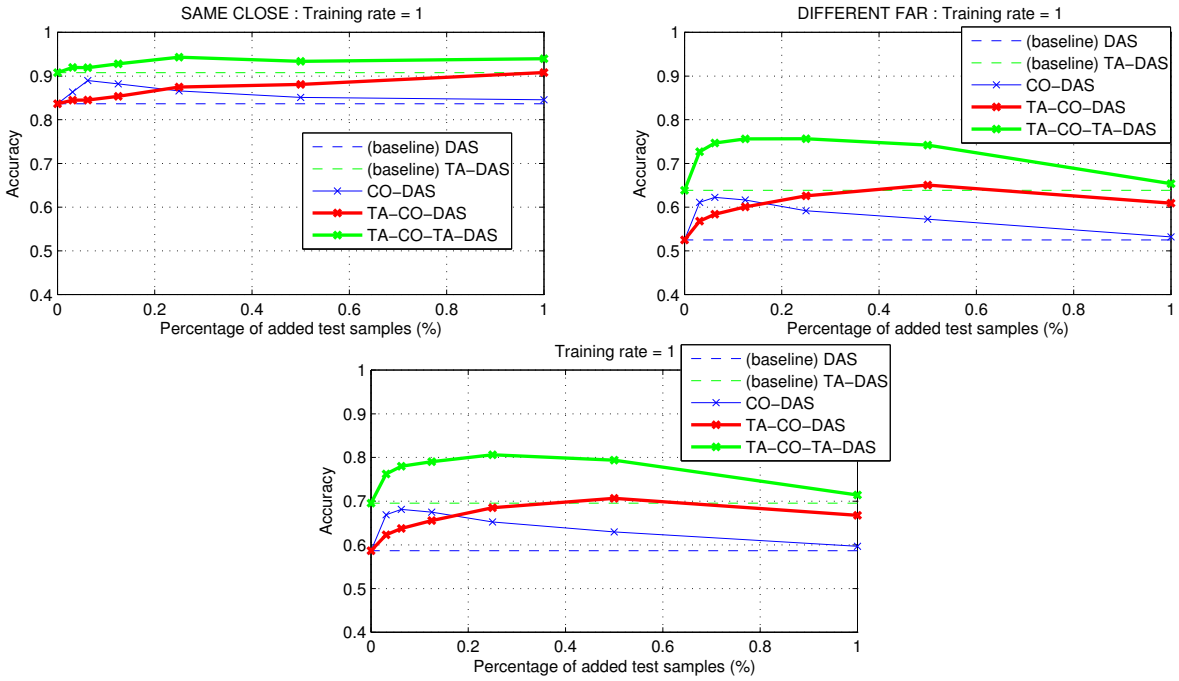


Figure 4.20: Summary of time-aware co-training methods: (left) same light, close time, (center) different light, close time, (right) all pairs of videos

4.5.4 Presentation and discussion of results using time-aware co-training method

In this subsection we study several variants of multiple cue learning methods on the base of co-training framework. Contrary to previous labeling scenarios, we test all possible video pairs (labeled and unlabeled video sequence) and report the averaged results. These results are summarized in Fig. 4.20 and show the performance of respective baselines in comparison to co-training enabled methods. The global accuracy measure is plotted versus percentage of added top confidence patterns in one iteration of feedback in co-training.

We start with simple methods and build right up to the complete multiple cue fusion method in the following order:

1. High level fusion, Co-Training disabled learning
   We consider two methods: DAS and TA-DAS.

2. Co-Training enabled learning
   We consider one methods: CO-DAS

3. Co-Training enabled with temporal information injection
   We consider two methods: TA-CO-DAS and TA-CO-TA-DAS

Co-Training disabled learning Standard DAS and temporal accumulation enabled DAS methods are the two baselines which are shown in Fig. 4.20. Performance of standard DAS performs is rather insensitive to mixing parameter selection. If two distinct visual features are relevant for classification then their linear combination performs at least as good as the weakest classifier. See left panel of Fig. 4.18 depicting influence of the mixing parameter at different supervision levels.

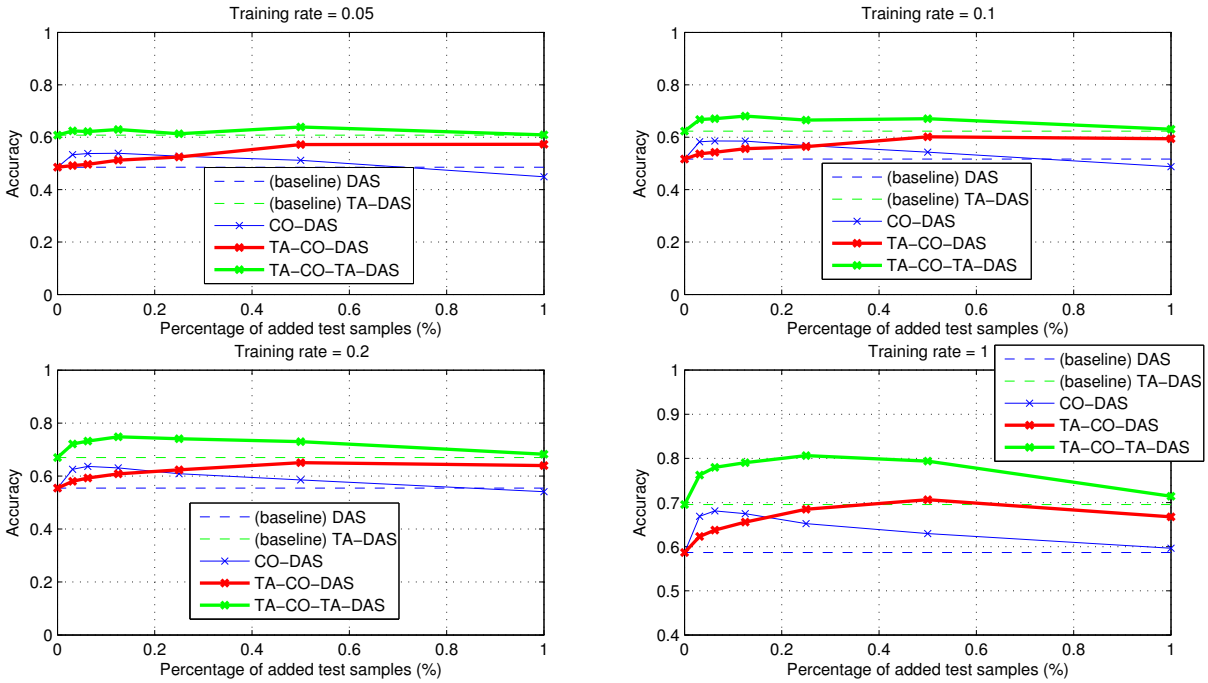


Figure 4.21: Performance of the proposed methods in low supervision scenario: (a) 5%, (b) 10%, (c) 20% and (d) 100% annotation of the training video

The DAS scheme can be boosted using the temporal accumulation scheme (TA-DAS) and experimental results confirm this. We argue that this is possible thanks to the smoothing procedure which eliminates random score error and finally a proper selection of the mixing parameter allows to profit from two visual cues using the DAS fusion module.

Co-Training enabled learning We believe that injection of top confidence estimations of one classifier into the training set of another one (one iteration of co-training feedback loop) may boost the performance. Indeed, empirical results show that relatively small amount (see Fig. 4.20) of top confidence estimates can boost the performance in one co-training loop.

Our experiments showed that often the overall performance increases over the baseline DAS method within the few very first iterations, afterwards it starts to decrease slowly towards the baseline. This behavior only reflects the overall quality of the confidence measure - it is capable to correctly classify the initial top confidence patterns, then more errors are introduced and the overall performance suffers.

In Fig. 4.19 we present the results obtained using CO-DAS method in the random sampling scenario of labeling. We performed only one co-training iteration in order to portray the efficiency of one feedback loop. Together with global accuracy measure of classification (left panel), the best parameters selected by the cross-validation procedure are reported as well (middle and right panels). In first place, we note an overall classification performance improvement at all supervision levels with improvement being the largest at higher supervision levels. Analysis of the best validation performance parameters reveal that relatively large amount of top confidence patterns are added to the training set. As well, notice the large DAS method α values which reflects the fact that at lower supervision levels the best performing single feature (SPH features) should be attributed the largest weight. Finally, notice the very low amounts of top confidence patterns selected to be added at high supervision levels by cross-validation procedure.

Co-Training enabled learning with temporal information injection One step further to achieve higher accuracy levels can be inclusion of temporal structure of the video which has not been taken into account. Empirical results using direct application of temporal accumulation scheme show that substantial performance gain can be obtained.

Motivated by success of the co-training method, an attentive reader may notice that the top confident patterns can be detected all across the video. At this point, the temporal accumulation scheme can be applied and temporal neighbors of the high confidence estimations can be included into the training set.

The idea to use co-training to detect high confidence estimates and then feed temporal accumulation result can be implemented in various ways. Immediately it may not be clear in which part of the processing chain the temporal accumulation should be performed. We evaluate two possibilities of inclusion of temporal information:

1. within the co-training feedback loop;

2. within the co-training feedback loop and prior classifier fusion using DAS module;

The classification results for all conditions show remarkable performance gain for both methods in comparison to baseline DAS, TA-DAS and also CO-DAS. Notice also the striking difference between both time-aware co-training methods. The results for the top performing method TA-CO-TA-DAS hints that temporal accumulation should be done both in co-training feedback loop and also prior classification fusion.

Low supervision scenario In real world applications it is not rare to possess a very limited amount of training data. To assess the performance of the proposed methods in the case of low amounts of annotation, we artificially lower the amount of provided ground truth from 100% to 20%, 10% and 5% percent per video sequence respectively while staying in the video versus video test setup. The average amount of frames in one video sequence is 945 frames so the respective amounts of labeled images is in average 945, 189, 95 and 47 images respectively. The amount of labeled images per class would be then even lower by roughly dividing these figures by the number of classes, which is 5 for this database.

In Fig. 4.21 the performance obtained on all 132 pairs of videos of the proposed methods and their respective baselines is shown. At 5% supervision we see that the time-aware co-training method does not provide any significant performance increase neither it harms the final performance performance. The major performance increase has bee achieved using temporal accumulation scheme only. As only one co-training feedback loop was performed and in the light of very low supervision, the lack of significant performance increase is not surprising. Qualitatively different picture is obtained when increasing supervision level up to 10% and 20% respectively. There the power of temporal accumulation scheme combined with the co-training method begins to bring the fruits. Finally, at 100% annotated training video we obtain a clear demonstration of leveraging both temporal structure of the video and confident estimations from the testing video to boost the final recognition performance. Notice that for most supervision levels cases around 20% of the test video images receive correct high confidence estimations and provide a consistent performance improvement. This was made possible by temporal accumulation scheme effectively removing occasional misclassifications and by the capacity of the proposed confidence measure to detect the high confidence test patterns to improve the quality of learned visual appearance model for location estimation.

4.5.5 Conclusion

In this chapter a time-aware semi-supervised learning method for image-based localization was proposed. The developed method was shown to be an attractive alternative to state-of-the-art semi-supervised learning methods such as Label Propagation in a graphs, Semi-Supervised SVM and standard co-training for video content indexing with respect to topological localization problem. We argue the interest of using the method for topological localization from the video based on the use of unlabeled information during the appearance model learning, temporal structure of the video as well as advanced discriminability power of the usage of two complementary global descriptors. Evaluated on the controlled environment database, these considerations explain the superior recognition performance of the method compared to the baselines as well as more advanced methods which we evaluated in the current study.

We would like also to outline the modular nature of the method which consists of diverse standard building blocks. New building blocks such as novel visual features, dimensionality reduction, classifier or any other processing module can be easily integrated. This modularity leaves room for future extensions and improvements.

Finally, the method features far more efficiency compared to the state-of-the-art methods for semi-supervised classification. Computation and especially memory demands are much lower when working on large-scale databases. This matter will be considered in Chapter 6, where evaluation of the method is conducted on a large corpus of videos.

Chapter 5

Invariant Visual Features for Image-Based Localization

Contents

5.1 Introduction **92**
5.2 Literature Review **92**
5.3 Prior information **94**
5.4 Invariant Support Vector Machine Formulation **95**
5.5 Experimental Results **97**
5.6 Conclusion **99**

5.1 Introduction

For visual object and scene recognition multiple visual features were developed. One of state-of-the-art features for scene recognition uses spatial pyramid representation initially proposed in [77]. Our main contribution of this chapter is to render these visual features invariant to spatial translation and evaluate them in our standard benchmarks. The objective is to evaluate if such descriptors can be improved with the goal of making them more appropriate for low supervision scenarios.

The chapter is organized in several sections. A brief literature review is given in Section 5.2 and followed by discussion on several state-of-the-art methods of interest in Section 5.3. In Section 5.4 we formalize supervised and semi-supervised SVM approaches and conclude with experimental results in Section 5.5.

5.2 Literature Review

5.2.1 Translation invariant Spatial Pyramid Histograms

In this chapter we review state-of-the-art Spatial Pyramid Histograms [77] and contribute with novel derived visual features featuring spatial translation invariance.

Brief literature review Spatial Pyramid Histograms [77] has enjoyed particular interest from pattern recognition community. Created to remedy the lack of spatial information in Bag of Visual Words [37, 100, 134], the visual features resulted in a rich and discriminative source of information for scene and object classification. Numerous studies were then conducted to improve further the discriminability of the visual features. In [129] the more sophisticated image region division rules were studied and combined with the power of multi-resolution histograms. Spatial pyramid coding together with improved word weighting schemes was studied in [174]. Significantly more compact features, while being as performing as original Spatial Pyramid Histograms, were proposed in [70].

Review of Spatial Pyramid Histograms The features belong to a global image descriptor family. Spatial Pyramid Histograms are created over a grid of segments of an image at different scales and where the resulting Bag of Visual Words histograms are concatenated.

The power of the approach possibly lies in the fact that both local and global information is captured by the use of fine to coarse grid of regions. Order-less statistics of the region as histogram of local features effectively captures the information at particular scale and spatial locality.

The choice of the low level features can be arbitrary and may be be as well as simple intensity or gradient values and up to sophisticated local features as SIFT, SURF and others. In original paper [77] authors use densely sampled gradient magnitudes at two different scales and at 8 orientations as well as densely sampled SIFT over 16×16 size patches with 8 pixel spacing. This is done in order to capture also information from relatively homogeneous regions of an image.

Indeed, a study in [103] reveals that humans are able to recognize scenes not paying attention to details. Importance of both global and local information for scene recognition was discovered in [161].

Perspectives of improvement Numerous evaluation of the Pyramid Histogram visual features proved the capacity to capture relevant low level visual information on challenging databases as on Caltech-101 in [77, 129], Caltech-256 in [129], on 15-scene in [174] and many others. In previous chapters we evaluated a selection of visual features among which this feature type was found to be the top performing single visual feature.

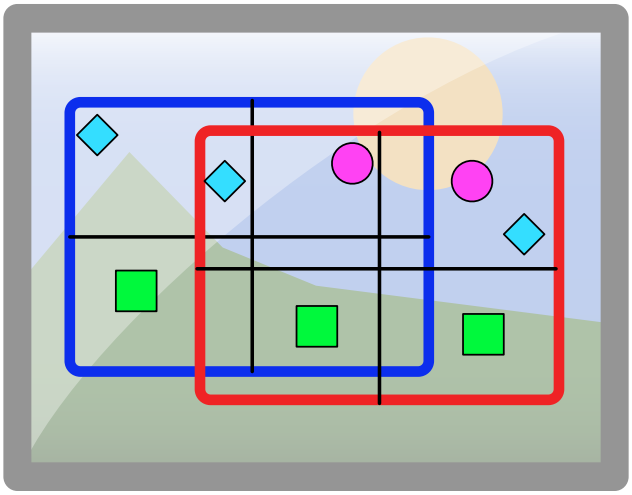


Figure 5.1: Issue with translated region of interest (translation occurred from red to blue region of interest)

Nevertheless, in the context of video indexing we are constantly facing the problem of the lack of training data which lead us to explore various complex semi-supervised learning techniques. Recall that the displacement of a wearable video camera are basically of horizontal nature when the person turns, therefore we propose to provide a translation accounting version of Spatial Pyramid Histograms which may reduce excessively large variability of the low level features. Indeed, suppose that a small horizontal translation occurs as portrayed in Fig. 5.1. It is evident that some regions are shifted with respect to the initial placement of the region of interest and result in different representations of basically same visual content scene. The effect may be more pronounced at finer grid sampling as some regions may end up in different parts of the concatenated histogram.

In this chapter we propose to exploit the kernel learning framework to achieve a certain level of invariance with respect to such translations.

5.2.2 Methods and applications incorporating invariance

Brief literature review Practical applications of invariance enforcement were found for EEG signal reading classification [107, 111] where imaginary movements of the left and right hands are supposed to be detected. Scaling and rotation invariance [108] is useful for face recognition due to large variability of the visual appearance, light conditions and in the presence of noise. Optical Character Recognition (OCR) is one of applications enjoying particular attention because of high reliability and robustness requirements in practical applications. The performances of multiple methods incorporating invariances for OCR [109, 57, 26, 30, 81, 58] showed promising results. A challenging problem is that of object classification [110, 53] where local transformation invariant descriptors provided an interesting gain of performance using tangent approximation.

Common methods enforcing invariance The transformation invariance can reduce excessively variable input space and therefore render learning of the separating hyperplane in the context of SVM an easier task.

Following [125], there are three large groups of methods using transformation invariance in the context of SVM:

1. Virtual examples [101, 124]
2. Jittered samples [39]
3. Building transformation invariant kernel [110, 111, 131, 26]

The first method augments the training set with artificially crafted patterns in a hope that transformation invariance will be taken into account. In the context of SVM, support vectors patterns are considered and then new virtual support vectors are obtained after a certain transformation. The intuition is to expect the new virtual patterns to become support vectors after another iteration of learning. If so, the decision hyperplane may take the desired shape. Despite its ease of interpretation, this approach may not be appropriate in practical applications due to its increasingly growing computational demands.

The jittered sample method is similar to virtual support vector creation method. In this method the transformations take place at kernel level by moving the inputs of the kernel function using some transformation. The best match is kept. The advantage over virtual support vector method is that training time can be in some cases considerably reduced. A disadvantage may be noticed in the testing phase, since the jittering should be carried out here too.

The last method uses an elegant formulation such that a change of representation or a mapping to a more suitable space of the data yields invariance properties. Remarkably, the transformation invariance enforced using this method will be seen as a design of a new kernel without changing the internals of the underlying learning machine. In this chapter we consider the methods from this group.

See [106] and [76] for a more complete and detailed review of the methods and related works incorporating invariance.

5.3 Prior information

Prior information in SVM framework Lets introduce the invariance property in general terms in the context of Support Vector Machines (SVM). Recall that the solution to the regularized learning problem in the context of linear kernel SVM is a function

$$f(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \tag{5.1}$$

or equivalently

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b \tag{5.2}$$

using separating hyperplane notion where $\mathbf{x}, \mathbf{x}_i, \mathbf{w} \in \mathbb{R}^d$ and $\alpha_i, b \in \mathbb{R}$. As previously, we call the output $f(\mathbf{x}) \in \mathbb{R}$ as decision value.

Invariance property can be seen as the invariance of output $f(\mathbf{x})$ with respect to a small perturbation of a pattern $\mathbf{x}$ along a specific path. In more general terms, the decision value can remain

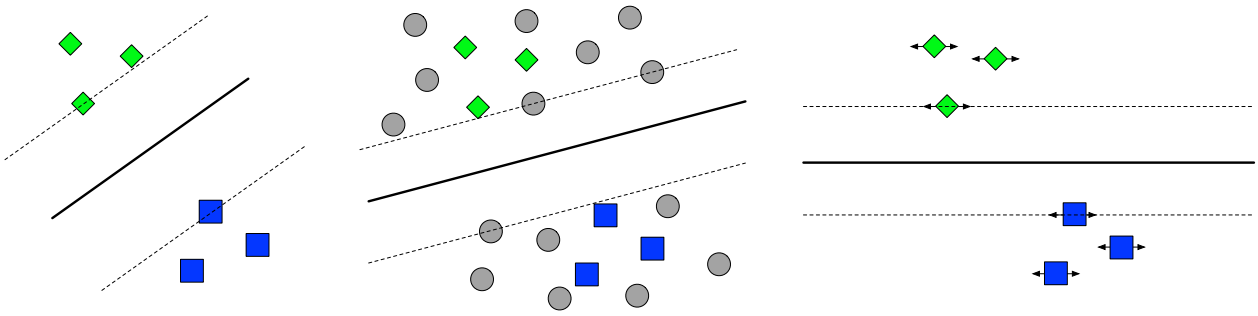


Figure 5.2: Illustrating invariance : (left) standard SVM, (middle) Semi-Supervised SVM, (right) Invariant SVM

unchanged when the pattern is transformed by some operator $\mathcal{L}$. The transformation can be governed by several local transformations $v = 1, \ldots, p$ giving a rise to an operator $\mathcal{L}_v$. In the following discussion we will use 1-parameter operator which we denote as $\mathcal{L}_t$ corresponding to horizontal displacement.

Graphical intuition The property of invariance can be useful for the classification task in the context of SVM. Let us consider a simple classification problem depicted in Fig. 5.2.

The problem consists of two distinct class patterns and the goal is to find the simple model corresponding to Eq. 5.1. In standard supervised SVM setup the separating hyperplane can be drawn midway between two points as depicted in left panel. Notice that without additional data patterns or some other prior information about the data, there is no reason to prefer another hyperplane which does not maximize the margin.

Given also some unlabeled data, one can resort to semi-supervised SVM which can find a margin passing through low density regions. Notice the change of the hyperplane and the margin around it from the standard supervised SVM.

Finally, suppose that there exist a tangent subspace around each training patterns such that the decision value is locally invariant on that manifold.

This intuitive analysis shows that a better model can be learned with less training data if a prior knowledge is leveraged properly.

5.4 Invariant Support Vector Machine Formulation

Consider a standard classification problem with a training set $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and unlabeled set $U = \{\mathbf{x}_i\}_{i=l+1}^{l+u}$ where $\mathbf{x} \in \mathbb{R}^d$ and $y \in \{-1, +1\}$. Given a valid kernel function

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle \tag{5.3}$$

a non-linear mapping function $\Phi : \mathbb{R} \to \mathcal{H}$ is implicitly induced in RKHS (See Chapter 2 for a review and references). In the case of SVM, the decision function can be written as follows

$$f(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b \tag{5.4}$$

where the real-valued coefficients $\{\alpha_i\}_{i=1}^{l}$ and $b \in v\mathbb{R}$ are found solving the Quadratic Program

$$J(\boldsymbol{\alpha}) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{5.5}$$

subject to the constraints $\sum_{i=1}^{l} \alpha_i y_i = 0$ and $\alpha_i = 0, i = 1, \ldots, l$. See Chapter 2 for a review on SVM classifier.

5.4.1 Incorporating invariance

As suggested early in [155] and more recently in [26], we consider that the local transformation operator $\mathcal{L}_t$ applied on the data should not affect the class label.

Suppose that a pattern $\mathbf{z}$ is locally transformed in $\mathcal{L}_t\mathbf{z}$ by some pre-defined 1-parameter transformation t. Then we can assume that a tangent vector

$$\delta\mathbf{z}_i = \lim_{t \to 0} \frac{1}{t} (\mathcal{L}_t\mathbf{z}_i - \mathbf{z}_i) = \left. \frac{\partial}{\partial t} \right|_{t=0} \mathcal{L}_t\mathbf{z}_i \tag{5.6}$$

associated to the pattern $\mathbf{x}_i$ defines a direction in the input space along which the decision value ideally should not change. Following [26], we can therefore require that the maximum margin separating hyperplane $\mathbf{w} \in \mathbb{R}^d$ to be orthogonal to the tangent vectors in Eq. 5.6. An important point is that the possible margins which cross the tangent vectors are penalized therefore resulting in a restricted space of solutions in the form of Eq. 5.4.

This leads to a regularized formulation of the standard hard-margin SVM

$$\arg\min_{\mathbf{w}} (1 - \gamma) \|\mathbf{w}\|^2 + \gamma \sum_{i=1}^{l} \langle \mathbf{w}, \delta\mathbf{z}_i \rangle^2 \tag{5.7}$$

subject to the constraints $y_i (\langle \mathbf{w}, \mathbf{z}_i \rangle + b) \geq 1$ and where the parameter $0 \leq \gamma < 1$ governs the amount of invariance incorporated in the solution. Notice that for $\gamma = 0$ a standard hard-margin SVM is obtained.

Following [26], a regularized covariance matrix is computed from all tangent vectors

$$S_\gamma = (1 - \gamma) I + \gamma \sum_{i=1}^{n} \delta\mathbf{z}_i \cdot \delta\mathbf{z}_i^T \tag{5.8}$$

can be used to transform the patterns

$$\mathbf{z}_i^\gamma = S_\gamma^{-\frac{1}{2}} \mathbf{z}_i \tag{5.9}$$

and then used in the standard SVM classifier. Hence, incorporating invariance using tangent vector approach can be as well seen as a pre-processing step.

5.4.2 Linearization of the input space

The previous analysis is valid for linear feature space. As reviewed in Chapter 2, the non-linear nature of the input space can be accounted for by a appropriate change of representation. Assuming that the relevant dynamics of the data is described by its variance, then the change of representation can be done using PCA or its kernelized version Kernel PCA [125]. Idea is to project the data onto the maximum variance directions which is linear for PCA and non-linear depending on the choice of the kernel for Kernel PCA.

After the computation of perturbed patterns, Out-of-Sample extension of the Kernel PCA embedding may be done on $\mathcal{L}_t\mathbf{z}$

$$\mathbf{z}_i = A_k^T \mathbf{K}(\mathbf{x}_i) \tag{5.10}$$

where $\mathbf{K}(\mathbf{x}_i) = [k(\mathbf{x}_i, \mathbf{x}_1), \ldots, k(\mathbf{x}_i, \mathbf{x}_n)]^T$ and A_k is a matrix with k largest eigenvector expansion coefficients (See Chapter 2).

Note that the proper kernel function should be used and that the projection operation can result in a loss of information.

5.4.3 Algorithmic perspective

In practice, little modifications to standard kernel learning methods are necessary to include described 1-parameter invariance. We consider here the non-linear SVM case.

Suppose that our goal is to enforce horizontal translation invariance for Spatial Pyramid Histogram [77] visual features (see section 5.2.1). The following steps can be taken:

1. Compute multi-level Spatial Histogram signatures $\mathbf{x}_\mathrm{L}$ and $\mathbf{x}_\mathrm{R}$ for two horizontally overlapping regions of an image
2. Build the χ^2 kernel Gram matrix K_LL from left region patterns $\mathbf{x}_\mathrm{L}$
3. Compute the m-dimensional embeddings $\mathbf{z}_\mathrm{LL}$ for Gram matrix K_LL
   (a) Compute m top eigenvalue $\{\lambda_i\}_{i=1}^m$ and eigenvectors $\{\mathbf{e}_i\}_{i=1}^m$ of the Gram matrix K_LL;
   (b) Populate the matrix A_k with k largest eigenvalue whitened eigenvectors $\tilde{\mathbf{e}}_i = \frac{\mathbf{e}_i}{\sqrt{\lambda_i}}$;
   (c) Construct embeddings matrix $Z_\mathrm{L} = A_k^T K_\mathrm{LL}$;
4. Use Out-of-Sample Extension to compute embeddings for right region patterns
   (a) Build the χ^2 kernel Gram matrix K_LR comparing left and right region patterns $\mathbf{x}_\mathrm{L}$ and $\mathbf{x}_\mathrm{R}$;
   (b) Construct embeddings matrix $Z_\mathrm{R} = A_k^T K_\mathrm{LR}$;
5. Compute embedding difference $\Delta Z = Z_\mathrm{R} - Z_\mathrm{L}$
6. Estimate sample covariance matrix $S = \Delta Z \cdot (\Delta Z)^T$
7. Inject invariance by setting $0 < \gamma \leq 1$ and building $R_\gamma = (1-\gamma) I + \gamma S$
8. Compute translation invariant embeddings $Z_\mathrm{TI} = R_\gamma^{-\frac{1}{2}} Z_\mathrm{L}$
9. Train and the standard SVM classifier on the translation invariant embeddings Z_TI

5.5 Experimental Results

In this section we present the experimental results for the IDOL2 database [86] using Spatial Pyramid Histogram visual features. The main goal is to assess the usefulness of the invariance enforcement and reveal the test cases when it is effective.

We emphasize the choice of horizontal translation invariance by the matter of fact that the principal movement of the robot platform is horizontal displacement.

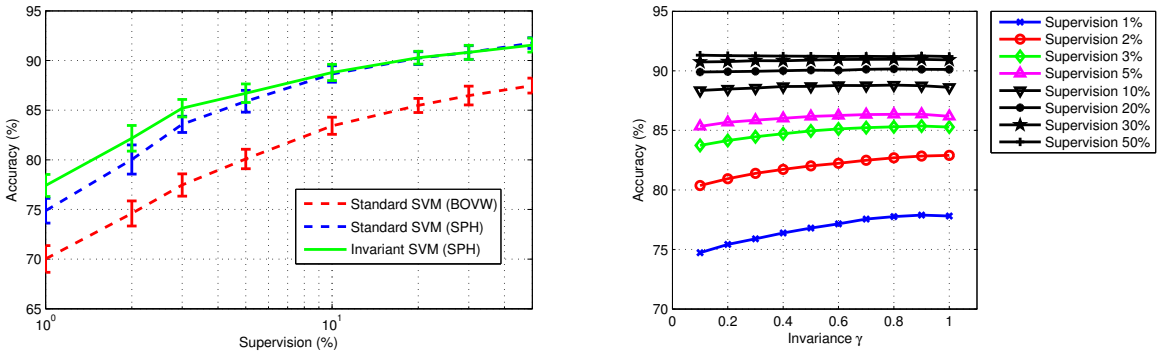


Figure 5.3: Random sampling scenario: (left) performance of standard and invariant SVM at varying supervision levels; (right) effect of the invariance regularization parameter

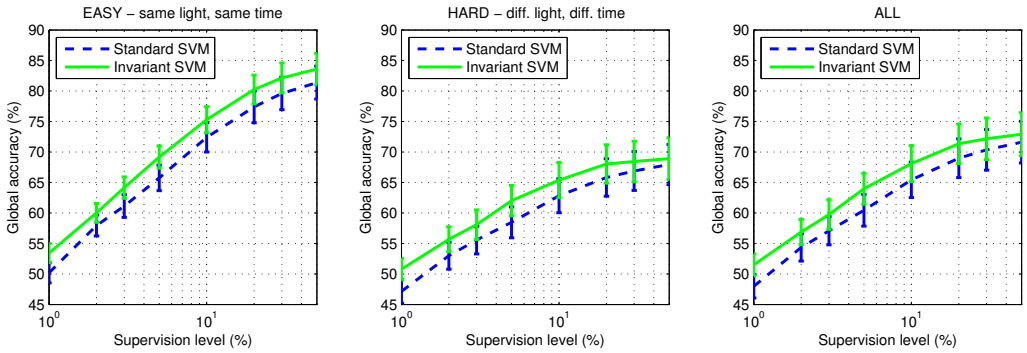


Figure 5.4: Video vs Video scenario: (left) same light, close time pairs; (middle) different light and time pairs; (right) all video pairs

Evaluation setup The comparison between standard soft-margin SVM and invariance enforced SVM is done using two annotation scenarios. In the first scenario we follow the random sampling strategy as in all previous experiments, while in the second video versus video strategy for all the possible pairs is employed. In the latter scenario, one video sequence is divided in training and validation parts, while the second video sequence is left for testing purposes only.

Discussion of results In this basic evaluation the main accent was put on validation of spatially translation invariant visual features in low supervision situations. Lets review the two labeling scenarios in turn.

The random labeling scenario has been widely used in the previous chapters and can be considered as a standard classification problem. In Fig. 5.3 in the left panel we compare classification accuracy of standard soft-margin SVM (on BOVW and SPH features) and invariance enabled SVM (SPH features) classifier results. We can notice the overall superiority of SPH over BOVW features which is clearly due to spatial information encapsulated in the former. Therefore in the following discussion we focus only on improvement of the SPH while the same invariance enforcement can be carried out also for the BOVW.

Both methods (standard and invariant SVM on SPH features) provide relatively good performance for as low as 1-5% of labeled data with invariant SVM being superior. However, we notice that there is a room of improvement since with 10%-20% of training data the classification accuracy achieves around 90% which is a good result. Spatial translation invariance is clearly useful at very low supervision levels. The effect diminishes as the training set grows in size past by 10-20% of the total size of the data set. These results suggest that it is possible to find a good separating hyperplane even at very low supervision levels given that the invariance property is enforced correctly.

A different aspect concerns the amount or the strength of invariance injected which in our exampled is controlled by increasing regularization parameter γ from 0 to 1. From the right panel of Fig. 5.3 it is clear that a stronger emphasis on invariance is necessary for small training sets which is around 1-5% of the data for our test example. Notice that no gain can be brought at higher supervision levels where apparently the training set is large enough to bring information about invariance properties useful for classification task. Authors in [107, 111] arrived at the same conclusions in face recognition task where scale and rotation invariance was incorporated.

For video-vs-video scenario we tested all possible pairs of the database consisting of 12 video sequences. The training set is sampled from the first video, while the testing set corresponds to the second video. We divided the video pairs in two large groups: same light and close time pairs, as well as different lighting condition and different time pairs. The summary of this evaluation is shown in Fig. 5.4 where the right figure is averaged performance over all pairs. First of all we can notice that the best performance is lower in comparison to the random labeling scenario. Secondly, notice in the middle panel of the figure the harmful impact of the light conditions and new visually different content. We observe a clear and constant improvement of invariant SVM at all supervision levels, lighting and new visual information introduced by a time span between some pairs of videos. All these observation in video versus video scenario allow us to observe a typical lack of training data and practical utility of translation invariant visual features.

5.6 Conclusion

In this chapter, a promising avenue of efficient exploitation of invariance properties and thus limiting excessive variability of the visual information has been discussed and evaluated for the task of image-based localization. We note a particular usefulness of invariance in the cases of small or insufficiently

representative training sets which is more likely to occur in real-world automatic indexing conditions.

Our demonstration on the publicly available data shows that the prior knowledge can be taken into account as a mere pre-processing step in the familiar framework of kernel learning. Therefore, the approach has an interesting modular nature that could be exploited to speed up to computation.

Chapter 6

Application to the IMMED Context

Contents

6.1 Introduction 102
6.2 Context 102
6.3 Description of the corpus 102
6.4 Experimental protocol 104
6.5 Evaluation on a test case 104
6.6 Computational Cost 107
6.7 Baseline image-based localization results 108
6.8 Conclusion 108

6.1 Introduction

In the previous chapters we evaluated our own methods for image-based localization compared to baseline and state-of-the-art methods on the IDOL2 [86] database. The database has been captured by a mobile robot platform in a well controlled environment which may not be always the case for the video captured by a conventional person. The goal of this chapter is to evaluate the best performing methods on a large corpus of video data captured in a realistic setup.

6.2 Context

Introducing the project With rapidly increasing proportion of elderly population in Europe, severe health-care problems emerged. In a near future a large percentage of the elderly people will be touched by the Alzheimer disease. Some estimates even claim that each tenth of the population can be concerned by the disease. The disease is supposedly best treated if detected in early stages. A recent study [40] claims that the first signs of the disease can be detected using MRI by measuring structural changes in the brain.

To answer this challenge, an unique project IMMED was launched in 2009 as a continuation of the PEPS project. The project occupies unique place among other projects solving the problem of aging population and enjoys particular interest in the sphere of research and development at national and European levels. The originality of the project lies in early detection of instrumental and memory disorders in ecological environments of potential patients without the need to install additional equipment and systems as in [99, 184].

Project tasks and the place of the study There are three main tasks of the project:

1. Specification and development of the wearable recording device;
2. Develop the algorithms for video content structuring by detecting the localization and the activities carried out by a person;
3. Develop efficient cross-media fusion algorithms for joint audio and visual information exploitation;

The place of the current work is in the task for topological localization indoors. The main efforts were focused to develop the algorithms capable to localize the potential patient using only visual information recorded by the wearable video camera. The computed localization output is then used as an input for action recognition framework being researched and developed by the project partners.

Wearable video camera acquisition device As a part of the project, a light and highly portable acquisition device prototype was developed. The prototype contains the lightweight GoPro video camera (See Fig. 6.1) to be worn by a potential patient in his or her casual environment at home. When attached to the person's shoulder, the prototype records the instrumental area in front of patient and the local context using a fish-eye High Definition video camera. A short technical specification of the video camera is given in Table 6.1.

6.3 Description of the corpus

After multiple recording sessions in realistic setup, a large corpus of 26 videos acquired using a wearable video camera worn by different patients and volunteers in their ecological environment has

Lens Type	Fixed focus, 0.6m - ∞
Aperture	f / 2.8
Angle of View	127° at 1080p, 170° at 960p and lower
Recording Resolution	960p - 1280 x 960 @ 30 fps
Sensor Type	1 / 2.5", HD CMOS
Light Sensitivity	$>$ 1.4 V / lux-sec
Exposure Control	automatic
Storage	SD card, up to 32 GB
Battery	Lithium-Ion, 1100 mAh
Size	4.2 x 6.2 x 3.0 cm
Weight	150 g

Table 6.1: Wearable video camera technical specifications



Figure 6.1: GoPro video camera (Half Moon Bay Company, California, Woodman Labs)

been created, starting in 2011. The corpus consists of 24 sequences sets representing each one house of a patient or a volunteer and contains typically two video sequences: a short annotated bootstrap and a longer unlabeled video with actual activities.

Experience with this challenging corpus of videos is presented in two parts. In Section 6.5 we evaluate and analyze the performance of the methods used in previous chapters on a selected pair of bootstrap and unlabeled video. In Section 6.7 we present the results on the rest of the corpus by evaluating of single feature approaches.

In Annex E more information about the corpus is provided.

Topological locations The ground truth of the corpus has been created manually and uses an uniform list of possible topological locations. Of course, there can be a larger diversity of the classes but we limit ourselves to the most common locations found in a typical house and use these class names while annotating all the videos in the corpus. In Table 6.2 six topological location names are

"bathroom"	"bedroom"	"kitchen"	"living_room"	"outside"	"other"
Hygiene, teeth brushing	Sleep, Rest	Food preparation, dish washing, complex machines	Eating, watching TV, reading	Outdoors	Other locations indoors

Table 6.2: Uniform topological location names

given as well as short description for each of them.

6.4 Experimental protocol

The whole processing pipeline consists of two visual feature BOVW and SPH descriptor extraction, dimensionality reduction and followed by a classification method. Five classification methods are evaluated:

1. Standard single feature SVM (SVM)
2. Temporal Accumulation followed by SVM (TA-SVM)
3. Temporal Accumulation for two features followed by DAS fusion (TA-DAS)
4. Standard Co-Training followed by DAS fusion (CO-DAS)
5. Time-Aware Co-Training with DAS fusion (TA-CO-TA-DAS)

The Co-Training method is a method that can re-define its learned model iteratively after each step. In all experiments we use only one step of Co-Training to ease the evaluation and comparison to other methods. One step is defined as an addition of certain percentage of top confidence estimates from the unlabeled set into the training set. The percentage measures as a portion of the total size of the current testing set (top confidence estimates from the testing set are removed after being added to the training set).

The performance of a method is measured using global accuracy metric.

6.4.1 Visual features

Low level visual information from image data has been extracted using two methods: Bag of Visual Words (BOVW) [37, 100, 134] and Spatial Pyramid Histograms (SPH) [77]. Dimensionality of these global descriptors are 1111 and 4200 dimensions respectively, and were reduced to a maximum of 500 dimensions using Kernel PCA with χ^2-kernel on each sequence set using all data.

6.4.2 Exploitation of annotation

Evaluation follows the path: training the appearance-based model on the annotated bootstrap video then apply location estimation on the un-annotated video sequence. For several databases only one unlabeled video was provided hence we simulate a sparse annotation by labeling every 60 seconds (1800 frames) a window of 3 seconds (90 frames) which is then used in place of the bootstrap.

For proper evaluation in a realistic context, some of the methods depend on parameters that need to be adapted to the data processed. To this end, the bootstrap video sequence is divided in training and validation sets where the patterns are selected randomly into 5 folds. We devote 20% of the bootstrap for the validation purposes.

6.5 Evaluation on a test case

We selected one pair of annotated bootstrap and unlabeled video from the corpus of videos to analyze the performance of the algorithms reviewed and evaluated previously only on a database taken in controlled environment. The length of the annotated bootstrap video is 6 minutes and the length of unlabeled video is 33 minutes. More information about the data is given in Table 6.3 and in Fig. 6.2.

Bootstrap Video	Testing Video	Number of classes
11'709 frames (6 min)	59'784 frames (33 min)	6

Table 6.3: Description of the database

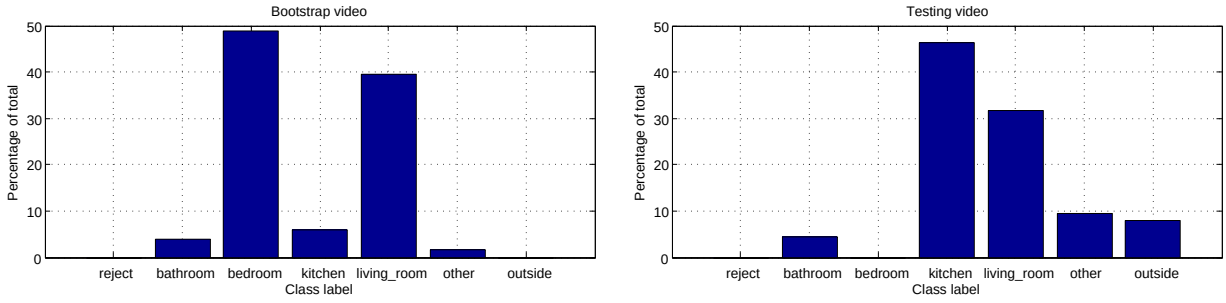


Figure 6.2: Class proportions in (a) bootstrap and (b) testing videos

6.5.1 Supervised classification

The baseline results presented in Fig. 6.3 show clearly the challenging nature of image-based localization for real world data. Both visual feature SVM classifiers trained on the bootstrap video and applied to the un-annotated video yield rather mediocre baseline performance, around 43% and 35% of global accuracy for BOVW and SPH features respectively. This is not surprising since many visual scenes appearing in the unlabeled video has not been observed in the annotated bootstrap. In Fig. 6.2 it is evident that some classes may not be covered enough in bootstrap even though they are dominant in the unlabeled video (e.g. class "kitchen").

Compared to experimental evaluations of the methods on the IDOL2 database, we notice a surprisingly low performance of the SPH visual descriptor compared to its simpler version BOVW. A possible explanation to this phenomena can be linked to substantially richer SPH descriptor than BOVW. In conditions of low and insufficient supervision, this may lead to more severe overfitting and thus lower performance on new testing data.

In the left panel of the same figure we depict the effect of temporal accumulation using the TA module. Performance increase is rather substantial for both visual features and achieve up to 53% and 41% of global accuracy for Temporal Accumulation applied on BOVW and SPH features respectively. Notice a rapid increase of performance and then decline as the averaging window gets larger. We explain this effect by the fact that most of the temporally close images in the video are assigned a correct estimated class label and the temporal smoothing effectively removes the occasional misclassifications.

In the right panel we show the classification results obtained by late fusion using DAS module. A gain of performance of up to 47% of global accuracy can be achieved by setting the α parameter properly. This increase of performance can be attributed to the complementarity property of the BOVW and SPH visual features. It interesting to note that without knowing a priori the performance of the global descriptors, one may select approximately $\alpha = 0.5$ and still get an improved final performance without knowing explicitly which feature performs the best. Compared to the temporal accumulation strategy shown in left panel, the improvement is more modest for classifier fusion.

Finally, in Fig. 6.3 in bottom panel we present the results for the combination of temporal accumulation (using $h_{\mathrm{BOF}} = 100$ and $h_{\mathrm{SPH}} = 500$) and late fusion strategies. Curiously, we do not observe performance increase, which possibly means that Temporal Accumulation defeated the

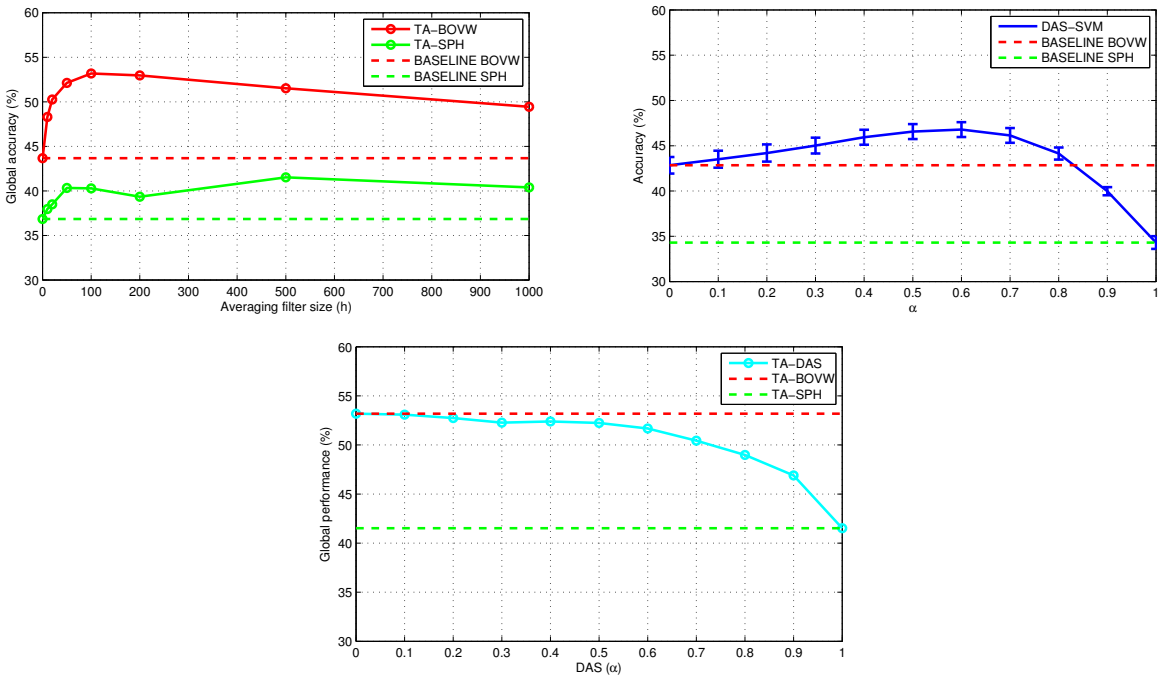


Figure 6.3: Baseline performances and effect of Temporal Accumulation: (left) Temporal; (right) Effect of Late Fusion; (bottom) Temporal accumulation followed by Late Fusion

complementarity of the features such that no additional improvement is possible.

The analysis of the current results showed that substantial performance increase is possible using straightforward temporal averaging and high-level classifier fusion strategies. In the next part of experiments we present the method capable leveraging both improvements.

6.5.2 Semi-supervised classification

The Co-Training based approach introduced in Chapter 4 includes an additional source of information corresponding to the exploration of unlabeled data for improving the training of visual appearance model.

The results for the CO-DAS method (using $\alpha_{DAS} = 0.7$) are shown in left panel of Fig. 6.4. Compared to the baseline methods BOVW-SVM and SPH-SVM on this particular data, we notice an increase in performance when less than 1% of ranked test set patterns are added to the training set (Fig. 6.4 left panel). Note that the increase of performance was made possible by an addition of a small part of the testing set. Going beyond this amount increases the risk of including incorrect estimates back into the training.

In right panel of Fig. 6.4 we compare the TA-CO-TA-DAS method to the rest by varying the internal temporal accumulation window size. Classification performance for this data clearly outperforms all the baselines and the individual methods exploiting solely feature complementarity or temporal structure. In this particular case we can recognize the majority of the improvement which is due to temporal accumulation done at basic level for the BOVW visual feature classifier since the improvement over TA-BOVW method is only 2% of global accuracy. All the results are summarized in Table 6.4.

The TA-CO-TA-DAS method requires two additional parameters in comparison to the CO-DAS

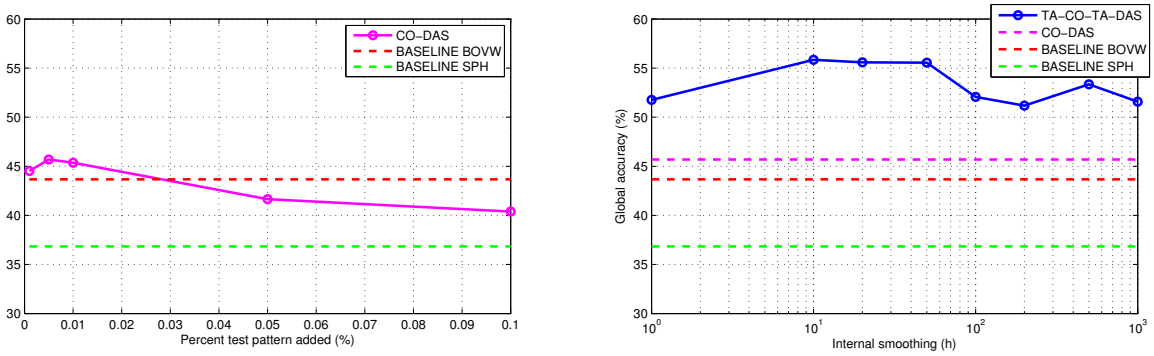


Figure 6.4: Semi-Supervised performance: (left) Co-Training and High-Level Classifier fusion; (right) Time-Aware Co-Training with High-Level Classifier fusion

BOVW-SVM	SPH-SVM	TA-BOVW-SVM	TA-SPH-SVM	TA-DAS	CO-DAS	TA-CO-TA-DAS
43 %	36 %	53 %	41 %	53 %	45 %	**55 %**

Table 6.4: Summary of the results on the selected video

method. We separate temporal accumulation in the inner feedback loop of the Co-Training method and the temporal accumulation before DAS fusion. The current best results were obtained using the window of size $h_{\text{int}} = 10$ in the internal feedback loop and $h_{\text{post}} = 100$ in the final stage.

6.6 Computational Cost

When dealing with large-scale applications, computation and storage requirements should be considered in addition to pure recognition performance. In the current study we work with 26 video sequences with an equivalent length of 10.8 hours. The video recordings are of High Definition (1280x960) with the frame rate of 30 frames per second which amounts to 1'171'508 images for the entire corpus of videos.

We separate two important factors: computation time and storage demands for visual features. The provided time estimates were obtained using 1 core in the HP Z800 machine with Intel Xeon E5520 processor, equipped with 12GB of RAM and a total amount of 2.7TB of disk space.

In Table 6.6 the cost of extraction of the three visual descriptors for 1 hour long video sequence is provided. We see that the computational time for the BOVW global descriptor is the shortest (10 hours) opposed to Spatial Pyramid Histogram global descriptor (40 hours). The latter is primarily due to the SIFT descriptor extraction using a dense and regular grid covering entire image.

According to these estimates, it would require 108, 324 and 442 hours to extract the global descriptors for BOVW, CRFH and SPH respectively. Equivalently, this amounts to 4.5, 13.5 and 18 days for the respective global descriptors.

Visual Feature	Processing time per 1 hour video
Bag of Visual Words (BOVW)	10 hours
Composite Receptive Field Histograms (CRFH)	30 hours
Spatial Pyramid Histograms (SPH)	40 hours

Table 6.5: Computational time requirements

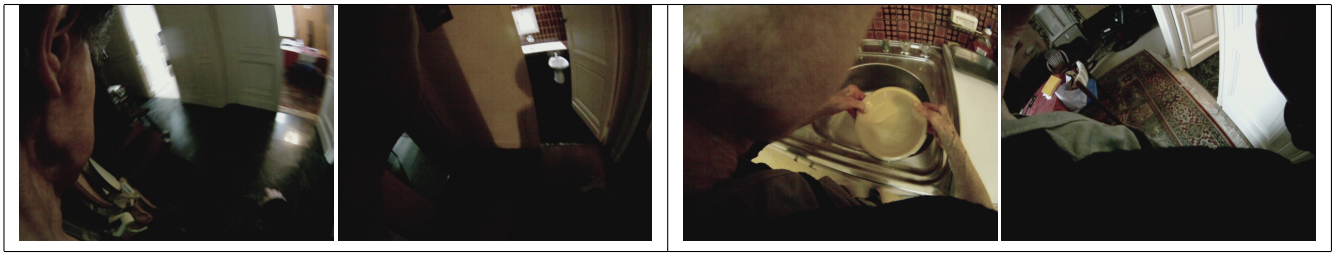


Figure 6.5: Principal issues: (left) dark scenes; (right) camera occlusion

6.7 Baseline image-based localization results

In this section we provide the image-based localization results on the large-scale by processing the remaining video sequence from the corpus. Based on the computational time considerations carried out in Section 6.6, we consider single feature approach methods for topological localization. The BOVW global descriptor was used as it yields comparably good recognition performance while being the fastest to compute.

The results for the baseline BOVW-SVM and the TA-BOVW-SVM method are depicted in Fig. 6.6 and 6.7. Generally we can note rather variable classification performance as a function of the patient or volunteer house. This variability can be explained mainly due to the quality of the the annotated bootstrap video and the complexity of the scenes found in the unlabeled videos. For example, in both bootstrap and unlabeled videos from the "P13" database (Fig. 6.6), the extracted frames feature very dark environment and considerable occlusion of the field of view of the camera. The sample frames are shown for this database in Fig. 6.5.

The main issue however is in the availability of sufficient amounts of training data. Basically, there is a difference between the information captured by the annotated bootstrap (global context) and unlabeled videos (specific activities). Of course, it is a challenging problem to recognize a location that has not been seen in the bootstrap video (e.g. close-up on the sink, reading a book) in a completely supervised fashion, although the former is expected to cover at least partially all the major places of interest. This difference in difficulty can be seen from the results obtained in realistic benchmark in video versus video setup (Fig. 6.6) and from in-video annotation strategy setup in left panel of Fig. 6.7.

Using a single feature for automatic place classification, we can partially alleviate the problem by applying the temporal accumulation scheme. The improvement for the vast majority of videos is evident. We argue that for starting performances higher than a chance and more or less temporally distributed correct location estimation, the temporal accumulation is a simple yet powerful method which can be used to bootstrap more advanced methods.

6.8 Conclusion

In this chapter we investigated application of the image-based localization methods on visual data captured in realistic setup and also revealed the challenges of large-scale processing. The study on a larger corpus of videos taken in realistic setup permitted to draw several conclusions when using our best performing method:

1. Leveraging temporal information is crucial when working with challenging video data;
2. Two or more visual descriptors may maximize the discriminative power on a frame basis but may not bring additional information after temporal consistency has been enforced;

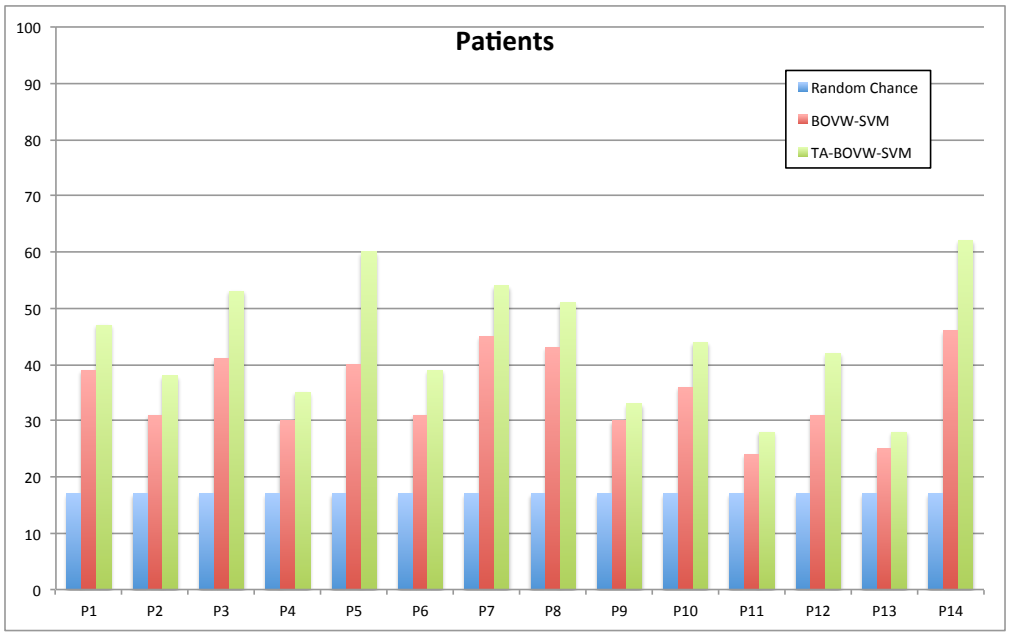


Figure 6.6: Patients: Image-based localization results

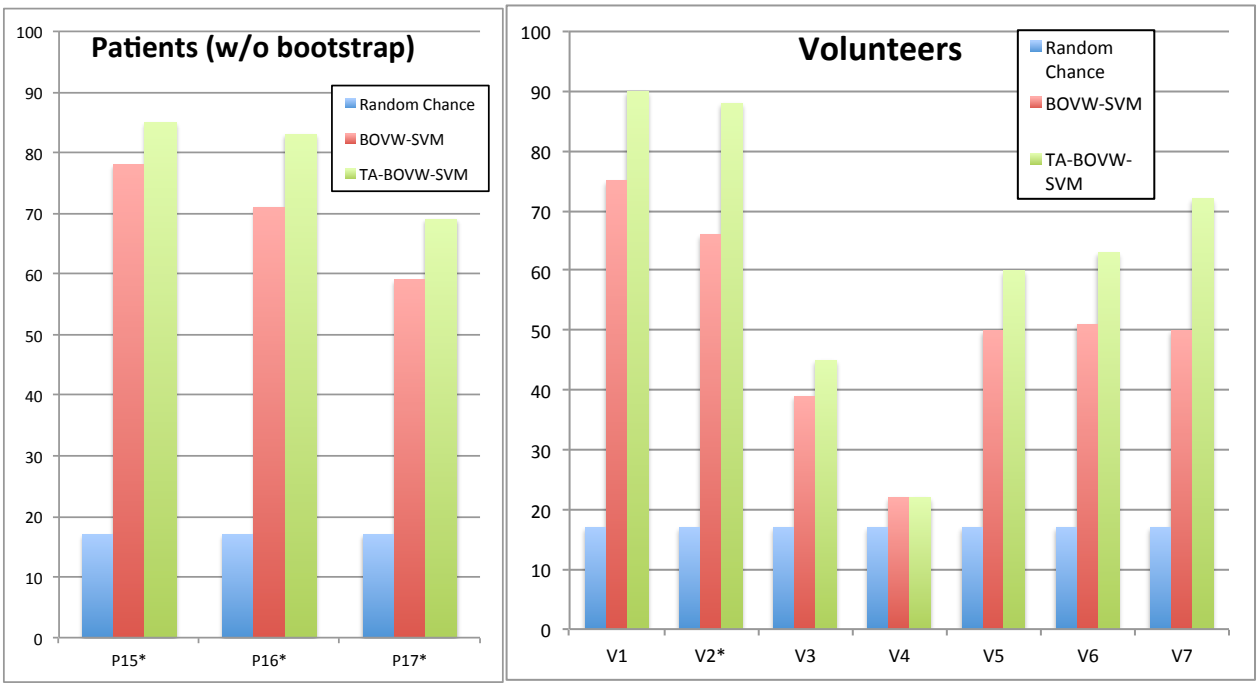


Figure 6.7: Image-based localization results: (left) Patients w/o bootstrap video; (right) Volunteers videos

3. Building iteratively the appearance model using most confident estimates may help to fill the discrepancies and missing parts between the bootstrap and unlabeled video content;

Chapter 7

Conclusion

7.1 Overview of the work

In this thesis a framework for image-based topological localization from wearable video has been proposed. The approach is purely image-based such that topological localization indoors of a camera wearer should be found bootstrapping the learning system using a small manually annotated video sequence. The problem of location estimation is turned into classification problem as every frame (or a subset) from the video sequence is expected to carry an index or tag of topological location.

The studied framework is based on extraction of one or several visual features from every frame of the video and constructing a global image representation using visual descriptors. The high dimensionality of descriptors is inherent to such global descriptors, and it is reduced using feature selection or dimensionality reduction methods. Dimensionality reduction effectively extracts the information in a data-dependent way such that the following decision stage receives all the class-specific information while removing irrelevant variability.

We have seen the discriminative power can be improved by exploiting the complementarity properties of several descriptors.

Another difficulty is the lack of training data. This led us to investigate semi-supervised learning methods such that information brought by unlabeled video sequence frames also are taken into account when learning the visual appearance model. Additionally, video sequences have the property that location estimation cannot be independent from frame to frame of the video sequence. Temporal continuity of the video is an additional constraint, which restricts the variability of possible location estimations. This information as integrated in the framework of the proposed time-aware co-training methods.

Finally we investigated a last type of prior knowledge that can stem from the knowledge about the possible visual information changes. Small horizontal displacement or rotation should result in no change to the localization result. These considerations were taken into account when creating a translation invariant descriptor.

In this thesis we have therefore investigated multiple ideas to answer the initial problematic of indoors localization estimation from wearable camera video data. In the next section we provide more details on the corresponding contributions.

7.2 Contributions

7.2.1 Place of feature selection in location estimation

In initial stages of the location recognition, it is crucial to extract relevant location-specific information. The global image descriptors often carry a lot of redundant information which poses a risk of overfitting when learning the visual appearance model for localization. We studied several dimensionality reduction methods which effectively restrict the variability of the visual descriptors. The compact image representations were created using an adapted intersection or χ^2 kernel in Kernel PCA or Laplacian Eigenmaps. The former discovers and projects the data in new axis dictated by the largest variance, while the latter preserves the locality property in reduced dimensionality space. We have shown the interest and applicability of the both methods when using the data-adapted kernel. The Laplacian Eigenmaps method also allows to represent the data in graph form where intuitive operations such as node pruning allow to remove erroneous links. We have shown that such pruning could improve the performances, but with the constraint of using a low number of neighbors. An arbitrary value will possibly decrease the performance if is not selected properly.

Ideal global image descriptors encoding the topological location can be very simple such that no sophisticated method for classification is necessary. The remaining complexity of the data after dimensionality reduction is effectively treated using an SVM classifier, which is known for its generalization capabilities.

7.2.2 Evaluating early and late fusion strategies with multiple features for localization

For successful localization estimation, discriminative power of the existing global descriptors should make difference between same and different topological locations. Unfortunately, without knowing what measurements from the image correspond exactly to the place "kitchen" and "living room", often low level information is extracted from which then global descriptors are created. Global descriptors also are different and may describe the color, shape, texture, corners, stable regions. This said, no global descriptor is absolutely superior for a specific task.

We evaluated an early and late fusion strategies, where the main idea is to exploit information contained within two or more descriptors. An example of early fusion is Multiple Kernel Learning method, which effectively selects the features and builds a new kernel. Late fusion strategy using individual visual feature classifier output merging follows a different way by combing SVM score outputs using either linear or non-linear kernel. We have shown experimentally on a controlled environment database of video sequences that both strategies bring an improvement therefore confirming the usefulness of complementary information visual features. Therefore, a novel discriminant visual feature can be seamlessly integrated into the framework and improve the recognition performance.

7.2.3 Exploiting unlabeled data and temporal continuity of the video

Indoor localization in real conditions is a typical scenario, where the lack of training data can impair the quality of the results. Indeed, to record all possible scenes and situations in house, and perform this video annotation and processing may be very costly. The idea is to leverage unlabeled data when learning the visual appearance model. State-of-the-art methods working in this paradigm are label propagation in the graph and Semi-Supervised SVM exploiting low density separation assumption.

Both methods perform relatively well when certain conditions are met. A properly constructed graph is very important for label propagation in the graph, and often suffers from class imbalance problem. Semi-supervised SVM requires an adapted kernel and careful parameter tuning.

We proposed a time-aware co-training method performing in semi-supervised paradigm, exploiting two visual features and also enforcing temporal continuity constraints. The novel theoretically motivated confidence measure is used in this framework to enrich the training set with confident estimates from the unlabeled video, whereas temporal accumulation effectively uses the information that frames in a small time interval should have similar class labels. Therefore, we have shown that in in small increments of the training set with novel patterns allows to avoid tackling the difficult problem of learning a possibly complex visual appearance model at once. The final fusion stage with prior temporal accumulation provides that both visual feature information is used and localized misclassifications are removed from the final localization estimation. Practical evaluation on real world data with typically small and incomplete bootstrap videos have shown that such learning strategy pays off, when using small additions of confident estimates to the training set.

7.2.4 Proposing a translation invariant global descriptor

The Spatial Pyramid Histogram (PH) global descriptor was shown to be one of the best discriminant descriptors for our task of image-based localization. The wearable camera is subject to motion and can exhibit some movements, which can result in the change of the class or not. The nature of how the PH descriptor is built is that spatial translation, which typically does not result in topological location change, nevertheless modify the descriptor. Enforcing the invariance of the classifier to horizontal spatial translations, which is also a principal type of displacement found in the videos, we showed how such a descriptor could be improved for the use in low supervision scenarios.

7.3 Future perspectives

In this thesis we have studied an image-based approach for topological localization indoors. Several other attractive research directions exist that can complement or improve the current recognition performance.

The current annotation strategy is limited to annotation of a bootstrap video. This corresponds to, in the context of the IMMED project, a 5 minute presentation of the places supervised by a medical assistant, which is the same person annotating the places from their knowledge, which is then used in an automatic indexing algorithm to learn the places models. This approach limits the final performance due to the quality and the representativity of the annotated bootstrap video. In a different annotation context, a pre-analysis of the data could be done before the annotation step. The quality of the bootstrap then could be improved by allowing the system to suggest image to be annotated in the bootstrap as well as in the unlabeled video. This alternative paradigm, referred to as active learning [128], has received particular attention for image retrieval from large unordered image databases. Unlabeled images automatically selected by an algorithm and annotated interactively could boost the performance of the overall location recognition. An example of such system from image retrieval domain is the RETIN system [56] and features noisy classifcation result removal as well as batch selection of relevant images in each relevenace feedback loop. A similar approach can be done also for location estimation where initial small annotation is gradually improved by a user providing his opinion on the estimated locations, which in turn could help to bootstrap the rest of the framework.

Another perspective for improvement is linked to the computational efficiency of the aforementioned kernel methods. Typically, large kernel matrix computation is required thus making the memory and computation burden heavier. Online kernel learning [64, 67] might be implemented thus opening the possibility to work with even larger scale video corpuses.

Appendix A

Feature Conditioning and Classification

A.1 Details on Principal Component Analysis

In this section the technical details for the Principal Component Analysis is provided, introduced in Subsection 2.4.1.

Variance and its maximization Lets define a sample covariance matrix of the input patterns

$$S_X = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \tag{A.1}$$

$$= \frac{1}{n-1}\left(X - \bar{X}\right)\left(X - \bar{X}\right)^T \tag{A.2}$$

with $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{d}\mathbf{x}_i$ being sample mean and $\bar{X} = \bar{\mathbf{x}}\mathbf{1}_n$ in a matrix form. Similarly, the sample mean for the projected data writes as $\bar{\mathbf{z}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{z}_i$ and the variance for the projected data as

$$S_Z = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{z}_i - \bar{\mathbf{z}})(\mathbf{z}_i - \bar{\mathbf{z}})^T \tag{A.3}$$

In a case of mapping to one dimension (one mapping vector $\mathbf{a}$), the variance of the data projected expresses

$$J(\mathbf{a}) = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{z}_i - \bar{\mathbf{z}})^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(\mathbf{a}^T\mathbf{x}_i - \mathbf{a}^T\bar{\mathbf{x}}\right)^2 = \mathbf{a}^T S_X \mathbf{a} \tag{A.4}$$

with the use of Eq. 2.1.

In PCA, the quantity to be maximized is the expression $J(\mathbf{a})$. To avoid arbitrarily large scaling of the data, the projection vector should be constrained. Typically the restriction is usually taken in the form of a constraint on the Euclidean norm $\|\mathbf{a}\|_2 = 1$.

Optimization problem With all these elements, the one dimensional case for dimensionality reduction reduces to the constrained maximization problem

$$\arg\max_{\mathbf{a}} \mathbf{a}^T S\mathbf{a} \quad \text{s.t. } \|\mathbf{a}\|_2 = 1 \tag{A.5}$$

Mapping vectors maximizing the the objective are found by solving the eigenproblem

$$S\mathbf{a} = \lambda\mathbf{a} \tag{A.6}$$

Largest eigenvalue eigenvector correspond to a direction which maximizes expression in Eq. A.4.

Generalizing to multiple dimensions The multi-dimensional case transformation involves k linear mappings $A = (\mathbf{a}_i)_{i=1}^k$. Following Eq. A.4 and requirement for the basis to be orthonormal $A^T A = 1$, the optimization problem easily generalizes to multiple dimension case

$$\arg\max_A \frac{\mathrm{Tr}\left(A^T S A\right)}{\mathrm{Tr}\left(A^T A\right)} \tag{A.7}$$

According to the Rayleigh-Ritz theorem [87], solution of the problem amounts to find the k largest eigenvalue eigenvectors of the covariance matrix S in Eq. A.6. In general, there are n solutions $\{(\mathbf{a}_i, \lambda_i)_{i=1}^n\}$ for a covariance matrix S of size $d \times d$. The eigenvalue λ_i describes the variance of the projected data on a particular projection direction $\mathbf{a}_i$.

For data of intrinsic linear dimensionality of $k < d$ dimensions, the top k eigenvalues will be the largest in the sorted list of eigenvalues. Typically, there will be a sharp difference between top valued eigenvalues and the remaining ones. Taking the top k valued eigenvectors $\mathbf{a}_i$, and using them in linear transformation manner, makes it possible in dimensionality reduction with PCA.

Pattern embedding Keeping the top k eigenvectors $A_k = (\mathbf{a}_i)_{i=1}^k$, any new pattern $\mathbf{x}_{\text{new}}$ can be embedded in k dimensions by applying the linear transformation

$$\mathbf{z}_{\text{new}} = A_k^T \mathbf{x}_{\text{new}} \tag{A.8}$$

A.2 Details on Kernel PCA

In this section the technical details for the Kernel PCA dimensionality reduction method is provided, introduced in Subsection 2.4.2.

Covariance matrix diagonalization in the RKHS Given a set of patterns $X = \{\mathbf{x}_i\}_{i=1}^n \in \mathcal{X}$ and an appropriate kernel function k, a Gram matrix $(K)_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ can be computed. KPCA tries to find maximum variance directions in the high dimensional space $\mathcal{H}$ obtained by non-linear mapping of the data. Following [136], covariance matrix can be written using mapped patterns

$$S_{\mathcal{H}} = \frac{1}{n}\sum_{i=1}^{n} \Phi(\mathbf{x}_i)\, \Phi(\mathbf{x}_i)^T \tag{A.9}$$

where the mapping function Φ is induced by an explicit choice of kernel function. As in the case of linear PCA, a covariance matrix is diagonalized by its eigenvectors

$$S_{\mathcal{H}} = U \Lambda U^T \tag{A.10}$$

which is equivalent to solve an eigenproblem

$$S_{\mathcal{H}} \mathbf{e}_i = \lambda_i \mathbf{e}_i \tag{A.11}$$

where $U = (\mathbf{e}_i)_{i=1}^n$ and $\Lambda = \mathrm{diag}\,(\lambda_i)_{i=1}^n$ are the matrices of eigenvectors and eigenvalues respectively.

Solving for principal components In practice, the covariance matrix $S_{\mathcal{H}}$ cannot be computed due to unknown mapping function Φ. The eigenproblem in Eq.A.11 therefore cannot be solved explicitly. However, the eigenvectors of this matrix can expressed in terms of mapped patterns $\Phi(\mathbf{x})$. By plugging Equation A.9 into A.11, the eigenvector $\mathbf{e}_i$ can be expressed in the basis of $\Phi(\mathbf{x}_j)$

$$\mathbf{e}_i = [\Phi(\mathbf{x}_1), \ldots, \Phi(\mathbf{x}_n)]\,\mathbf{a}_i \tag{A.12}$$

where $A = (\mathbf{a}_i)_{i=1}^n$ are the expansion coefficient vectors. It can be shown that the eigenproblem in Eq. A.11 can be expressed purely in terms of the expansion coefficients $\mathbf{a}_i$ and the kernel matrix K. Then the eigenproblem writes

$$K\mathbf{a}_i = \lambda_i n \mathbf{a}_i \tag{A.13}$$

Refer to [26] and [16] for in-depth discussion and derivation details.

To this end, the problem has been reduced to an eigenproblem involving the kernel matrix and expansion coefficients.

Transductive pattern embedding The best representation of the data in KPCA sense means to project it onto the directions of the largest variance. Suppose that a selection of $k < n$ eigenvectors U_k was done corresponding to the largest eigenvalues. Embedding of a d-dimensional pattern $\mathbf{x}$ in k dimensions would correspond to a projection

$$\mathbf{z} = U_k^T \Phi(\mathbf{x}) \tag{A.14}$$

As in case with covariance matrix and the eigenproblem, in practice such projection cannot be computed due to unknown mapping function and the eigenvectors. However, the eigenvector expansion in Eq. A.12 can be used. Using this result, a k-dimensional embedding can be expressed using the computed expansion coefficients and the kernel matrix as follows

$$\mathbf{z}_i = A_k^T K(\mathbf{x}_i) \tag{A.15}$$

where the matrix A_k corresponds to a selection of k expansion vectors $\mathbf{a}_i$ corresponding to the largest eigenvalues found by solving A.13 and $K(\mathbf{x}_i) = [k(\mathbf{x}_i, \mathbf{x}_1), \ldots, k(\mathbf{x}_i, \mathbf{x}_n)]^T$ is a column vector. Refer to [136] for detailed derivation.

Out-of-Sample Extension Embedding of an unseen pattern $\mathbf{x}_{\text{new}}$ can be carried out into the embedding space. Suppose that an additional kernel matrix column has been computed

$$K(\mathbf{x}_{\text{new}}) = [k(\mathbf{x}_1, \mathbf{x}_{\text{new}}), \ldots, k(\mathbf{x}_n, \mathbf{x}_{\text{new}})]^T \tag{A.16}$$

and the same expansion coefficient matrix A_k is reused from the transductive embedding step. Then embedding of the pattern $\mathbf{x}_{\text{new}}$ can be found from

$$\mathbf{z}_{\text{new}} = A_k^T K(\mathbf{x}_{\text{new}}) \tag{A.17}$$

Kernel matrix centering As in linear PCA, the centering of mapped patterns $\Phi(\mathbf{x})$ is also required to find largest variance projection directions in the feature space. In general this is not assured when using a proper kernel function k. Centered mappings

$$\tilde{\Phi}(\mathbf{x}_i) = \Phi(\mathbf{x}_i) - \frac{1}{n}\sum_{j=1}^{n}\Phi(\mathbf{x}_j) \tag{A.18}$$

yield a centered kernel $\tilde{K}_{ij} = \left\langle \tilde{\Phi}(\mathbf{x}_i), \tilde{\Phi}(\mathbf{x}_j) \right\rangle_{\mathcal{H}}$ that is still PSD. Denoting a matrix $(\mathbf{1}_n)_{ij} = \frac{1}{n}$, the centered kernel matrix can be expressed in its non-centered version

$$\tilde{K}_{ij} = K - \mathbf{1}_n K - K\mathbf{1}_n + \mathbf{1}_n K \mathbf{1}_n \tag{A.19}$$

Refer to [136] for derivation of this result. In practical applications using KPCA, the kernel matrix should be always centered regardless the kernel function.

A.3 Details on Laplacian Eigenmaps

In this section we provide the technical details for dimensionality reduction with the Laplacian Eigenmaps method, introduced in Subsection 2.4.3. The place and the role of graph Laplacian is also discussed.

Definition of Graph Laplacian Lets define a function defined on a graph $f : V \to \mathcal{R}$ which returns a real value for any node in the graph. Regularizing according to the graph structure should enforce similar values for nodes with strong affinities W_{ij}. Such intuition is used in regularization for semi-supervised methods, see [27] for excellent work devoted to diverse methods and [181] for extensive review of the subject. The following energy function has the required properties [162][31]

$$E(f) = \frac{1}{2}\sum_{i,j} w_{ij}\left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2 \tag{A.20}$$

Lets define a diagonal degree matrix $D_{ii} = \sum_j W_{ij}$. Then non-normalized graph Laplacian [31, 162] is defined as

$$\mathcal{L} = D - W \tag{A.21}$$

and it can be shown that Eq. A.20 takes the following form

$$E(f) = f^T \mathcal{L} f \tag{A.22}$$

In practical application for dimensionality reduction, the function f is finite dimensional.

Embedding problem The method utilizes the notion of the graph which is constructed from a data set $X = \{\mathbf{x}_i\}_{i=1}^n \in \mathcal{R}^d$ and represented by an affinity matrix W. Suppose that one wants to find a mapping vector $\mathbf{e}$ such that an element $\mathbf{e}(k)$ can be seen as one dimensional embedding of a pattern $\mathbf{x}_k$. Those one dimensional embeddings should respect the similarities in matrix W in the sense that patterns with strong affinities should be matched close. Such mapping can be found if one manages to minimize the following energy function

$$J(\mathbf{e}) = \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n}\left(\mathbf{e}(j) - \mathbf{e}(k)\right)^2 W_{jk} = \mathbf{e}^T(D - W)\mathbf{e} = \mathbf{e}^T\mathcal{L}\mathbf{e} \tag{A.23}$$

This objective function states that for similar patterns $\mathbf{x}_j$ and $\mathbf{x}_k$ (according to W_{jk}) a penalty will be incurred if their respective embeddings $\mathbf{e}(j)$ and $\mathbf{e}(k)$ are mapped far apart. To this end, minimization involves a quantity termed Laplacian of the graph.

Minimization problem The quantity in Eq. A.23 should be minimized together with a constraint that removes arbitrary scaling of the data. Together with the constraint $\mathbf{e}^T\mathbf{e} = 1$, this corresponds to the non-normalized Laplacian Eigenmap [11]

$$\mathcal{L}\mathbf{e} = \lambda\mathbf{e} \tag{A.24}$$

Each solution vector $\mathbf{e}_i$ to this eigenproblem contains all pattern embeddings in one dimension. Dimension i corresponds to a degree of smoothness (or coarsity) of a function f which is measured by an eigenvalue λ_i.

Graph Laplacian $\mathcal{L}$ has two interesting properties that will be exploited in further discussions:

1. Clustering property
   If a graph has p disjoint components, eigendecomposition of this graph Laplacian will amount exactly p zero eigenvalue eigenvectors. The corresponding eigenvectors will indicate the interconnected nodes with non-zero elements. This property can be useful for data clustering.

2. Label Smoothness property
   If eigenvalues are sorted in ascending order, then smaller eigenvalues will correspond to smoother eigenvectors (functions f). It is interesting to note that if there are large weight paths between some nodes in a graph, the respective elements in the smooth function (eigenvector) will have similar value. This property is useful for semi-supervised learning.

These properties are useful for the task of dimensionality reduction where input space feature vectors are of very high dimensionality and with complex class boundaries. A regularizer exhibiting such properties can restrict the complexity of the learned model [91].

Refer to [162, 31] for more comprehensive reference about graph relationships with graph Laplacian, eigendecomposition, adjacency matrix and related subjects in spectral graph theory.

Normalized and un-normalized graph Laplacians From literature [162] there are three types of graph Laplacian:

1. Un-normalized $\mathcal{L} = D - W$
2. Random walk $\mathcal{L}_{\text{rw}} = I - D^{-1}W$
3. Symmetric $\mathcal{L}_{\text{sym}} = I - D^{-\frac{1}{2}}WD^{-\frac{1}{2}}$

Although all of them have been used for clustering, there are subtle differences in their properties. According to [162], if node degree distributions vary broadly, then both normalized Laplacians $\mathcal{L}_{\text{rw}}$ or $\mathcal{L}_{\text{sym}}$ would take that into account.

In clustering the goal is to achieve separation between different clusters and optionally to find a partition such that points within one cluster are close to each other. Un-normalized Laplacian satisfies the former but does not support the latter. A normalized graph Laplacian supports both requirements which may be preferable for some applications.

The difference between two normalized versions of Laplacian is in an additive multiplicative term in eigenvectors for $\mathcal{L}_{\text{sym}}$ which may be a problem. Besides, from a computational point of view there

are advantages using $\mathcal{L}_{\text{rw}}$ Laplacian. The eigenproblem for this Laplacian is equivalent to solving a generalized eigenproblem

$$W\tilde{\mathbf{e}} = \tilde{\lambda}D\tilde{\mathbf{e}} \tag{A.25}$$

where one needs to find largest eigenvalue $\tilde{\lambda} = 1 - \lambda$ eigenvectors. Such formulation does not require explicit computation of graph Laplacian matrix. Many software packages implement the methods where it is numerically more efficient as Nystrom approximation [52] to find largest eigenvalue eigenvectors than the smallest ones.

Transductive pattern embedding The solution of an eigenproblem is a set of eigenvectors which approximate the eigenfunctions of Laplace Beltrami operator [11] defined on some manifold $\mathcal{M}$. If sorted the eigenvalues in ascending order $0 = \lambda_1 < \lambda_2 < \ldots$, the corresponding eigenvectors will range from the smoothest to less smooth functions. Smooth functions correspond to large and coarse graph representation while less smooth functions correspond to small variations in the graph.

This is the key in dimensionality reduction using the spectral properties of the graph. In this framework a pattern $\mathbf{x}_i$ correponds to an embedding with $k < d$ dimensions by taking i-th element from the first k most smooth eigenvectors $\hat{U} = (\mathbf{e}_1, \ldots \mathbf{e}_k)$. Thus k-dimensional embedding for a pattern $\mathbf{x}_i$ respecting locality information takes the form

$$\mathbf{z}_i = \left[\mathbf{e}_1(i), \ldots, \mathbf{e}_k(i)\right]^T \tag{A.26}$$

A.4 Elements of Bayesian decision theory

Bayesian decision theory is a field in statistical learning that applies probabilistic treatment to the problem of pattern classification. We do not aim to provide comprehensive insight into the field but rather as an introduction into the field of pattern classification. Many tools and frameworks not necessarily stem from probabilistic domain can be cast or analyzed using Bayesian view.

Basic elements Suppose that a binary classification problem is given. We denote the two classes as $y = \omega_1$ and $y = \omega_2$. Usually some prior information about the problem is available, such as prior probabilities

$$0 \leq \quad P(y = \omega_1) \quad \leq 1 \tag{A.27}$$
$$0 \leq \quad P(y = \omega_2) \quad \leq 1 \tag{A.28}$$

With only this information, it makes sense to classify an object $\mathbf{x}$ to a class

$$\hat{y} = \begin{cases} \omega_1 & \text{if } P(\omega_1) > P(\omega_2) \\ \omega_2 & \text{if } P(\omega_1) < P(\omega_2) \end{cases} \tag{A.29}$$

For real world classification problems this would not be enough. Usually a training set $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ is given. Then the problem is defined as how to find the missing labels for the set $U = \{\mathbf{x}_i\}_{i=l+1}^{l+u}$.

The cues useful for classification can be found from class conditional probability densities $p(\mathbf{x}|y)$. If class conditional densities are generally different (e.g. not overlapping), we can say that information contained in $\mathbf{x}_i$ is characteristic or more probable for the class y_i.

Recall that the goal is to obtain a class label y for an object $\mathbf{x}_j \in U$. This can be expressed as a probability

$$0 \leq P\left(y|\mathbf{x}_j\right) \leq 1 \tag{A.30}$$

also known as posterior probability. For example, one may ask : what is a probability of given object $\mathbf{x}_j$ belonging to the class, say $y = \omega_1$? Naturally, the object belongs to a class which is more probable

$$\hat{y}_j = \arg\max_y P\left(y|\mathbf{x}_j\right) \tag{A.31}$$

Bayes formula All the data together with its corresponding class labels can be seen as samples from joint probability density $p\left(\mathbf{x}, y\right)$. From the assumption of independence, the joint distribution can be decomposed in two ways

$$p\left(\mathbf{x}, y\right) = p\left(\mathbf{x}|y\right) P\left(y\right) = P\left(y|\mathbf{x}\right) p\left(\mathbf{x}\right) \tag{A.32}$$

We are interested in posterior $P\left(y|\mathbf{x}\right)$ which can be obtained from equality of the two decompositions of the joint probability

$$P\left(y|\mathbf{x}\right) = \frac{p\left(\mathbf{x}|y\right) P\left(y\right)}{p\left(\mathbf{x}\right)} \tag{A.33}$$

This result is known as Bayes formula. Knowing class conditional distribution $p\left(\mathbf{x}|y\right)$ and priors probabilities $P\left(y\right)$, it is possible to infer the probability of an object belonging to any of the two classes. Note that the denominator can be obtained from the joint distribution by marginalizing or summing over variable y

$$p\left(\mathbf{x}\right) = \sum_{k=1}^{2} p\left(\mathbf{x}|y\right) P\left(y\right) \tag{A.34}$$

Final remarks The presented classifier is often known as Naive Bayes classifier. However, there are two principal difficulties : independence assumption and density estimation problem.

Naivety comes from the independence assumption between the features in the object $\mathbf{x} = \left(x_1, \ldots, x_d\right)^T$ and the label

$$p\left(\mathbf{x}, y\right) = P\left(y\right) \prod_{k=1}^{d} p\left(x_k|y\right) \tag{A.35}$$

In many problems this assumption is false although it greatly simplifies the computation.

Note that for application of Bayes formula the class conditional densities are necessary. The problem of density estimation is not trivial and is further complicated by the matter of many variables or features in object $\mathbf{x}$. In practical application the models of density functions can be very complex or completely unknown.

For more details on the subject, refer to standard texts [16, 46] and a good review about Bayesian inference [144].

A.5 Linearly separably and non-separable data

In this section, we detail on how to learn the SVM classifier introduced in Section 2.5.1, in both separable and non-separable cases.

A.5.1 Problem for Linearly separable data

The margin should be maximized in order to get the best generalization performance of a classifier. A classifier with poor generalization properties will perform poorly on new unlabeled patterns. Similarly, a classifier with good generalization properties will more likely classify correctly unseen patterns. To maximize a margin and taking into account the training data, the following constrained problem should be solved, which minimizes the inverse of the margin under constraint that the pair of shifted hyper-planes are separating hyper-planes.

$$\min_{\mathbf{w}\in\mathcal{H},b\in\mathcal{R}} \frac{1}{2}\|\mathbf{w}\|^2 \tag{A.36}$$

$$y_i\left(\langle\mathbf{w},\mathbf{x}_i\rangle_{\mathcal{H}}+b\right)\geq 1,\quad i=1,\ldots,m \tag{A.37}$$

This optimization problem is usually solved using the method of unknown Lagrange coefficients where objective function and constraints are present in one expression. The method defines Lagrange coefficients $\alpha_i,\quad i=1,\ldots,m$ and Lagrangian

$$L\left(\mathbf{w},b,\alpha\right)=\frac{1}{2}\|\mathbf{w}\|^2-\sum_{i=1}^{m}\alpha_i\left(y_i\left[\langle\mathbf{x}_i,\mathbf{w}\rangle_{\mathcal{H}}+b\right]-1\right) \tag{A.38}$$

which should be minimized with respect to the so-called primal variables $\mathbf{w},b$ and maximized with respect to $\alpha_i,\quad i=1,\ldots,m$. It is important to note that if inequality constraints in Eq. A.37 is not met for some pattern $\mathbf{x}_i$, the corresponding α_i is set to zero thus ensuring that the Lagrangian is maximized. This is one of Karush-Kuhn-Tucker (KKT) conditions [71] stating that derivatives of L with respect to the primal variables should vanish.

$$\frac{\partial}{\partial b}L\left(\mathbf{w},b,\alpha\right)=0\quad\text{and}\quad\frac{\partial}{\partial\mathbf{w}}L\left(\mathbf{w},b,\alpha\right)=0 \tag{A.39}$$

which leads to

$$\sum_{i=1}^{m}\alpha_i y_i=0 \tag{A.40}$$

and a solution defining a normal vector of a maximal margin hyperplane defined by its normal vector $\mathbf{w}$

$$\mathbf{w}=\sum_{i=1}^{m}\alpha_i y_i\mathbf{x}_i \tag{A.41}$$

Another interesting property is that the expansion of $\mathbf{w}$ in Equation A.41 contains only the terms with $\alpha_i\neq 0$. Patterns with corresponding non-zero α_i are called support vectors. From KKT conditions support vectors lie on the margin and render other patterns irrelevant for the optimization. Other patterns could have been as well left out. It follows that the hyperplane is determined by the support vectors since the solution does not depend on other patterns.

Substituting back Eq. A.40 and A.41 into the expression of the Lagrangian in Eq. A.38, the dual problem of the optimization is obtained. Note that the primal variables are eliminated.

$$\max_{\boldsymbol{\alpha}\in\mathcal{R}^m} W(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i\alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle_{\mathcal{H}} \tag{A.42}$$

$$\text{s.t. } \alpha_i \geq 0, \quad i = 1,\ldots,m \quad \text{and} \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \tag{A.43}$$

The decision function f can now be written. Note that only support vectors contribute to the estimation of the class for a new pattern. Offset variable b is computed by exploiting KKT conditions. Using the notion of kernel defined in Subsection 2.5.1

$$f(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^{m} y_i\alpha_i \langle \mathbf{x}_i, \mathbf{x}\rangle_{\mathcal{H}} + b\right) = \text{sign}\left(\sum_{i=1}^{m} y_i\alpha_i k(\mathbf{x}_i, \mathbf{x}) + b\right) \tag{A.44}$$

A.5.2 Non-separable case of SVM

In real world conditions a separating hyperplane may not exist. There will be no solution to the problem defined in Eq. A.36 if there is a class overlap.

From literature, a classifier defined in Section A.5.1 is known as hard margin classifier since the patterns violating inequality constraints in Eq. A.37 may influence the solution strongly. Allowing some patterns to violate the inequality conditions increases the chance to find a solution that is more robust to noisy patterns. This gives a rise to soft-margin classifier by introducing slack variables ξ

$$\xi_i \geq 0 \quad \text{and} \quad i = 1,\ldots,m \tag{A.45}$$

that relax the constraints

$$y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle_{\mathcal{H}} + b\right) \geq 1 - \xi_i, \quad i = 1,\ldots,m \tag{A.46}$$

Depending on the data, there may be a different proportion of noisy patterns. By keeping some amount of noisy patterns (corresponding $\xi_i \neq 0$) control of generalization properties of a classifier is done. With no or few noisy patterns renders a classifier to have a margin with low tolerance of error. Such classifier will classify correctly most of the training patterns but may fail on new unlabeled patterns. A classifier with greater generalization capabilities may tolerate more erroneous patterns in the training set but may classify the unlabeled patterns correctly with a higher chance.

Therefore, the maximal margin is found by minimizing the norm of $\mathbf{w}$ and by tolerating some errors in the training set. Pulling all the elements together soft-margin classifier emerges as

$$\min_{\mathbf{w}\in\mathcal{H}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m} \xi_i \tag{A.47}$$

$$\text{s.t. } y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle_{\mathcal{H}} + b\right) \geq 1 - \xi_i \quad \text{and} \quad \xi_i \geq 0, i = 1,\ldots,m \tag{A.48}$$

where the parameter C controls the trade-off between margin maximization and training error minimization.The Lagrangian for soft-margin classifier leads to a similar formulation the quadratic problem as in Eq. A.42 and A.43 but subject to additional constraints

$$0 \leq \alpha_i \leq C, \quad i = 1,\ldots,m \tag{A.49}$$

Solution in form of decision function is the same as in Eq. A.44 but the influence of individual (and possibly outlier) patterns used for training is now limited.

Appendix B

Multiple Feature exploitation

In this annex we provide technical details on Multiple Kernel Learning method, introduced in Section 3.2.

B.1 Details on Multiple Kernel Learning

Multiple Kernel Learning is often cast in SVM framework. Recall that standard single kernel SVM has a following primal problem (see Chapter 2 for review)

$$\min_{f} \quad \tfrac{1}{2}\sum_{i=1}^{n}\|f\|^2 + C\sum_{i=1}^{n}\xi_i \tag{B.1}$$

$$\text{s.t.} \qquad y_i f(\mathbf{x}_i) \geq 1 - \xi_i \tag{B.2}$$

$$\forall i, \xi_i \geq 0 \tag{B.3}$$

where $f \in \mathcal{H}$ and the solution complies with 3.1. Using sum rule to build a new kernel, a richer model (function f^{MKL}) can be learned. Substituting Eq. 3.8 into Eq. 3.1 we obtain

$$f^{\text{MKL}}(\mathbf{x}) = \sum_{k=1}^{m}\boldsymbol{\beta}_k f_k(\mathbf{x}) = \sum_{k=1}^{m}\boldsymbol{\beta}_k \sum_{i=1}^{n}\alpha_i K_k(\mathbf{x}_i, \mathbf{x}) + b \tag{B.4}$$

The solution to the problem now includes three parameters - $\boldsymbol{\alpha} \in \mathcal{R}^n, b \in \mathcal{R}, \boldsymbol{\beta} \in \mathcal{R}^m$.

Primal Problem Following [118], the primal problem for multiple kernel case changes from Eq. B.1 to

$$\min_{\{f_i\},\boldsymbol{\beta}} \quad \tfrac{1}{2}\sum_{k=1}^{m}\frac{\|f_k\|^2_{\mathcal{H}_k}}{\boldsymbol{\beta}_k} + C\sum_{i=1}^{n}\xi_i \tag{B.5}$$

$$\text{s.t.} \qquad y_i\sum_{k=1}^{m} f_k(\mathbf{x}_i) \geq 1 - \xi_i \tag{B.6}$$

$$\xi_i \geq 0 \qquad \forall i \tag{B.7}$$

$$\sum_{k=1}^{m}\boldsymbol{\beta}_k = 1, \boldsymbol{\beta}_k \geq 0 \qquad \forall k \tag{B.8}$$

An important property of this formulation is that the minimization problem remains convex and smooth which allows to use a standard SVM solver. This property can be attributed to a chain depicted in Fig. B.1.

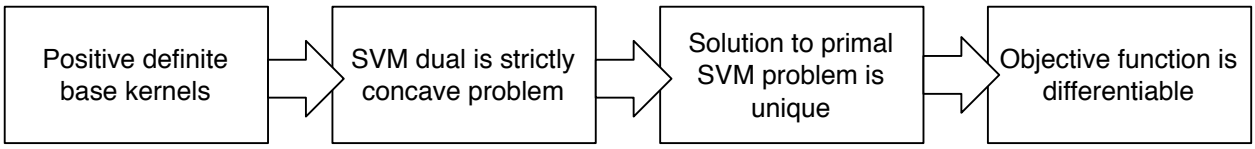


Figure B.1: Chain of reasoning leading to well-posed optimization problem for MKL

If solved in primal, a gradient descent method to minimize the objective function can be used. Smoothness of the function f_k is measured by the norm $\|f_k\|_{\mathcal{H}_k}$ weighted by a coefficient $\boldsymbol{\beta}_k$. Convention used in [118] assumes that if $\boldsymbol{\beta}_k = 0$, then the $\|f_k\|_{\mathcal{H}_k} = 0$ (f is a zero element of $\mathcal{H}_k$) to obtain a finite final objective function value. This formulation endorses weighted sparsity in kernel selection.

Dual Problem The primal problem B.5 can be rewritten in Lagrangian dual representation. This representation is interesting since the inequality constraints from the primal problem are now equality constraints which makes the problem easier to handle. Another important improvement is that now training patterns enter in pairs in form of dot products. Finally, the cost function is not dependent upon the dimensionality of the input space [143].

Omitting the derivation details, we write the new optimization problem as

$$\max_{\alpha,\lambda} \quad \sum_{\alpha=1}^{n} \alpha_i - \lambda \tag{B.9}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq C \qquad i = 1, \ldots, n \tag{B.10}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{B.11}$$

$$\tfrac{1}{2} \sum_{i=1}^{n} \alpha_i y_i \sum_{j=1}^{n} \alpha_j y_j k_k (\mathbf{x}_i, \mathbf{x}_j) \leq \lambda \quad k = 1, \ldots, m \tag{B.12}$$

In spite of aforementioned properties of dual problems, the problem in Eq. B.9 is difficult to optimize due to the last inequality constraint. Instead, one can write the Lagrangian of the MKL problem and set the derivatives with respect to its variables to zero. This operation yields a number of conditions from which a dual problem can be found (see [118] for more details). It is interesting to note that the objective function for primal and dual problems are the same due to duality property and strongly resembles the standard single kernel objective function

$$\max_{\alpha,\boldsymbol{\beta}} \quad \sum_{i=1}^{n} \alpha_i - \tfrac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \underbrace{\sum_{k=1}^{m} \boldsymbol{\beta}_k k_k (\mathbf{x}_i, \mathbf{x}_j)}_{(K)_{ij}} \tag{B.13}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \; 0 \leq \alpha_i \leq C \qquad i = 1, \ldots . n \tag{B.14}$$

$$\sum_{k=1}^{m} \boldsymbol{\beta}_k = 1, \boldsymbol{\beta}_k \geq 0 \qquad k = 1, \ldots, m \tag{B.15}$$

where the new combined kernel Gram matrix K enters as an internal weighted sum of individual base kernel functions. As stated previously, to solve the maximization problem, the algorithm must find the coefficients $\{\alpha_i\}_{i=1}^{n}$ and the kernel weights $\{\boldsymbol{\beta}_k\}_{k=1}^{m}$. Note that a new constraint has been added : the kernel weights should sum up to 1.

Refer to Chapter 2 for review on primal and dual problem formulations.

Appendix C

Details on Semi-Supervised Learning

C.1 Semi-Supervised Laplacian SVM

In this section we provide the technical details for the Laplacian Semi-Superivsed SVM classifier, introduced in Subsection 4.2.3.

Solving LapSVM Following [13], optimization problem in Eq. 4.15 can be written as a quadratic program in dual formulation

$$J(\beta) = \max_{\beta \in \mathcal{R}^l} \sum_{i=1}^{l} \beta_i - \frac{1}{2} \beta^T Q \beta \tag{C.1}$$

subject to constraints $\sum_{i=1}^{l} y_i \beta_i = 0, \quad 0 \leq \beta \leq \frac{1}{l}, \quad i = 1, \ldots, l$ and we denote

$$Q = YTK \left(2\gamma_A I + 2 \frac{\gamma_I}{(l+u)^2} LK \right)^{-1} T^T Y \tag{C.2}$$

where we denote

- K - a Gram matrix over labeled and unlabeled data;
- Y - a diagonal matrix with labeled data information $Y_{ii} = y_i$;
- L - a graph Laplacian over labeled and unlabeled data;
- T - label indicator matrix of size $l \times (l+u)$ and is built such that $T_{ij} = 1$ if $i = j$ and $\mathbf{x}_i$ is a labeled pattern;

The goal is to find the expansion coefficients $\hat{\alpha}_i \in \mathcal{R}^{(l+u)}$ and the new data-dependent Gram kernel $\tilde{K}$ conforming to Eq. 4.16. Knowing the solution $\hat{\beta}$ for Eq. C.1, the expansion coefficients can be found by solving

$$\hat{\alpha} = \left(2\gamma_A I + 2 \frac{\gamma_I}{(l+u)^2} LK \right)^{-1} T^T Y \hat{\beta} \tag{C.3}$$

With all these elements, the algorithm for Laplacian SVM using soft margin loss function can be expressed as an algorithm following [13, 27]:

1. Build an affinity matrix W from labeled and unlabeled data;

2. Build a Gram matrix K;
3. Compute the graph Laplacian matrix L;
4. Select regularization parameters γ_A and γ_I;
5. Solve quadratic program in Eq. C.1 for $\hat{\beta}$;
6. Solve for expansion coefficients $\hat{\alpha}$ using Eq. C.3;
7. Use the function $f(\mathbf{x}) = \sum_{i=1}^{l+u} \hat{\alpha}_i K(\mathbf{x}_i, \mathbf{x})$ to obtain predictions on the future data;

Appendix D

The IDOL2 database

Overview

In this annex we provide more detailed description of the KTH-IDOL2 or shortly the IDOL2 database. The database consists of video sequences captured by two robot platforms with a goal to evaluate the robustness and suitability of the image-based localization algorithms in real-world conditions. The video sequences were captured in The Computational Vision and Active Perception Laboratory at the Royal Institute of Technology, Sweden.

The database consists of 24 short video sequences recorded with the perspective camera Canon VC-C4 camera at the framerate of 5 fps. The effective resolution of the extracted images is 309 x 240. Half of the video sequences were recorded by the "minnie" robot with a height of the camera above the ground of 98 cm, and a half by the "dumbo" robot with respective height of 36 cm above the floor.

Typical size of the image database extracted from one video sequence is 800-1100 images.

Topological locations

The both robot platforms were manually driven in the same indoors environment following approximately the same path through 5 different functionality localizations and visual, laser scans and odometry data was recorded. We use these 5 locations for topological localization (see Fig. D.1 for sample images). The names of the locations are given in the Table:

One-Person Office	Two-Person Office	Corridor	Kitchen	Printer Area

It should be noted the "Printer Area" is actually a mere prolongation of the "Corridor" and some of the doors separating two different functionality areas are made of transparent glass. Moreover, groundtruth of the images occuring on a transition between two functional areas was attributed to one of the classes arbitrary thus creating labeling noise issue.

Visual data variability

The video recordings has been recorded in 3 different lighting conditions:

Cloudy	Night	Sunny



Figure D.1: Sample images: (a) Printer Area, (b) Corridor, (c) Two-Person Office, (d) One-Person Office and (e) Kitchen

with two video recordings per condition. The artificial lighting indoors was always kept on. Additional two video sequences per lighing condition and the robot platform were made across a span of 6 months. Therefore, both lighting conditions and natural scene changes (moved furniture, people walking etc).

The video camera was equipped with a sensor featuring automatic exposure control. This resulted occasionally in overly dark regions in the presence of large light contrasts and some blur due to rapid robot rotations.

Evaluation protocols

The evaluation of the image-based localization was made on the "minnie" robot platform only. The 12 video sequences from this robot platform compose an image database with 11'363 images.

The annotation was performed in two annotation setups: random and video-vs-video. In both setup three image sets were always considered: labeled training and validation sets and an unlabeled set.

Random sampling

In the first setup, the random sampling was used by dividing the database in three sets. We simulated 8 supervision levels by setting the size of the training set to a percentage of the full corpus

1%	2%	3%	5%	10%	20%	30%	50%

The remaining images were split in two random halves and used respectively for validation and testing purposes. In order to account for the effects of random sampling, 10 fold sampling was made at each supervision level.

Video vs Video

In the second setup, two video sequences were considerated. One video was completely annotated while the second was used for evaluation purposes. The validation set was sampled randomly from the annotated video sequence.

With 12 video sequences under consideration, 12 pairs featuring the same video sequences were excluded from evaluation, thus resulting in 132 video pairs. We differentiate three sets of pairs: "SAME", "HARD" and "AVERAGE" result cases. The "SAME" set contains only the video sequence pairs where the light conditions are same and the recordings were made in a very short span of time. Countrary is for the set "HARD" by considering only different lighting condition and video sequences recorded with a large time span. The last set effectively contains all the 132 video pairs to provide an overall averaged performance.

Appendix E

The IMMED corpus

E.1 Overview

In this annex we describe the video corpus taken during the recording sessions of the IMMED project. The database consists of video sequences recorded using wearable video camera, which was attached to the right shoulder of a person. All the recordings were made indoors with occasional scenes of surrounding area outside of a house location.

The database of consists of various length video sequences recorded by patients and several volunteers in their ecological environment. There are 14 patient house locations with bootstrap and unlabeled video recordings, and 3 patient houses without bootstrap video sequence. Conversely, from 7 volunteer house locations, only one location has been recorded without bootstrap video sequence.

All recordings were carried out using the GoPro video camera with framerate of 30 fps which was then down-sampled to 5 fps. The average size of patients and volunteer video sequences used for bootstrap and testing is given in the table:

	Bootstrap	Unlabeled
Patients	6'400 images (3.5 min)	36'000 images (20 min)
Volunteers	3'000 images (1.6 min)	52'100 images (29 min)

The resolution of extracted images is 1280 x 960.

E.2 Acquisition protocol

The video recordings within the IMMED project have been captured in different houses. There are typically two video recordings from one house, which are available after such acquisition: a short bootstrap video and a longer unlabeled video with actual activites.

The bootstrap video is usually several minutes long as the patient records the representative topological locations in the house. This is the only supervised information available for the automatic localization estimation algorithms. The bootstrap video is annotated manually by the accompanying medical specialist using a specialized interface.

The unlabeled video captures actual activities and displacements of the patient in the house, and no manual annotation is performed for it.

We would like to outline the fact that the boostrap, its annotation and unlabeled videos are made available after the recording session such that no manual human effort is carried out afterwards.



Figure E.1: Sample images depicting different topological locations

E.3 Topological locations

Uniform location annotation was done for all the houses and yielded 6 topological locations (see Fig. E.1 for sample images):

bathroom	bedroom	kitchen	living room	outside	other

where the class "other" was used to designate specific and rarely visited topological locations in a house. The ground truth information for the frames found in transition between the topological locations was assigned arbitrary manner.

E.4 Evaluation protocols

The evaluation protocol for the IMMED corpus followed video versus video setup. For several locations the bootstrap video was not provided, thus a random part of such videos was used for training.

Bibliography

[1] The pyramid match kernel: discriminative classification with sets of image features. *IEEE International Conference on Computer Vision*, 2:1458–1465 Vol. 2, 2005.

[2] Azizi Abdullah, Remco Veltkamp, and Marco Wiering. Spatial pyramids and two-layer stacking SVM classifiers for image categorization: A comparative study. In *International Joint Conference on Neural Networks Neural Networks*, pages 5–12, 2009.

[3] Steven Abney. *Semisupervised Learning for Computational Linguistics*. Computer Science and Data Analysis Series. Chapman & Hall, University of Michigan, Ann Arbor, USA, July 2008.

[4] S Amini and F Razzazi. Confidence measure extraction for SVM speech classifiers using artificial neural networks. *Signal Processing*, 2008.

[5] S Amini and F Razzazi. A multi-class SVM based phonemes classifier based on a trainable confidence measure. *IEEE International Symposium on Signal Processing and Information Technology*, August 2010.

[6] Francis R Bach. Exploring Large Feature Spaces with Hierarchical Multiple Kernel Learning. *Neural Information Processing Systems Foundation*, 2008.

[7] Francis R Bach and Gert R G Lanckriet. Multiple Kernel Learning, Conic Duality, and the SMO Algorithm. *International Conference on Machine Learning*, August 2004.

[8] Lamberto Ballan, Marco Bertini, Alberto del Bimbo, and Giuseppe Serra. Video event classification using bag of words and string kernels. *International Image Analysis and Processing*, pages 170–178, 2009.

[9] M Bardideh and F Razzazi. An SVM based Confidence Measure for Continuous Speech Recognition. *Signal Processing and Communications*, 2007.

[10] Herbert Bay, Tinne Tuytelaars, and L Van Gool. SURF: speeded-up robust features. *European Conference on Computer Vision*, 110(3):346–359, 2008.

[11] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, 1:585–592, 2002.

[12] Mikhail Belkin and Partha Niyogi. Using manifold structure for partially labeled classification. *Neural Information Processing Systems*, 2002.

[13] Mikhail Belkin and Partha Niyogi. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 2006.

[14] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–46, November 2008.

[15] Serge J. Bengio, JF Paiement, P Vincent, O Delalleau, N Le Roux, and M Ouimet. Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. *Advances in Neural Information Processing Systems*, 16:177, 2004.

[16] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, Secaucus, NJ, USA, 2006.

[17] Avrim Blum and Tom Mitchell. Combining Labeled and Unlabeled Data with Co-Training. *Conference on Computational Learning Theory*, October 1998.

[18] Anna Bosch. Image Classification for Large Number of Object Categories. *Master's Thesis*, pages 1–207, July 2007.

[19] Anna Bosch, Xavier Munoz, and Robert Marti. Which is the best way to organize/classify images by content? *Image and vision computing*, 25(6), 2007.

[20] Anna Bosch, Andrew Zisserman, and Xavier Munoz. Image classification using rois and multiple kernel learning. *International journal of computer vision*, 2008:1–25, 2008.

[21] Bernhard Boser, Isabelle Guyon, and Vladimir Vapnik. A training algorithm for optimal margin classifiers. *Proceedings of the 5th annual ACM Workshop on Computational Learning Theory*, pages 144–152, 1992.

[22] E Brunskill, T Kollar, and N Roy. Topological mapping using spectral clustering and classification. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3491–3496, 2007.

[23] Heinrich Bulthoff and Alan Yuille. Bayesian Models for Seeing Shapes and Depth. *Harvard Robotics Laboratory Technical Report Technical report No. 90-11*, November 1990.

[24] Gertjan J. Burghouts and Jan-Mark Geusebroek. Performance evaluation of local colour invariants. *Computer Vision and Image Understanding*, 113:48–62, January 2009.

[25] Shih-Fu Chang, Dan Ellis, Wei Jiang, Keansub Lee, Akira Yanagawa, Alexander C Loui, and Jiebo Luo. Large-scale multimodal semantic concept detection for consumer video. In *International workshop on Workshop on multimedia information retrieval*. ACM Request Permissions, September 2007.

[26] Olivier Chapelle and Bernhard Scholkopf. Incorporating Invariances in Nonlinear Support Vector Machines. *Neural Information Processing Systems*, June 2002.

[27] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

[28] Olivier Chapelle, Vikas Sidhwani, and Sathiya Keerthi. Optimization Techniques for Semi-Supervised Support Vector Machines. *The Journal on Machine Learning Research*, 9:203–233, July 2008.

[29] Olivier Chapelle, Jason Weston, and Bernhard Scholkopf. Cluster kernels for semi-supervised learning. *Neural Information Processing Systems Foundation*, 2002.

[30] Youcef Chibani. Integrating class-dependant tangent vectors into SVMs for handwritten digit recognition. *International Conference on Signals, Circuits and Systems*, pages 1–4, 2009.

[31] Fan R. K. Chung. *Spectral Graph Theory*. CBMS Regional Conference Series in Mathematics, No. 92. American Mathematical Society, 1997.

[32] Ciaran O Conaire, Michael Blighe, and Noel O'Connor. Sensecam image localisation using hierarchical surf trees. *International Multimedia Modeling Conference*, page 15, 2009.

[33] Matthieu Cord, Sylvie Philipp-Foliguet, Philippe-Henri Gosselin, and Jérôme Fournier. Interactive exploration for image retrieval. *EURASIP journal on Applied Signal Processing*, pages 2173–2186, 2005.

[34] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 1995.

[35] Koby Crammer, Joseph Keshet, and Yoram Singer. Kernel Design Using Boosting. *Advances in Neural Information Processing Systems*, May 2003.

[36] Nello Cristianini, John Shawe-Taylor, Andre Elissee, and Jaz Kandola. On Kernel-Target Alignment. *The Journal of Machine Learning Research*, 2002.

[37] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Williamowski, and Cedric Bray. Visual Categorization with Bag of Keypoints. *European Conference on Computer Vision*, September 2004.

[38] Mark Culp and George Michailidis. An iterative algorithm for extending learners to a semi-supervised setting. *Journal of Computational and Graphical Statistics*, 17(3):545–571, 2008.

[39] Dennis DeCoste and Michael Burl. Distortion-invariant recognition via jittered queries. In *Computer Vision and Pattern Recognition*, pages 732–737. Machine Learning Systems Group, Jet Propulsion Laboratory, California Institute of Technology, 2000.

[40] BC Dickerson, TR Stoub, RC Shah, and RA Sperling. Alzheimer-signature MRI biomarker predicts AD dementia in cognitively normal adults. *Neurology*, 76(16):1395–1402, 2011.

[41] M.W.M.G Dissanayake, P Newman, S Clark, H.F Durrant-Whyte, and M Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, 2001.

[42] Cailing Dong, Yilong Yin, Xinjian Guo, Gongping Yang, and Guangtong Zhou. On Co-Training Style Algorithms. *International Conference on Natural Computation*, 7:196–201, 2008.

[43] J Drish. Obtaining calibrated probability estimates from support vector machines. *Neural computation*, 7(2):219–269, 1995.

[44] Kai-Bo Duan and Sathiya Keerthi. Which is the best multiclass SVM method? An empirical study. *International Conference on Multiple Classifier Systems*, 2005.

[45] Lixin Duan, Dong Xu, Ivor W Tsang, and Jiebo Luo. Visual Event Recognition in Videos by Learning from Web Data. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2010.

[46] Richard Duda, Peter Hart, and David Stork. *Pattern Classification*. John Wiley & Sons, 2001.

[47] Nelson Dunford and Jacob T Schwarz. *Linear Operators*. Wiley Classics Library. Wiley-Interscience, August 1988.

[48] Li Fei-Fei, R Fergus, and Pietro Perona. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. *Conference on Computer Vision and Pattern Recognition*, page 178, 2004.

[49] Rob Fergus, Pietro Perona, and Andrew Zisserman. Object Class Recognition by Unsupervised Scale-Invariant Learning. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, April 2003.

[50] Rob Fergus, Y Weiss, and Antonio Torralba. Semi-supervised learning in gigantic image collections. *Advances in Neural Information Processing Systems*, 1, 2009.

[51] Jérôme Fournier and Matthieu Cord. Retin: A content-based image indexing and retrieval system. *Pattern Analysis & Applications*, 2001.

[52] Charless Fowlkes, Serge Belongie, Fan Chung, and Jitendra Malik. Spectral grouping using the Nystrom method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225, 2004.

[53] Andrew Fraser, Nicolas Hengartner, Kevin Vixie, and Brendt Wohlberg. Incorporating invariants in Mahalanobis distance based classifiers: application to face recognition. In *International Joint Conference on Neural Networks*, pages 3118–3123. Los Alamos National Laboratory, 2003.

[54] Peter Gehler and Sebastian Nowozin. On feature combination for multiclass object classification. *International Conference on Computer Vision*, pages 221–228, 2009.

[55] Sally Goldman and Yan Zhou. Enhancing Supervised Learning with Unlabeled Data. *International Conference in Machine Learning*, 2000.

[56] Philippe-Henri Gosselin and Matthieu Cord. Active Learning Methods for Interactive Image Retrieval. *Image Processing, IEEE Transactions on*, 17(7):1200–1211, 2008.

[57] Bernard Haasdonk and Hans Burkhardt. Invariant kernel functions for pattern analysis and machine learning. *Machine learning*, 2007.

[58] Bernhard Haasdonk and Daniel Keysers. Tangent distance kernels for support vector machines. In *International Conference on Pattern Recognition*, pages 864–868. Computer Science Department, Albert-Ludwigs University Freiburg, 2002.

[59] Trevor Hastie, Robert Tibshirani, J Friedman, and J Franklin. *The elements of statistical learning: data mining, inference and prediction*, volume 27. The Mathematical Intelligencer, 2005.

[60] M Hearst. *Noun homonym disambiguation using local context in large text corpora*. In Proceedings, Seventh Annual Conference of the UW Centre for the New OED and Text Research, 1991.

[61] Don Hindle and Mats Rooth. Structural ambiguity and lexical relations. *Annual Meeting of the Association for Computational Linguistics*, 1991.

[62] N Ikizler-Cinbis, R.G Cinbis, and S Sclaroff. Learning actions from the Web. In *International Conference on Computer Vision*, pages 995–1002, 2009.

[63] Yu-Gang Jiang, Chong-Wah Ngo, and Jun Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. *International Conference on Image and Video Retrieval*, 2007.

[64] Luo Jie, Francesco Orabona, M Fornoni, Barbara Caputo, and N Cesa-Bianchi. OM-2: An Online Multi-class Multi-kernel Learning Algorithm. 2010.

[65] Thorsten Joachims. Transductive inference for text classification using support vector machines. *Internation Conference in Machine Learning*, 1999.

[66] I Jolliffe. Principal component analysis. *Wiley Online Library*, 2002.

[67] Aniruddha Kembhavi, Behjat Siddiquie, Roland Miezianko, Scott McCloskey, and Larry S Davis. Incremental Multiple Kernel Learning for Object Recognition. pages 1–8, July 2009.

[68] Josef Kittler, Ali Ahmadyfard, and David Windridge. Serial Multiple Classifier Systems Exploiting a Coarse to Fine Output Coding. *International Conference on Multiple Classifier Systems*, 2709(Chapter 11):106–114, June 2003.

[69] Josef Kittler, Mohamad Hatef, Robert P W Duin, and Jiri Matas. On Combining Classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:226–239, 1998.

[70] Piotr Koniusz and Krystian Mikolajczyk. Spatial Coordiante Coding To Reduce Histogram Representations Dominant Angle and Colour Pyramid Match. *International Conference on Image Processing*, pages 1–4, September 2011.

[71] Harold W. Kuhn. Nonlinear programming: a historical view. *SIGMAP Bull.*, pages 6–18, June 1982.

[72] Shailesh Kumar, Joydeep Ghosh, and Melba M Crawford. Hierarchical Fusion of Multiple Classifiers for Hyperspectral Data Analysis. *Pattern Analysis & Applications*, 5(2):210–220, June 2002.

[73] Ludmila Ilieva Kuncheva. *Combining pattern classifiers.* methods and algorithms. Wiley-Interscience, 2004.

[74] Gert R G Lanckriet, Nello Cristianini, P Barlett, L El Ghaoui, and Michael I Jordan. Learning the Kernel Matrix with Semidefinite Programming. *Journal of Machine Learning Research*, 5(27-72):1–46, January 2004.

[75] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and B Rozenfeld. Learning realistic human actions from movies. In *Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[76] Fabien Lauer and Gerard Bloch. Incorporating prior knowledge in support vector machines for classification: A review. *Neurocomputing*, 71(7-9):1578–1594, 2008.

[77] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *Computer Vision and Pattern Recognition*, 2:2169–2178, 2006.

[78] HT Lin and CJ Lin. A note on Platt's probabilistic outputs for support vector machines. *Machine learning*, 68(3):267–276, 2007.

[79] Oskar Linde and Tony Lindeberg. Object Recognition using Composed Receptive Field Histograms of Higher Dimensionality. *International Conference on Pattern Recognition*, 2:1–6 Vol.2, 2004.

[80] Jingen Liu, Jiebo Luo, and M Shah. Recognizing realistic actions from videos "in the wild". In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1996–2003, 2009.

[81] Gaelle Loosli, Stepahan Canu, S V N Vishwanathan, and Alex Smola. Invariances in classification: an efficient svm implementation. *Applied Stochastic Models*, 2005.

[82] Ana Lopes, Rodrigo Oliveira, Jussara de Almeida, and Arnoldo de A Araujo. Spatio-Temporal Frames in a Bag-of-Visual-Features Approach for Human Actions Recognition. *Brazilian Symposium on Computer Graphics and Image Processing*, pages 315–321, 2009.

[83] Alexander Loui, Jiebo Luo, Shih-Fu Chang, Dan Ellis, Wei Jiang, Lyndon Kennedy, Keansub Lee, and Akira Yanagawa. Kodak's consumer video benchmark data set: concept definition and annotation. In *International workshop on Workshop on multimedia information retrieval*. ACM Request Permissions, September 2007.

[84] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[85] Fuxiang Lu, Xiaokang Yang, Weiyao Lin, Rui Zhang, and Songyu Yu. Image classification with multiple feature channels. *Optical Engineering*, 50(5), 2011.

[86] Jie Luo, Andrzej Pronobis, Barbara Caputo, and Patric Jensfelt. The KTH-IDOL2 Database. Technical report, Kungliga Tekniska Hoegskolan, CVAP/CAS, 2006.

[87] Helumut Lutkepohl and Gotz Trenkler. Handbook of Matrices. *Computational Statistics and Data Analysis*, New York: John Wiley and Sons(2):ISBN 0–471–96688–6, 1996.

[88] W.Y Ma and B.S Manjunath. NeTra: a toolbox for navigating large image databases. In *Multimedia Systems*, pages 568–571, 1997.

[89] Sebastian Maldonado and Gonzalo Paredes. A semi-supervised approach for reject inference in credit scoring using SVMs. *Advances in Data Mining. Applications and Theoretical Aspects*, pages 558–571, 2010.

[90] Ramin Mehran and Amin Shali. A Statistical Correction-Rejection Strategy for OCR Outputs in Persian Personal Information Forms. *International Conference on Information Technology and Applications*, 2004.

[91] Stefano Melacci and Mikhail Belkin. Laplacian Support Vector Machines Trained in the Primal. *Journal of Machine Learning Research*, 12:1149–1184, 2011.

[92] Krystian Mikolajczyk and Cordelia Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:1615–1630, October 2005.

[93] Michael Milford and Gordon Wyeth. Mapping a Suburb With a Single Camera Using a Biologically Inspired SLAM System. *IEEE Transactions on Robotics*, 24(5):1038–1053, 2008.

[94] Oscar Martinez Mozos and Wolfram Burgard. Supervised Learning of Topological Maps using Semantic Information Extracted from Range Data. In *International Conference on Intelligent Robots and Systems*, pages 2772–2777, 2006.

[95] Boaz Nadler and Nathan Srebro. Semi-Supervised Learning with the Graph Laplacian: The Limit of Infinite Unlabelled Data. *Conference on Neural Information Processing Systems*, 2009.

[96] Shinichi Nakajima, Alexander Binder, Christiana Müller, Wojciech Wojcikiewicz, Marius Kloft, Ulf Brefeld, Klaus-Robert Muller, and Motoaki Kawanabe. Multiple kernel learning for object classification. *Workshop on Information-based Induction Sciences*, 2009.

[97] Wanas Nayer. Feature Based Architecture for Decision Fusion. *phD thesis*, pages 1–144, May 2003.

[98] M Nilsback and Barbara Caputo. Cue integration through discriminative accumulation. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:II–578 – II–585 Vol.2, 2004.

[99] C. N. NiScanail, S. Carew, P. Barralon, N. Noury, D. Lyons, and G. M. Lyons. A Review of Approaches to Mobility Telemonitoring of the Elderly in Their Living Environment. *Annals of Biomedical Engineering*, 34(4):547–563, April 2006.

[100] David Nister and H Stewenius. Scalable Recognition with a Vocabulary Tree. *Computer Vision and Pattern Recognition*, 2:2161–2168, 2006.

[101] Partha Niyogi, F Girosi, and T Poggio. Incorporating prior information in machine learning by creating virtual examples. In *Proceedings of the IEEE*, pages 2196–2209. MIT Center for Biological and Computational Learning, 1998.

[102] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. (2nd edition). Springer-Verlag, Berlin, New York, August 2006.

[103] Aude Oliva and Antonio Torralba. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. *International Conference on Computer Vision*, 42:145–175, May 2001. http://people.csail.mit.edu/torralba/code/spatialenvelope/.

[104] J Platt. *Probabilities for SV Machines, Advances in Large Margin Classifiers*. MIT Press, Cambridge, 1999.

[105] John C Platt. Fast training of support vector machines using sequential minimal optimization. In Bernhard Scholkopf, Christopher Burges, and Alexander Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 185–208. MIT Press, 1999.

[106] Alexei Pozdnoukhov. Prior Knowledge in Kernel Methods. Technical report, IDIAP Research Insititute, 2006.

[107] Alexei Pozdnoukhov and Samy Bengio. Tangent vector kernels for invariant image classification with SVMs. In *International Conference on Pattern Recognition*, pages 486–489. IDIAP, 2004.

[108] Alexei Pozdnoukhov and Samy Bengio. A Kernel Classifier for Distributions. *IDIAP Research Report*, 2005.

[109] Alexei Pozdnoukhov and Samy Bengio. Graph-based transformation manifolds for invariant pattern recognition with kernel methods. In *International Conference on Pattern Recognition*, pages 1228–1231. IDIAP Research Institute, 2006.

[110] Alexei Pozdnoukhov and Samy Bengio. Improving Kernel Classifiers for Object Categorization Problems. *International Conference in Machine Learning*, 2006.

[111] Alexei Pozdnoukhov and Samy Bengio. Invariances in kernel methods: From samples to objects. *Pattern Recognition Letters*, 27(10):1087–1097, July 2006.

[112] Andrzej Pronobis and Barbara Caputo. Confidence-Based Cue Integration for Visual Place Recognition. *International Conference on Intelligent Robots and Systems*, pages 2394–2401, 2007.

[113] Andrzej Pronobis, Barbara Caputo, Patric Jensfelt, and HI Christensen. A discriminative approach to robust visual place recognition. *International Conference on Intelligent Robots and Systems*, pages 3829–3836, 2006.

[114] Andrzej Pronobis, Barbara Caputo, Patric Jensfelt, and HI Christensen. A realistic benchmark for visual indoor place recognition. *Robotics and Autonomous Systems*, 58(1):81–96, 2010.

[115] Andrzej Pronobis, O Martínez Mozos, Barbara Caputo, and Patric Jensfelt. Multi-modal semantic place classification. *The International Journal of Robotics Research*, 29(2-3):298, 2010.

[116] Andrzej Pronobis, M Mozos, and Barbara Caputo. SVM-based discriminative accumulation scheme for place recognition. *IEEE International Conference on Robotics and Automation Robotics and Automation*, pages 522–529, 2008.

[117] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[118] Alain Rakotomamonjy, Francis R Bach, Stepahan Canu, and Yves Grandvalet. SimpleMKL. *The Journal of Machine Learning Research*, 9:2491–2521, 2008.

[119] T Rose, J Fiscus, P Over, J Garofolo, and M Michel. The TRECVid 2008 Event Detection evaluation. *Workshop on Applications of Computer Vision*, pages 1–8, 2009.

[120] Stefan Rüping. A Simple Method For Estimating Conditional Probabilities For SVMs. *American Society of Agricultural Engineers*, July 2004.

[121] Hichem Sahbi, P Etyngier, JY Audibert, and R Keriven. Manifold learning using robust Graph Laplacian for interactive image search. *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[122] Koen E A van de Sande, Theo Gevers, and Cees G M Snoek. Evaluating Color Descriptors for Object and Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, July 2009.

[123] Grant Schindler, Matthew Brown, and Richard Szeliski. City-Scale Location Recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2007.

[124] Bernhard Scholkopf, Chris Burges, and Vladimir Vapnik. Incorporating invariances in support vector learning machines. *International Conference on Artificial Networks*, 1996.

[125] Bernhard Scholkopf and Alexander J Smola. *Learning with Kernels*. MIT Press, Cambridge, 2002.

[126] C Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local SVM approach. In *International Conference on Pattern Recognition*, pages 32–36, 2004.

[127] Niku Sebe, MS Lew, X Zhou, TS Huang, and EM Bakker. The State of the Art in Image and Video Retrieval. *International Conference on Image and Video Retrieval*, pages 1–7, May 2003.

[128] Burr Settles. Active Learning Literature Survey. Technical Report 1648, University of Wisconsin-Madison, 2009.

[129] Mohammad Shahiduzzaman, Dengsheng Zhang, and Guojun Lu. Improved Spatial Pyramid Matching for Image Classification. *Asian Conference on Computer Vision*, pages 1–11, November 2010.

[130] Jontahon Shlens. A tutorial on principal component analysis. *Systems Neurobiology Laboratory, University of California at San Diego*, 2005.

[131] Patrice Simard, Yann LeCun, John Denker, and Bernard Victorri. Transformation invariance in pattern recognition—tangent distance and tangent propagation. *Neural Networks*, 1998.

[132] Vikas Sindhwani and Sathiya Keerthi. Large scale semi-supervised linear SVMs. *29th Annual International ACM SIGIR*, 2006.

[133] Aarti Singh, Robert Nowak, and Zhu Xiaojin. Unlabeled data: Now it helps, now it doesn't. *Advances in neural information processing*, 2008.

[134] Josef Sivic and Andrew Zisserman. Video Google: A text retrieval approach to object matching in videos. *International Conference on Computer Vision*, 2003.

[135] AF Smeaton and P Over. Evaluation campaigns and TRECVid. In *International Workshop on Multimedia information Retrieval*, 2006.

[136] Alexander J Smola, Bernhard Scholkopf, and Klaus-Robert Muller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural computation*, 10(6):1299–1319, 1998.

[137] Yan Song, Yan-Tao Zheng, Sheng Tang, Xiangdong Zhou, Yongdong Zhang, Shouxun Lin, and Tat-Seng Chua. Localized Multiple Kernel Learning for Realistic Human Action Recognition in Videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(9):1193–1202, 2011.

[138] Soren Sonnenburg, Gunnar Ratsch, C Schafer, and Bernhard Scholkopf. Large Scale Multiple Kernel Learning. *Journal of Machine Learning Research*, 7(1-35):1531–1565, July 2006.

[139] Alexander Statnikov, Constantin Aliferis, and Ioannis Tsamardinos. A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics*, 21(5), 2005.

[140] Fan Sun and Maosong Sun. A new transductive support vector machine approach to text categorization. In *IEEE International Conference on Natural Language Processing and Knowledge Engineering*, pages 631–635, 2005.

[141] Ju Sun, Xiao Wu, Shuicheng Yan, Loong-Fah Cheong, Tat-Seng Chua, and Jintao Li. Hierarchical spatio-temporal context modeling for action recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 2004–2011, 2009.

[142] Martin Szummer and Rosalind W Picard. Indoor-outdoor image classification. *IEEE International Workshop on Content-Based Access of Image and Video Database*, pages 42–51, 1998.

[143] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition, Fourth Edition.* Academic Press. Elsevier, 2008.

[144] Michael Tipping E. Bayesian inference: An introduction to principles and practice in machine learning. *Advanced lectures on machine Learning*, pages 41–62, 2004.

[145] Tatiana Tommasi, Francesco Orabona, and Barbara Caputo. An svm confidence-based approach to medical image annotation. *Evaluating Systems for Multilingual and Multimodal Information Access*, pages 696–703, 2009.

[146] Frank Tompkins and Patrick Wolfe. Image analysis with regularized Laplacian eigenmaps. In *IEEE International Conference on Image Processing*, pages 1913–1916, 2010.

[147] Wei Tong, Tianbao Yang, and Rong Jin. Co-training For Large Scale Image Classification: An Online Approach. *Analysis and Evaluation of Large-Scale Multimedia Collections*, pages 1–4, May 2010.

[148] N Tornatis, I Nourbakhsh, and R Siegwart. Hybrid simultaneous localization and map building: closing the loop with multi-hypotheses tracking. In *International Conference on Robotics and Automation*, pages 2749–2754, 2002.

[149] Antonio Torralba, K Murphy, W Freeman, and M Rubin. Context-based vision system for place and object recognition. *IEEE International Conference on Computer Vision*, pages 273–280 vol.1, 2003.

[150] Andreas Uhl and Peter Wild. Parallel versus Serial Classifier Combination for Multibiometric Hand-Based Identification. *International Conference on Advances in Biometrics*, 5558(Chapter 96):950–959, 2009.

[151] Iwan Ulrich and Illah Nourbakhsh. Appearance-based place recognition for topological localization. In *International Conference on Robotics and Automation*, pages 1023–1029. The Robotics Institute, Carnegie Mellon University, 2000.

[152] Eduardo Valle and Matthieu Cord. Advanced Techniques in CBIR Local Descriptors, Visual Dictionaries and Bags of Features. *Computer Graphics and Image Processing*, pages 1–7, October 2009.

[153] V. E. van Beusekom, I. G. Sprinkuizen-Kuyper, and L. G. Vuurpul. Empirically Evaluating Co-Training. *Student Report*, 2009.

[154] Vladimir Vapnik. Pattern recognition using generalized portrait method. *Automation and Remote Control*, 1963.

[155] Vladimir Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag. New York, 1995.

[156] Vladimir Vapnik. *Statistical Learning Theory*. Wiley-Interscience. NY: Wiley, 1998.

[157] Manik Varma and Bodla Rakesh Babu. More Generality in Efficient Multiple Kernel Learning. *International Conference on Machine Learning*, 2009.

[158] Manik Varma and Debajyoti Ray. Learning The Discriminative Power-Invariance Trade-Off. *International Conference on Computer Vision*, pages 1–8, 2007.

[159] Andrea Vedaldi, Varun Gulshan, Manik Varma, and Andrew Zisserman. Multiple kernels for object detection. *International Conference on Computer Vision*, pages 606–613, 2009.

[160] S V N Vishwanathan, Zhaonan Sun, Nawanol Theera-Ampornpunt, and Manik Varma. Multiple Kernel Learning and the SMO Algorithm. *Advances in Neural Information Processing Systems*, 23(1-9):2361–2369, November 2010.

[161] Julia Vogel, Adrian Schwaninger, Christian Wallraven, and Heinrich H. B u lthoff. Categorization of natural scenes: Local versus global information and the role of color. *ACM Transactions on Applied Perception*, 4, November 2007.

[162] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[163] Fei Wang and Changshui Zhang. Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, pages 55–67, 2007.

[164] Meng Wang, Xian-Sheng Hua, Li-Rong Dai, and Yan Song. Enhanced Semi-Supervised Learning for Automatic Video Annotation. *International Conference on Multimedia and Expo*, pages 1–4, June 2006.

[165] Wei Wang and Zhi-Hua Zhou. Analyzing Co-training Style Algorithms. *European Conference on Machine Learning*, 2007.

[166] Wang Wei and Zhou Zhi-Hua. A New Analysis of Co-Training. *International Conference in Machine Learning*, pages 1135–1142, May 2010.

[167] J Wolf, W Burgard, and H Burkhardt. Robust vision-based localization by combining an image-retrieval system with Monte Carlo localization. *IEEE Transactions on Robotics*, 21(2):208–216, 2005.

[168] Jianxin Wu and James M Rehg. CENTRIST: A visual descriptor for scene categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[169] Dong Xu, Tat-Jen Cham, Shuicheng Yan, and Shih-Fu Chang. Near duplicate image identification with patially Aligned Pyramid Matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2008.

[170] Dong Xu and Shih-Fu Chang. Visual Event Recognition in News Video using Kernel Methods with Multi-Level Temporal Alignment. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[171] Jingjing Yang, Yuanning Li, Yonghong Tian, Lingyu Duan, and Wen Gao. Group-sensitive multiple kernel learning for object categorization. *International Conference on Computer Vision*, pages 436–443, 2009.

[172] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Annual Meeting of the Association for Computational Linguistics*, pages 189–196. University of Pennsylvania, 1995.

[173] Tom Yeh, Konrad Tollmar, and Trevor Darrell. Searching the web with mobile images for location recognition. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004.

[174] Chunjie Zhang, Jing Liu, Jinqiao Wang, Qi Tian, Changsheng Xu, Hanqing Lu, and Songde Ma. Image Classification Using Spatial Pyramid Coding and Visual Word Reweighting. *Asian Conference on Computer Vision*, pages 1–11, November 2010.

[175] Dell Zhang and Wee Sun Lee. Validating co-training models for web image classification. In *Proceedings of SMA Annual Symposium*. National University of Singapore, 2005.

[176] Min-Ling Zhang and Zhi-Hua Zhou. CoTrade: Confident Co-Training With Data Editing. *IEEE Transactions on Systems, Man, and Cybernetics*, (99):1–15, 2011.

[177] Wei Zhang and Jana Kosecka. Image based localization in urban environments. *3D Data Processing, Visualization, and Transmission*, 2006.

[178] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Scholkopf. Learning with Local and Global Consistency. *Advances in Neural Information Processing Systems*, February 2004.

[179] Xueyuazan Zhou and Belkin Mikhail. Semi-supervised Learning by Higher Order Regularization. *The Journal of Machine Learning Research*, 2011.

[180] Yan Zhou and Sally Goldman. Democratic co-learning. In *IEEE International Conference on Tools with Artificial Intelligence*, pages 594–602. University of South Alabama, 2004.

[181] Xiaojin Zhu. Semi-Supervised Learning Literature Survey. *Technical Report 1530, Department of Computer Sciences, University of Winsconsin, Madison*, pages 1–60, July 2008.

[182] Xiaojin Zhu and Zoubin Ghahramani. Learning from Labeled and Unlabeled Data with Label Propagation. *Technical Report CMU-CALD-02-107*, June 2003.

[183] Xiaojin Zhu and Andrew B. Goldberg. *Introduction to Semi-Supervised Learning*. Morgan and Claypool Publishers, 2009.

[184] Nadia Zouba, Francois Bremond, Monique Thonnat, and Vu Van Thinh. Multi-sensors Analysis for Everyday Activity Monitoring. *SETIT, Tunisie*, January 2007.

[185] Zoran Zovkovic, Olaf Booij, and Ben Krose. From images to rooms. *Robotics and Autonomous Systems*, 2007.

Appendix F

List of Contributed Articles

F.1 Published and accepted

1. **V. Dovgalecs**, S. Ilcus, R. Mégret and Y. Berthoumieu, "*Pyramides spatiales d'histogrammes invariantes aux transformations pour la reconnaissance de lieux*", Reconnaissance des Formes et Intelligence Artificielle, Lyon, France, 2012.

2. H. Wannous, **V. Dovgalecs**, R. Mégret and M. Daoudi, "*Place Recognition via 3D Modeling for Personal Activity LifeLog using Wearable Camera*", The 18th International Conference on MultiMedia Modeling, Klagenfurt, Austria, January 4-6, 2012.

3. **V. Dovgalecs**, R. Mégret, Y. Berthoumieu. "*Time-aware Co-Training for Indoors Localization in Visual Lifelogs*", ACM Multimedia, Scottsdale, Arizona, 2011.

4. S. Karaman, J.Benois-Pineau, **V. Dovgalecs**, R. Mégret et al., "*Hierarchical Hidden Markov Model in Detecting Activities of Daily Living in Wearable Videos for Studies of Dementia*", Multimedia Tools and Applications, CBMI 2011.

5. **V. Dovgalecs**, R. Mégret, H. Wannous, Y. Berthoumieu. "*Semi-Supervised Learning for Location Recognition from Wearable Video*". CBMI 2010, Grenoble.

6. R. Mégret, **V. Dovgalecs**, H. Wannous, S. Karaman, J. Benois-Pineau, E. El Khoury, J. Pinquier, P. Joly, R. André-Obrecht, Y. Gaëstel, J-F. Dartigues. "*The IMMED Project: Wearable Video Monitoring of People with Age Dementia*" ACM Multimedia 2010, Florence, Italie. Video session.

7. S. Karaman, J. Benois-Pineau, R. Mégret, **V. Dovgalecs**, Y. Gaëstel, J.-F. Dartigues. "*Human Daily Activities Indexing in Videos from Wearable Cameras for Monitoring of Patients with Dementia Diseases*". ICPR 2010, Istanbul.